

Received 18 February 2025, accepted 14 March 2025, date of publication 24 March 2025, date of current version 11 April 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3554154



Recursive Forward-Backward EDMD: Guaranteed Algebraic Search for Koopman Invariant Subspaces

MASIH HASELI[✉] AND JORGE CORTÉS[✉], (Fellow, IEEE)

Department of Mechanical and Aerospace Engineering, University of California San Diego, La Jolla, CA 92093, USA

Corresponding author: Masih Haseli (mhaseli@ucsd.edu)

This work was supported by the Office of Naval Research (ONR) under Award N00014-23-1-2353.

ABSTRACT The implementation of the Koopman operator on digital computers often relies on the approximation of its action on finite-dimensional function spaces. This approximation is generally done by orthogonally projecting on the subspace. Extended Dynamic Mode Decomposition (EDMD) is a popular, special case of this projection procedure in a data-driven setting. Importantly, the accuracy of the model obtained by EDMD depends on the quality of the finite-dimensional space, specifically on how close it is to being invariant under the Koopman operator. This paper presents a data-driven algebraic search algorithm, termed Recursive Forward-Backward EDMD, for subspaces close to being invariant under the Koopman operator. Relying on the concept of temporal consistency, which measures the quality of the subspace, our algorithm recursively decomposes the search space into two subspaces with different prediction accuracy levels. The subspace with lower level of accuracy is removed if it does not reach a satisfactory threshold. The algorithm allows for tuning the level of accuracy depending on the underlying application and is endowed with convergence and accuracy guarantees.

INDEX TERMS Accuracy bound, algebraic algorithm, dynamic mode decomposition, invariant subspace, Koopman operator, unknown nonlinear system.

I. INTRODUCTION

Koopman operator theory has emerged as an alternative viewpoint of dynamical systems by encoding nonlinear behavior as a linear operator acting on a vector space of functions. This viewpoint allows for the use of spectral methods to analyze nonlinear systems. A major appeal of Koopman operator theory comes from the fact that it provides a unified framework to study general nonlinear systems, since the operator is always linear independently of the system's structure. This has led to a broad range of applications, specially in data-driven learning. A major challenge, however, is that the operator is often defined on infinite-dimensional spaces, thus requiring infinite computational/memory capabilities unless additional structure is identified/imposed. A popular way to address this challenge

The associate editor coordinating the review of this manuscript and approving it for publication was Hassen Ouakad[✉].

is to approximate the operator's action on finite-dimensional spaces. The accuracy of this approximation depends on how close the finite-dimensional subspace is to being invariant under the Koopman operator. In this paper, we consider the problem of identifying close-to-invariant subspaces from data with tunable levels of accuracy.

A. LITERATURE REVIEW

Koopman operator theory represents a nonlinear dynamical system via a linear operator acting on a vector spaces of functions [1], [2]. This operator-based viewpoint naturally lends itself to spectral methods [3], [4] with algebraic descriptions that are much easier to analyze for complex systems compared to traditional geometric state-space representations [5]. This has led to many applications in a number of different fields. In complex system analysis, the Koopman operator has been used in model reduction [6], [7], [8], and the study of fluid flows [9], [10], [11], biological

systems [12], [13], [14], [15], power networks [16], [17], [18], traffic systems [19], [20], [21], and fault detection in complex systems [22], [23], [24]. Recent works utilize Koopman operator in cryptography [25] and signal processing [26]. In systems and control theory, the Koopman operator has seen a recent surge in popularity in both theoretical and practical domains. In particular, the work [27] provides several criteria for global stability of attractors based on the eigenfunctions of the Koopman operator associated with the system. The work [28] studies the relationship between Koopman operator and contraction theory. The work [29] provides a Koopman-based data-driven method to approximate auxiliary functions, including Lyapunov functions used for stability analysis of attractors. In addition, [30], [31] take different approaches to learn Lyapunov functions. Koopman-based methods have also found their way in the stability analysis of switched nonlinear systems [32] and the estimation of regions of attraction [33], [34], [35], [36]. Although the original definition of Koopman operator does not allow for inputs to the system, the theory has been extended to control systems [6], [37], [38], [39], allowing for the utilization of tools from control theory such as feedback linearization [40], control Lyapunov functions [41], optimal control [42], [43], [44], [45], robust control [46], and eigenstructure assignment [47]. The advances in control have also led to applications in general areas of robotics [48], [49], [50], [51] as well as specific applications in soft robots [52], [53], nano-positioning systems [54], micro-electromechanical systems [55], sensing [56], pose estimation [57], trajectory tracking [58], and space applications [59]. Other works have relied on Koopman operator theory for safety critical control [60], [61], [62] and reachability analysis [63], [64], [65], [66], [67].

Given the infinite-dimensional nature of the Koopman operator, many applications above rely on representations on finite-dimensional spaces. In general, it is very difficult, and in some cases impossible, to find exact and informative linear finite-dimensional representations based on the Koopman operator (we refer the reader to [68] and [69] for results on the existence or non-existence of such representations). To address this, one often relies on finite-dimensional *approximated* models. Such approximations are generally constructed by composing the Koopman operator with a projection operator on the subspace of choice [70, Section 1.4]. Dynamic Mode Decomposition (DMD) [71] and its variant Extended Dynamic Mode Decomposition (EDMD) [72] are special cases of such projection-based methods. In particular, EDMD performs a data-driven orthogonal projection on a finite-dimensional space spanned by a predefined dictionary of functions. The work [73] studies the asymptotic properties of EDMD's solutions and their connection to the Koopman operator in cases where the size of the data set or the subspace's dimension go to infinity. Moreover, [74] provides probabilistic bounds on EDMD's accuracy given finite data sets.

Although finite-dimensional approximations of the Koopman operator are sought for their computational efficiency, the projection used for the approximation leads to two related but distinctly important problems: (i) the approximation can introduce spurious eigenfunctions, (ii) the approximation of arbitrary functions (not necessarily approximated eigenfunctions) suffers from errors resulting from the truncation. Even though these problems are closely related and have the same root, depending on the application at hand, one is more pronounced than the other. In applications that involve the spectral decompositions as a way of extracting important information from large-scale systems (such as model reduction for PDEs and applications in fluid dynamics, e.g. [8], [9], [10], and [11], or stability analysis, e.g [27]), directly addressing the spurious eigenfunctions, which lead to non-physical behavior, plays a central role (whereas the loss of information for arbitrary functions is more tolerable). On the other hand, for cases where the Koopman-based model is used for prediction of arbitrary functions (not just approximated eigenfunctions) and accuracy bounds are required, one should pay special attention to the prediction of all functions, which include the approximated eigenfunctions and all their linear combinations (see, e.g. [39]). Below, we briefly discuss different methods employed in the literature to address the aforementioned problems.

A simple observation reveals that on a Koopman-invariant subspace, the aforementioned problems do not arise, since the projection does not introduce any errors. Therefore, one often seeks to identify Koopman-invariant subspaces or subspaces that are close to invariant [75]. To this end, one can identify or approximate Koopman eigenfunctions and then use them to build approximately invariant subspaces [76], [77]. On the other hand, one can use optimization-based methods to learn approximate Koopman-invariant subspaces via neural networks [78], [79], [80] or rank-constraint semi-definite programs [81].

Alternatively, instead of relying on optimization, our previous work [82], [83] has used the algebraic structure of the Koopman operator and its eigenfunctions to provide data-driven necessary and almost surely sufficient conditions for identification of all Koopman eigenfunctions in any arbitrary finite-dimensional space of functions and synthesized algorithms to identify the maximal Koopman-invariant subspace in the original space. Subsequently, [84] designed an algebraic algorithm to identify subspaces that are close to invariant with tunable level of accuracy, accompanied by finite-iteration convergence guarantees. The aforementioned work uses the concept of invariance proximity to provide bounds on the relative approximation error of *all* functions (not necessarily eigenfunctions) in the identified subspace.

A different line of work that is more focused on identifying the spectral properties of the Koopman operator and addressing the spectral pollution and spurious eigenvalues is [85], which relies on the concept of Ritz residuals to measure the accuracy of eigenpairs and uses low-rank approximations to

improve the DMD method. The work [86] uses a similar residual term to measure the accuracy of eigenfunctions identified by DMD method. On the other hand, [87] blends these residual accuracy measures with a sparsity promoting optimization approach to identify accurate approximate eigenfunctions that are also informative, that is, their span is close to observables of choice (e.g. the system's state variables). An interesting, more recent line of work is residual Dynamic Mode Decomposition (ResDMD) [88], [89], which uses a similar concept of residuals to handle spectral pollution. ResDMD also provides methods to approximate the pseudospectra of the Koopman operator and provides various spectral bounds.

In this paper, we consider the use of EDMD forward and backward in time to approximate Koopman invariant subspaces with tunable accuracy. Throughout the literature, executing (E)DMD forward and backward in time has been used for different purposes, such as dealing with noisy data [90], [91], [92], training neural networks [93], and providing necessary and sufficient conditions on the identification of Koopman eigenfunctions [82]. Recently, we have employed in [94] EDMD forward and backward in time to introduce an error metric termed consistency index which computes a tight bound on the worst-case relative Koopman prediction error on a subspace.

B. STATEMENT OF CONTRIBUTIONS

We consider the problem of approximating Koopman-invariant subspaces from data. To this end, we present a recursive algebraic algorithm that searches through a vector space of functions (termed search space) and finds approximate subspaces close to invariant under the Koopman operator, whose accuracy can be tuned using a parameter. The proposed algorithm, termed Recursive Forward-Backward EDMD (RFB-EDMD), first checks the accuracy of Koopman-based predictions on the search space and, if it does not meet the desired level, decomposes the search space into two different vector spaces, and removes the one leading to the maximum error. RFB-EDMD iteratively performs the aforementioned divide and prune procedure until the remaining subspace meets the desired level of accuracy. We show that RFB-EDMD finds the solution in finitely many iterations and that the relative error of the Koopman-based EDMD predictor on the resulting subspace is bounded by the chosen accuracy parameter for all (*uncountably many*) functions in the identified subspace. In addition, we prove that RFB-EDMD always retains all Koopman eigenfunctions in the search space for all values of the accuracy parameter. RFB-EDMD naturally leads to an accuracy hierarchy on the subspaces of the search space in the form of a nested sequence of subspaces whose Koopman-prediction accuracy ranges from exact to the least accurate (which is the entire search space). All the members of this hierarchy are accessible through the choice of accuracy parameter tuned by the user based on the specific underlying

application. Finally, we demonstrate the versatility of the RFB-EDMD algorithm in capturing accurate and relevant dynamical information in a planar nonlinear system, the Van der Pol oscillator, the Duffing equation, and a 7-dimensional example describing yeast glycolysis.

We finish by describing the differences between the present work and the Tunable Symmetric Subspace Decomposition (T-SSD) algorithm proposed in our previous work [84]. While the T-SSD algorithm requires custom-made subroutines, the RFB-EDMD algorithm relies on well-established numerical routines that are highly efficient and robust to numerical errors and can be easily implemented on parallel processing hardware. Moreover, given the same accuracy level, RFB-EDMD finds more expressive subspaces (with larger dimension) as compared to T-SSD. Finally, RFB-EDMD enjoys additional theoretical properties as it establishes an accuracy hierarchy on identified subspaces.

C. NOTATION

We represent the sets of natural, real, and complex numbers by \mathbb{N} , \mathbb{R} , and \mathbb{C} , respectively. Given the sets S_1 and S_2 and the point x , the notation $x \in S_1$ means x is a member of S_1 ; $S_1 \cup S_2$, $S_1 \cap S_2$, and $S_1 \setminus S_2$ denote the union, intersection, and set difference of S_1 and S_2 . In addition, $S_1 \subseteq S_2$ and $S_1 \subsetneq S_2$ mean that S_1 is a subset and proper subset of S_2 , respectively. Considering matrix $A \in \mathbb{R}^{m \times n}$, we represent its transpose, range space, number of columns, pseudo-inverse, and Frobenius norm by A^T , $\mathcal{R}(A)$, $\text{cols}(A)$, A^\dagger , and $\|A\|_F$, respectively. In addition, if A is a square matrix, A^{-1} denotes its inverse (upon existence), $\text{spec}(A)$ denotes its spectrum (set of eigenvalues), and $\text{sprad}(A) := \max\{|\lambda| \mid \lambda \in \text{spec}(A)\}$ denotes its spectral radius. Moreover, if all eigenvalues of A are real, then $\lambda_{\max}(A)$ represents its largest eigenvalue. The symbols I_m and $\mathbf{0}_{m \times n}$ represent the $m \times m$ identity matrix and $m \times n$ zero matrix. For convenience, we suppress the indices when the context is clear. Given the vector $v \in \mathbb{C}^n$, we use $\|v\|_2$ to denotes its 2-norm. Given the *abstract* vector space \mathcal{V} defined on the field \mathbb{C} , we represent its dimension by $\dim \mathcal{V}$. Note that in this paper, the members of \mathcal{V} can be functions or n -tuples of \mathbb{R} or \mathbb{C} . Given \mathcal{S} a subset of vector space \mathcal{V} , $\text{span}(\mathcal{S})$ represents the vector space comprised of all linear combinations of the elements of \mathcal{S} . Given functions $f : A \rightarrow B$ and $g : B \rightarrow C$, $g \circ f : A \rightarrow C$ denotes their composition. Given the set X equipped with measure μ , the space of square-integrable functions is denoted by $L_2(\mu)$ and is equipped with inner product $\langle f, g \rangle := \int_X f \bar{g} \, d\mu$, where \bar{g} denotes the complex conjugate of function g . Moreover, we denote by $\|\cdot\|_{L_2(\mu)}$ the norm induced by the aforementioned inner product on space L_2 .

II. PRELIMINARIES

In this section, we discuss some preliminary definitions and results regarding the Koopman operator, Extended Dynamic Mode Decomposition method, and the concept of consistency index which we use frequently throughout the paper. The

reader familiar with these concepts and their mathematical properties can safely skip this section.

A. KOOPMAN OPERATOR

Here, we briefly review the Koopman operator associated with a dynamical system and its properties following the terminology in [5]. Consider the following discrete-time system

$$x^+ = T(x), \quad (1)$$

with state space $\mathcal{M} \subseteq \mathbb{R}^n$. Let \mathcal{F} be a vector space (defined on the field \mathbb{C}) comprised of complex-valued functions with domain \mathcal{M} . Moreover, let \mathcal{F} be closed under composition with the map T , i.e., for all $f \in \mathcal{F}$, we have $f \circ T \in \mathcal{F}$. Then, we define the Koopman operator as

$$\mathcal{K}f = f \circ T. \quad (2)$$

One can easily verify that (2) is a linear operator, i.e.,

$$\mathcal{K}(\alpha f + \beta g) = \alpha \mathcal{K}f + \beta \mathcal{K}g, \quad \forall f, g \in \mathcal{F}, \quad \forall \alpha, \beta \in \mathbb{C}. \quad (3)$$

A nonzero function $\phi \in \mathcal{F}$ is an *eigenfunction* of \mathcal{K} with *eigenvalue* $\lambda \in \mathbb{C}$ if

$$\mathcal{K}\phi = \lambda\phi. \quad (4)$$

An important property of the Koopman operator is that the value of eigenfunctions evaluated on system's trajectories behave as solutions of linear difference equations, i.e., given the trajectory $\{x(i)\}_{i=0}^\infty$ of (1), and the eigenfunction (4),

$$\phi(x(i)) = \lambda\phi(x(i-1)), \quad \forall i \in \mathbb{N}. \quad (5)$$

We refer to (5) as *temporal linear evolution* of eigenfunctions. This temporal linearity of eigenfunctions combined with the linearity of the operator on \mathcal{F} defined in (3) enables us to linearly predict function values on system trajectories.

However, one should keep in mind that the space \mathcal{F} is generally infinite-dimensional; therefore, the knowledge of finitely many eigenfunctions (even if they exist) does not necessarily lead to capturing complete information about the system's behavior. Instead, one usually settles on approximations on finite-dimensional subspaces. Before discussing such approximations, we recall the notion of subspace invariance under the Koopman operator, which plays a key role in determining finite-dimensional forms under the Koopman operator. A subspace $\mathcal{G} \subseteq \mathcal{F}$ is *Koopman invariant* if, for all $f \in \mathcal{G}$, we have $\mathcal{K}f \in \mathcal{G}$. Note that the notion of subspace invariance is independent of the choice of basis for the subspace.

B. EXTENDED DYNAMIC MODE DECOMPOSITION

The Koopman operator is generally infinite dimensional and therefore is not directly amenable to implementation on computer hardware. At the same time, we are interested in extracting the system's properties from data. Here we discuss the Extended Dynamic Mode Decomposition

(EDMD) method that seeks to address the issues of infinite-dimensionality and information extraction from data [72]. EDMD uses data to approximate the action of the operator on a given *finite-dimensional* space of functions. To specify the function space, EDMD uses a dictionary comprised of N_d functions from \mathcal{M} to \mathbb{R} . Formally, we define $D : \mathcal{M} \rightarrow \mathbb{R}^{1 \times N_d}$ as¹

$$D(x) = [d_1(x), \dots, d_{N_d}(x)], \quad \forall x \in \mathcal{M},$$

where $d_1, \dots, d_{N_d} \in \mathcal{F}$ are the dictionary elements. To approximate the behavior of the Koopman operator (and therefore the system) on $\text{span}(D)$, EDMD uses data snapshots from the system trajectories in two data matrices $X, Y \in \mathbb{R}^{N \times n}$ such that

$$y_i = T(x_i), \quad \forall i \in \{1, \dots, N\}, \quad (6)$$

where x_i^T and y_i^T are the i th rows of matrices X and Y respectively. For convenience, we define the action of the dictionary on data matrix X (and similarly for any data matrix) as

$$D(X) = [D(x_1)^T, D(x_2)^T, \dots, D(x_n)^T]^T.$$

Note that based on (2) and (6), one can see

$$D(Y) = D \circ T(X) = \mathcal{K}D(X),$$

where $\mathcal{K}D := [\mathcal{K}d_1, \dots, \mathcal{K}d_{N_d}]$. Hence, the dictionary matrices $D(X)$ and $D(Y)$ capture the behavior of the Koopman operator on $\text{span}(D)$. EDMD estimates the action of the operator by solving a least-squares problem

$$\text{minimize}_K \|D(Y) - D(X)K\|_F, \quad (7)$$

which has the following closed-form solution

$$K_{\text{EDMD}} = \text{EDMD}(D, X, Y) := D(X)^\dagger D(Y). \quad (8)$$

Throughout this paper, we make the following assumption.

Assumption 1: (Full Rank Dictionary Matrices): $D(X)$ and $D(Y)$ have full column rank. \square

Note that Assumption 1 implies that the elements of D form a basis for $\text{span}(D)$. Also, it implies that data in X and Y are diverse enough to distinguish between functions in $\text{span}(D)$, i.e., there are not two functions with the same exact evaluation on data matrices. If Assumption 1 holds, K_{EDMD} is the unique solution of (7).

The matrix K_{EDMD} captures important information about the system's behavior and can be used to approximate the action of the Koopman operator on $\text{span}(D)$. Formally, for an arbitrary function $f \in \text{span}(D)$ with the representation $f(\cdot) = D(\cdot)v_f$ for $v_f \in \mathbb{C}^{N_d}$, we can define the EDMD approximation of $\mathcal{K}f$ as

$$\mathfrak{P}_{\mathcal{K}f}(\cdot) = D(\cdot)K_{\text{EDMD}}v_f. \quad (9)$$

¹ $\text{span}(D)$ contains complex-valued functions since the function space is defined on the field \mathbb{C} . However, the functions in the dictionary itself are chosen to be real-valued to simplify the computation without loss of generality.

Remark 1 (EDMD is an Orthogonal Projection): Note that $\mathfrak{P}_{\mathcal{K}_f} \in \text{span}(D)$ is the orthogonal projection (thus the best approximation) of \mathcal{K}_f on $\text{span}(D)$ with respect to the inner product on space $L_2(\mu_X)$ where $\mu_X = \frac{1}{N} \sum_{i=1}^N \delta_{x_i}$ and δ_{x_i} is the Dirac measure defined on x_i (the i th row of data matrix X). We refer the reader to [73] and [95] for more information. \square

Given an eigenvector $v \in \mathbb{C}^{N_d}$ of K_{EDMD} with eigenvalue λ , one can write the EDMD approximation of the function $\mathcal{K}\phi$ with $\phi(\cdot) = D(\cdot)v$ as

$$\mathfrak{P}_{\mathcal{K}\phi}(\cdot) = D(\cdot)K_{\text{EDMD}}v = \lambda D(\cdot)v = \lambda\phi,$$

which has a form similar to definition of Koopman eigenfunctions, cf. (4). In fact, this is not a coincidence, since every Koopman eigenfunction in $\text{span}(D)$ corresponds to an eigenvector of K_{EDMD} .

Lemma 1 (EDMD Captures All Koopman Eigenfunctions in $\text{span}(D)$ [82, Lemma 4.1]): Let $\phi(\cdot) = D(\cdot)v_\phi$ be a Koopman eigenfunction with eigenvalue λ where $v_\phi \in \mathbb{C}^{N_d} \setminus \{0\}$. Then, v_ϕ is an eigenvector of K_{EDMD} , cf. (8), with the same eigenvalue, i.e., $K_{\text{EDMD}}v_\phi = \lambda v_\phi$. \square

Lemma 1 shows that EDMD captures all *exact* eigenfunctions in the span of the dictionary. Lemma 1 is independent of data sampling as long as Assumption 1 holds. In other words, given sufficient amount of data, the sampling method does not matter in identification of *exact* Koopman eigenfunctions.

Remark 2 (Converse of Lemma 1 Does Not Hold): Lemma 1 provides a necessary condition for identification of Koopman eigenfunctions in $\text{span}(D)$. However, this condition is not sufficient. In other words, not every eigenvector of K_{EDMD} corresponds to a Koopman eigenfunction. We refer the reader to our previous work [82, Theorems 4.3 & 4.6] for necessary and (almost surely) sufficient conditions to identify Koopman eigenfunction by applying EDMD *forward and backward in time* and comparing their solutions' eigendecompositions. \square

Next, we show another important property of EDMD regarding the prediction of the Koopman operator's action on the span of the dictionary.

Proposition 1 (The EDMD's Koopman Predictor (9) is Invariant Under Choice of Basis for the Subspace): Let D_1 and D_2 be two dictionaries such that $\text{span}(D_1) = \text{span}(D_2)$ and let Assumption 1 hold for both of them. Given data matrices X, Y , define the EDMD solutions as

$$K_1 = \text{EDMD}(D_1, X, Y), \quad K_2 = \text{EDMD}(D_2, X, Y).$$

Given $f \in \text{span}(D_1) = \text{span}(D_2)$ with representations $f(\cdot) = D_1(\cdot)v_1 = D_2(\cdot)v_2$ for some $v_1, v_2 \in \mathbb{C}^{N_d}$, define the EDMD Koopman predictors using K_1 and K_2 according to (9) as

$$\mathfrak{P}_{\mathcal{K}f,1}(\cdot) = D_1(\cdot)K_1v_1, \quad \mathfrak{P}_{\mathcal{K}f,2}(\cdot) = D_2(\cdot)K_2v_2.$$

Then, $\mathfrak{P}_{\mathcal{K}f,1}(\cdot) = \mathfrak{P}_{\mathcal{K}f,2}(\cdot)$. \square

The proof is provided in the Appendix. Proposition 1 shows that the dynamical information captured by EDMD only depends on the *span of the dictionary* and not the particular

basis/dictionary. Interestingly, Proposition 1 does not require for the dictionary's span to be Koopman-invariant.

We conclude this section by noting that the quality of the predictor (9) depends on the quality of the vector space $\text{span}(D)$. The closer $\text{span}(D)$ is to being invariant under the Koopman operator, the more accurately (9) predicts the effect of the Koopman operator. In the extreme case when $\text{span}(D)$ is Koopman-invariant, then the predictor (9) is exact for all functions in $\text{span}(D)$. Therefore, to find an appropriate subspace (and consequently dictionary) for EDMD, we need to quantify how close a vector space is to being invariant under the Koopman operator.

C. MEASURING KOOPMAN-INVARIANCE PROXIMITY USING THE CONSISTENCY INDEX

Here, we recall the concept of temporal forward-backward consistency following [94]. The first step to find/approximate Koopman-invariant subspaces is to measure how close a subspace is to being Koopman invariant (we refer to this as *invariance proximity*). Since invariance proximity is a property of the vector space (not a particular basis), we need to find a measure that is *invariant with respect to the choice of basis* for the subspace.²

We start by noting that, if $\text{span}(D)$ is Koopman-invariant, under Assumption 1, the solutions of EDMD applied forward in time and backward in time are inverse of each other. This justifies the following definition. For convenience, we define the forward and backward EDMD matrices as follows

$$K_F = \text{EDMD}(D, X, Y) = D(X)^\dagger D(Y),$$

$$K_B = \text{EDMD}(D, Y, X) = D(Y)^\dagger D(X).$$

Definition 1 (Consistency Matrix and Index [94, Definition 1]): Given dictionary D and data matrices X and Y , the consistency matrix is $M_C(D, X, Y) = I - K_F K_B$ and the consistency index is $\mathcal{I}_C(D, X, Y) = \text{sprad}(M_C(D, X, Y))$. \square

We drop the arguments and use M_C and \mathcal{I}_C when the context is clear. The consistency index measures the deviation of K_F and K_B from being the inverse of each other. Interestingly, the spectrum of M_C and consequently \mathcal{I}_C only depend on $\text{span}(D)$ and not the choice of basis/dictionary itself.

Proposition 2 (Consistency Matrix's Spectrum is Invariant under Linear Transformation of Dictionary [94, Proposition 1]): Let D_1 and D_2 be two dictionaries such that $\text{span}(D_1) = \text{span}(D_2)$ and let Assumption 1 hold for them. Then $\text{spec}(M_C(D_1, X, Y)) = \text{spec}(M_C(D_2, X, Y))$. \square

We also recall the following properties of the consistency matrix that we use throughout the paper.

²The residual error of EDMD, $\|D(Y) - D(X)K_{\text{EDMD}}\|_F$, depends on the choice of basis and is not suitable for measuring the invariance proximity. In fact, it is easy to show that if $\text{span}(D)$ is *far from invariant* but contains one exact eigenfunction, then one can find a linear transformation on the dictionary to make the residual error *arbitrarily* close to zero. We refer the reader to [94, Example 1] for an example.

Lemma 2 (Consistency Matrix's Properties [94, Lemma 1]): Given Assumption 1, the consistency matrix $M_C(D, X, Y)$ has the following properties:

- (a) it is similar to a symmetric matrix;
- (b) it is diagonalizable with a complete set of eigenvectors;
- (c) $\text{spec}(M_C(D, X, Y)) \subset [0, 1]$. \square

An important consequence of Lemma 2 is that $\mathcal{I}_C(D, X, Y) = \lambda_{\max}(M_C(D, X, Y))$. Also, under an *appropriate similarity transformation*, one can see \mathcal{I}_C as the induced 2-norm of a positive semidefinite matrix.

The next result shows that the square root of consistency index provides a tight upper bound on the maximum relative root mean square error (RRMSE), which corresponds to the relative L_2 -norm error under the empirical measure of all functions in the $\text{span}(D)$.

Theorem 1 ($\sqrt{\mathcal{I}_C}$ Bounds RRMSE for EDMD's Koopman Predictions [94, Theorem 1]): Given Assumption 1 for dictionary D and data matrices X, Y , define the empirical measure based on data in X as $\mu_X = \frac{1}{N} \sum_{i=1}^N \delta_{x_i}$, where δ_{x_i} is the Dirac measure defined based on the i th row of X . Then

$$\begin{aligned} \text{RRMSE}_{\max} &:= \max_{f \in \text{span}(D)} \frac{\sqrt{\frac{1}{N} \sum_{i=1}^N |\mathcal{K}f(x_i) - \mathfrak{P}_{\mathcal{K}f}(x_i)|^2}}{\sqrt{\frac{1}{N} \sum_{i=1}^N |\mathcal{K}f(x_i)|^2}} \\ &= \max_{f \in \text{span}(D)} \frac{\|\mathcal{K}f - \mathfrak{P}_{\mathcal{K}f}\|_{L_2(\mu_X)}}{\|\mathcal{K}f\|_{L_2(\mu_X)}} \\ &= \sqrt{\mathcal{I}_C(D, X, Y)}, \end{aligned}$$

where x_i is the i th row of X and $\mathfrak{P}_{\mathcal{K}f}$ is defined in (9). \square

Theorem 1 provides a crucial tool for the approximation of Koopman-invariant subspaces. It provides a tight bound on the prediction accuracy of all (uncountably many) functions in $\text{span}(D)$, with a simple closed-form expression to evaluate it. Its derivation through the consistency matrix provides insight into the algebraic structure of the vector space and its connection with the Koopman operator. Beyond the space $L_2(\mu_X)$, we refer the interested reader to [39, Definition 8.5] and [96] for the definition of invariance proximity on general inner-product spaces and its computation using Jordan principal angles.

Remark 3 (Relationship between Invariance Proximity and Eigenspace Residuals): At first glance, one might think that the ratio in the relative error presented in Theorem 1 is a generalization of subspace residuals, see e.g. [88, Equations (3.1) and (3.4)] (also see [85], [86], [87]), to multi-dimensional spaces. However, this is not the case since the denominator of the expressions are different. Invariance proximity and residuals have different roles: the former measures the relative function prediction error of arbitrary functions (not necessarily eigenfunctions) under projected Koopman approximations, while the latter measures the quality of a candidate eigenpair (not arbitrary functions on a vector space). If the Koopman operator is bounded, invariance proximity also bounds the residuals of all candidate eigenpairs derived by orthogonal projection, including the special case of the EDMD method (cf. [96, Lemma 2.3]).

We refer the reader to [96, Section 2.5] for a detailed discussion of this matter over general inner-product spaces. \square

III. MOTIVATING CHALLENGES AND PROBLEM STATEMENT

Our general goal is to develop methods to identify and approximate Koopman-invariant subspaces. We discuss here the challenges that we face in doing so and formally introduce the problem statement. To make our presentation clearer, we use the following system as a running example.

Example 1 (Van der Pol Oscillator): Consider the Van der Pol oscillator

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= (1 - x_1^2)x_2 - x_1, \end{aligned} \quad (10)$$

with state space $\mathcal{M} \subseteq \mathbb{R}^2$. This system is a classical example of a nonlinear oscillator with a stable periodic orbit (limit cycle). Figure 1 shows the vector field and the limit cycle of the Van der Pol oscillator.

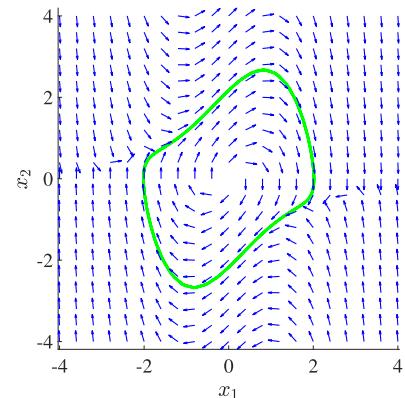


FIGURE 1. Vector field of Van der Pol oscillator (10) and its limit cycle.

We investigate the dependency of EDMD on the choice of dictionary to motivate the importance of developing solid criteria to judiciously choose the library of functions.

Data: We consider a discretization of system (10) with time step $\Delta t = 2.5 \times 10^{-2}$ s. We sample 5000 trajectories with initial conditions uniformly taken from $[-4, 4]^2$. The length of each trajectory is one second and is sampled according to time step Δt leading to $N = 2 \times 10^5$ data snapshots allocated in matrices X, Y , cf. (6).

Dictionaries: We use two dictionaries D_{pol} and D_{tailored} . D_{pol} is comprised of all 45 monomials up to degree 8, while D_{tailored} is comprised of 5 polynomials up to degree 8 tailored to the system.³ Note that $\text{span}(D_{\text{pol}})$ is the vector space of all polynomials with complex coefficients up to degree 8 and that $\text{span}(D_{\text{tailored}}) \subsetneq \text{span}(D_{\text{pol}})$.

³We later discuss how to find such tailored dictionaries. Our aim in this section is to show the importance of tailoring dictionaries compared to generic formulations.

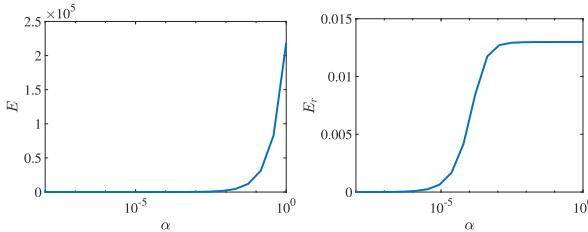


FIGURE 2. EDMD's residual error (left) and its normalized version (right) for $\alpha \in [10^{-8}, 1]$.

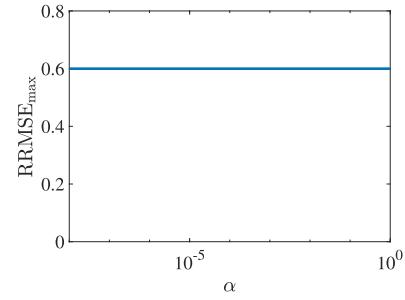


FIGURE 3. The worst-case RRMSE prediction error for all (uncountably many) functions in the vector space $\text{span}(D_\alpha)$ for $\alpha \in [10^{-8}, 1]$.

A. CHALLENGES

Here, we briefly discuss the challenges that we aim to address.

1) A VECTOR SPACE IS NOT JUST A BASIS

The main difficulty to approximate Koopman-invariant subspaces is that one needs to characterize the approximation's quality for the entire vector space which contains *uncountably many* functions. Generally, trying to quantify the EDMD's prediction accuracy by measuring the prediction accuracy of finitely many functions (e.g., the elements of a basis) is not sufficient and can lead to erroneous conclusions.

For instance, given Example 1, consider the parametrized dictionary D_α constructed in the following way: given the dictionary D_{pol} , each non-constant monomial $x_1^{k_1} x_2^{k_2}$ is replaced by $1 + \alpha x_1^{k_1} x_2^{k_2}$, where $\alpha \in \mathbb{R} \setminus \{0\}$ is a parameter. Note that $\text{span}(D_\alpha) = \text{span}(D_{\text{pol}})$ for all nonzero values of α , i.e., D_α 's for $\alpha \in \mathbb{R} \setminus \{0\}$ form different bases for the space of all polynomials up to degree 8. Now, consider the residual error⁴ of EDMD (frequently used as cost function for dictionary learning) and its normalized version

$$E(\alpha) = \|D_\alpha(Y) - D_\alpha(X)K_\alpha\|_F, \quad E_r(\alpha) = \frac{E(\alpha)}{\|D_\alpha(Y)\|_F},$$

where $K_\alpha = \text{EDMD}(D_\alpha, X, Y)$. Figure 2 shows these notions of error with different values of α . Clearly, both errors depend on the choice of basis and do not capture the quality of the vector space. This is despite the fact that the EDMD's Koopman predictor (9) is solely a property of the vector space and is independent of the basis (cf. Proposition 1).

To tackle the aforementioned issues regarding the dependencies on the basis, we use the consistency index, cf. Definition 1, as our invariance-proximity measure which quantifies the worst-case RRMSE of functions on the vector space. Figure 3 shows the $\text{RRMSE}_{\max, \alpha} = \sqrt{\mathcal{I}_C(D_\alpha, X, Y)}$ (cf. Theorem 1) for different values of α . It is clear that the consistency index is a property of data and the vector space $\text{span}(D_\alpha)$ and does not depend on the choice of basis.

Throughout the paper, we use the consistency index as our Koopman-invariance proximity measure and make sure that other computations are also basis-independent, capturing the same information in all bases representations.

⁴Even though this measure is called residual error, it is different from the error used in the residual dynamic mode decomposition in [88].

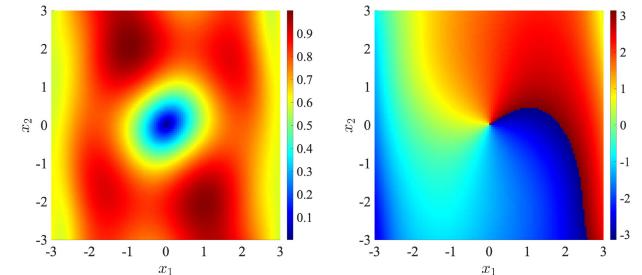


FIGURE 4. Absolute value (left) and phase (right) of the approximated eigenfunction with eigenvalue $\lambda = 0.9992 + 0.0239j$ identified by EDMD applied on D_{pol} .

2) A LARGER DICTIONARY MAY BE LESS ACCURATE

Applying EDMD on an m -dimensional space always leads to m identified eigenfunctions (or generalized eigenfunctions) whether the space contains m actual eigenfunctions or not. This leads to spurious eigenfunctions which should be removed from the set of identified eigenfunctions, see e.g., [88]. Although removing spurious eigenfunctions is an important step regarding the Koopman-based subspace identification, here, we also mention a more subtle problem regarding the inapplicability of Lemma 1 to *approximate* eigenfunctions.

Our aim is to show that simply having a richer subspace by adding functions to the dictionary does not necessarily improve the results. In fact, it can deteriorate the quality of the approximation. Consider Example 1 and note that $\text{span}(D_{\text{tailored}}) \subsetneq \text{span}(D_{\text{pol}})$. By applying EDMD on D_{pol} we find a (complex conjugate) pair of approximate eigenfunctions capturing the periodic behavior of trajectories. Figure 4 shows one of them. Similarly, if we apply EDMD on $\text{span}(D_{\text{tailored}})$ we find a pair of approximate eigenfunctions capturing similar information as Figure 4. Figure 5 shows one eigenfunction of the pair (the other is its complex conjugate).⁵

Both approximate eigenfunctions in Figure 4-5 capture similar information about the periodic behavior of trajectories. However, the accuracy level is different as depicted in Figure 6. Clearly, the quality of approximation for dictionary D_{tailored} is better than D_{pol} .

⁵Both functions in Figures 4-5 are normalized to have maximum absolute value equal to one over the depicted domain.

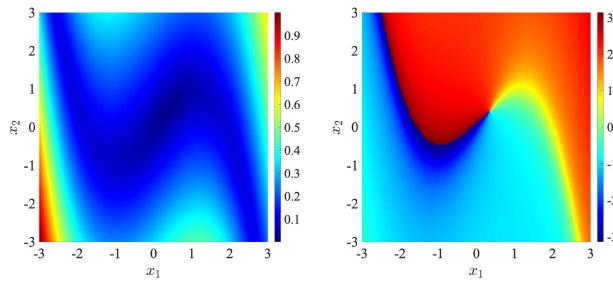


FIGURE 5. Absolute value (left) and phase (right) of the approximated eigenfunction with eigenvalue $\lambda = 0.9887 - 0.0083j$ identified by EDMD applied on $D_{tailored}$.

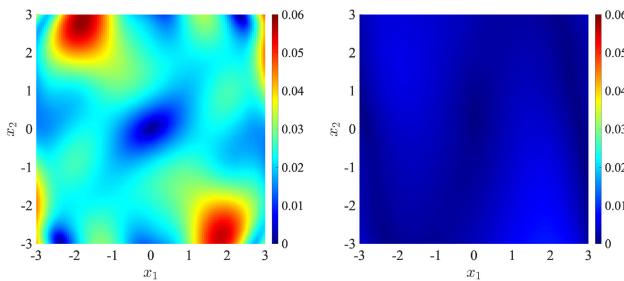


FIGURE 6. The prediction error $|\mathcal{K}\phi(x) - \lambda\phi(x)|$ over the domain $[-3, 3]^2$ for approximated eigenfunctions in Fig. 4 (left) and Fig. 5 (right). Note that both eigenfunctions are normalized to have maximum absolute value equal to one over the represented domain.

Remark 4 (Lemma 1 Does not Apply to Approximations): Even though EDMD can capture *exact* eigenfunctions in the span of any dictionary, cf. Lemma 1, this does not necessarily hold for approximations. In other words, if span of the dictionary contains a highly accurate *approximate* eigenfunction, it might not correspond to an eigenvector of EDMD matrix. Note that the function in Figure 5 also exists in $\text{span}(D_{\text{pol}})$ but is not captured as an eigenfunction if we apply EDMD on D_{pol} . The reason for this discrepancy is that $\text{span}(D_{\text{pol}})$ is far from Koopman-invariant and contains functions with highly inaccurate Koopman predictions whose inaccurate prediction leaks into other functions. \square

Remark 5 (Consistency with the Asymptotic Convergence Results of [73]): Our discussion in this section does not contradict the results regarding the convergence of EDMD in [73]. This is because the latter are of asymptotic nature, considering the case when the subspace's dimension goes to infinity, and do not claim that the prediction error for *all* functions in the subspace decreases *monotonically*. This highlights the need for methods identifying subspaces that satisfy desirable levels of accuracy. \square

3) ALGEBRAIC SEARCH VERSUS OPTIMIZATION

Many methods in the literature rely on optimization-based methods to find Koopman-invariant subspaces. Even though optimization methods are highly practical and accessible, they are also a form of abstraction on the problem. Specifically, by formulating the Koopman-invariant subspace

identification as an optimization problem, one often does not use the linear algebraic structure of the Koopman operator and its eigenfunctions (both spatial and temporal) in the learning process (solving the optimization problem). We view this as a missed opportunity. Especially since the aforementioned optimization problems are generally non-convex and therefore difficult to solve. Hence, we argue that one shall use the algebraic structure of the Koopman operator, not only for modeling and prediction, but also for the learning process itself.

Similarly to optimization-based learning methods that effectively search through a parametric family of functions (e.g., neural-network, polynomial interpolation, etc.), the methods proposed here also search through a family of functions. However, our search space has an algebraic property of being a vector space. This allows us to use the Koopman operator's linearity to design fast and reliable algebraic algorithms leading to convergence and accuracy guarantees on all (uncountably many) functions in the identified vector space.

B. PROBLEM STATEMENT

We are ready to formally pose our problem aimed at addressing the aforementioned challenges. Consider the data snapshot matrices $X, Y \in \mathbb{R}^{N \times n}$ from the (unknown) system (1) (cf. Section II-B). Moreover, consider a finite-dimensional search space of functions \mathcal{S} specified by a basis/dictionary D such that $\mathcal{S} = \text{span}(D)$ for which Assumption 1 holds. We aim to search through \mathcal{S} for a subspace on which the EDMD prediction for all functions reaches a desired accuracy level. We formalize this problem as follows:

Problem 1 (Search for Approximate Koopman-Invariant Subspaces): Given the subspace $\mathcal{S} = \text{span}(D)$ and data matrices X, Y , and any accuracy parameter $\epsilon \in [0, 1]$, we aim to design an iterative algorithm that utilizes the algebraic structure of the Koopman operator and its eigenfunctions (pertaining to linearity in space and temporal evolution) to search through \mathcal{S} and identify the subspace $\mathcal{L}_\epsilon \subseteq \mathcal{S}$ such that

- (a) $\frac{\|\mathcal{K}f - \mathfrak{P}\mathcal{K}f\|_{L_2(\mu_X)}}{\|\mathcal{K}f\|_{L_2(\mu_X)}} \leq \epsilon$ for all $f \in \mathcal{L}_\epsilon$, with $\|\mathcal{K}f\|_{L_2(\mu_X)} \neq 0$ ⁶;
- (b) for all $\epsilon \in [0, 1]$, \mathcal{L}_ϵ always contains all exact Koopman eigenfunctions in \mathcal{S} . \square

Part (a) in Problem 1 ensures that the prediction accuracy of all (uncountably many) functions in the identified space satisfies the bound specified by parameter ϵ while Part (b) ensures that the algorithm retains all the functions with exact prediction.

⁶A function f with $\|\mathcal{K}f\|_{L_2(\mu_X)} = 0$ satisfies $\|\mathcal{K}f - \mathfrak{P}\mathcal{K}f\|_{L_2(\mu_X)} = 0$; therefore, there is no prediction error in this case.

IV. RECURSIVE FORWARD-BACKWARD EDMD ALGORITHM

Here we introduce an algebraic search algorithm based on the use of EDMD forward and backward in time to solve Problem 1. Based on Theorem 1, one can use the consistency index to reformulate Problem 1(a) as finding a space $\mathcal{L}_\epsilon \subseteq \mathcal{S}$ such that

$$\mathcal{I}_C(D_\epsilon, X, Y) \leq \epsilon^2, \quad (11)$$

where D_ϵ is an arbitrary basis for \mathcal{L}_ϵ .

Algorithm 1 presents the Recursive Forward-Backward EDMD (RFB-EDMD) method.⁷

Algorithm 1 Recursive Forward-Backward EDMD

Inputs: $D(X), D(Y) \in \mathbb{R}^{N \times N_d}, \epsilon \in [0, 1]$
Output: $C^{FB} \in \mathbb{R}^{N_d \times N_o}$ $\triangleright N_o$ is the dimension of the identified space

```

1: Procedure RFB-EDMD( $D(X), D(Y), \epsilon$ )
2: Initialization
3:  $i \leftarrow 0, A_0 \leftarrow D(X), B_0 \leftarrow D(Y), C_0 \leftarrow I_{N_d}, V_0 \leftarrow I_{N_d}$ 
4: while  $1$  do
5:    $i \leftarrow i + 1$ 
6:    $K_{F,i} \leftarrow A_{i-1}^\dagger B_{i-1}$   $\triangleright$  Forward EDMD
7:    $K_{B,i} \leftarrow B_{i-1}^\dagger A_{i-1}$   $\triangleright$  Backward EDMD
8:    $M_{C,i} \leftarrow I - K_{F,i} K_{B,i}$   $\triangleright$  Consistency Matrix
9:    $\lambda_{\max,i} = \max\{\lambda \mid M_{C,i}v = \lambda v, v \neq \mathbf{0}\}$ 
10:  if  $\lambda_{\max,i} \leq \epsilon^2$  then
11:    return  $C_i \leftarrow C_{i-1}$   $\triangleright$  The procedure is complete
12:    break
13:  end if
14:   $S_i \leftarrow \text{span}\{v \mid M_{C,i}v = \lambda v, v \neq \mathbf{0}, \lambda < \lambda_{\max,i}\}$ 
         $\triangleright$  Removing eigenvectors with maximum eigenvalue
15:  if  $S_i = \emptyset$  then
16:    return  $C_i \leftarrow \mathbf{0}_{N_d \times 1}$   $\triangleright$  The space does not exist
17:    break
18:  end if
19:   $V_i \leftarrow \text{basis}(S_i)$ 
20:   $C_i \leftarrow C_{i-1} V_i$   $\triangleright$  Reduce the subspace
21:   $A_i \leftarrow A_{i-1} V_i, B_i \leftarrow B_{i-1} V_i$ 
         $\triangleright$  Calculating the new dictionary matrices
22: end while

```

In what follows, we provide a high-level description of the algorithm and then formally characterize its basic properties.

A. INFORMAL DESCRIPTION OF RFB-EDMD

We start by exploring the entire search space $\mathcal{S} = \text{span}(D)$. Given that our goal is to find a subspace for which (11) holds, we first compute the consistency matrix for the original space as $M_C = I - K_F K_B$. There are two possible scenarios:

(a) $\mathcal{I}_C = \lambda_{\max}(M_C) \leq \epsilon^2$: in this case, the search space \mathcal{S} satisfies (11) and the algorithm terminates;

⁷In Algorithm 1, the function $\text{basis}(\mathcal{S})$ returns a matrix whose columns provide an orthonormal basis for the vector space \mathcal{S} .

(b) $\mathcal{I}_C = \lambda_{\max}(M_C) > \epsilon^2$: in this case, we decompose the search space \mathcal{S} as

$$\mathcal{S} = \mathcal{S}_{<\lambda_{\max}} + \mathcal{S}_{=\lambda_{\max}}, \quad (12)$$

where

$$\mathcal{S}_{<\lambda_{\max}} = \text{span}(\{D(\cdot)v \mid v \in \mathcal{V}_{<\lambda_{\max}}\}),$$

$$\mathcal{S}_{=\lambda_{\max}} = \text{span}(\{D(\cdot)v \mid v \in \mathcal{V}_{=\lambda_{\max}}\}),$$

and the sets $\mathcal{V}_{<\lambda_{\max}}$ and $\mathcal{V}_{=\lambda_{\max}}$ are the parts of the eigenspace of M_C corresponding to eigenvalues smaller than λ_{\max} and equal to λ_{\max} , respectively,⁸

$$\mathcal{V}_{<\lambda_{\max}} = \{v \mid M_C v = \lambda v, v \neq 0, \lambda < \lambda_{\max}\},$$

$$\mathcal{V}_{=\lambda_{\max}} = \{v \mid M_C v = \lambda v, v \neq 0, \lambda = \lambda_{\max}\}.$$

Given that the functions in $\mathcal{S}_{=\lambda_{\max}}$ correspond to the maximum eigenvalue of M_C leading to the violation of (11), we remove them from the space and replace \mathcal{S} by $\mathcal{S}_{<\lambda_{\max}}$. Note that there is no direct guarantee for the pruned space $\mathcal{S}_{<\lambda_{\max}}$ to satisfy (11) since all the calculations of consistency index were done based on the original space \mathcal{S} . To tackle this, we recursively prune the subspace by applying the procedure above until the algorithm terminates by Scenario (a).

Noting the fact that one can numerically characterize functions in \mathcal{S} by vectors in $\mathbb{C}^{\dim(\mathcal{S})}$ through the bijective map $f(\cdot) = D(\cdot)v \leftrightarrow v$, we design the RFB-EDMD algorithm on the space of coefficients for computational efficiency (avoiding the complexity of having to deal directly with functions on digital computers). Moreover, given that the system is real-valued and all Koopman eigenfunctions come in complex-conjugate pairs, we restrict the computational operations in Algorithm 1 to real-valued vectors without loss of generality.⁹

After running the algorithm, let C_ϵ^{FB} be the matrix returned by the RFB-EDMD algorithm

$$C_\epsilon^{FB} := \text{RFB-EDMD}(D(X), D(Y), \epsilon). \quad (13)$$

We can characterize the identified space of functions through its basis by

$$D_\epsilon(\cdot) := D(\cdot)C_\epsilon^{FB}. \quad (14)$$

Moreover, similarly to (9), we provide a predictor for \mathcal{K}_f based on the application of EDMD on D_ϵ . Formally, given any arbitrary function $f \in \text{span}(D_\epsilon)$ with description $f(\cdot) = D_\epsilon(\cdot)v$ with $v \in \mathbb{C}^{\# \text{cols}(C_\epsilon^{FB})}$, we define the EDMD predictor for \mathcal{K}_f as

$$\mathfrak{P}_{\mathcal{K}_f, \epsilon} = D_\epsilon(\cdot) K_{\text{EDMD}, \epsilon} v, \quad (15)$$

where $K_{\text{EDMD}, \epsilon} := \text{EDMD}(D_\epsilon, X, Y) = D_\epsilon(X)^\dagger D_\epsilon(Y)$.

⁸Note that the eigenvalues of M_C are real based on Lemma 2(c).

⁹Note that the algorithm still captures all complex-valued objects since the vector spaces are defined over \mathbb{C} . For example, all the complex eigenfunctions can be identified by applying EDMD on the identified space and performing an eigendecomposition (which can lead to complex vectors) on the resulting matrix.

Remark 6: (Implementations on Finite Precision Machines): Although the eigenvalues and eigenvectors of the consistency matrix are real valued (cf. Lemma 2), most iterative algorithms for eigendecomposition lead to complex-valued results in which the imaginary part is negligible (at the level of machine precision). Therefore, we discard the imaginary parts in Step 9 and Step 14 of Algorithm 1. Moreover, to avoid the build-up of round-off errors, we make a change of basis on the dictionary D such that the columns of the matrix $D(X)$ with the new dictionary are orthonormal – the algorithm is not sensitive to change of basis for the search space as a result of Proposition 1. \square

B. BASIC PROPERTIES OF RFB-EDMD

We formally study basic properties of RFB-EDMD algorithm. We first show that the algorithm always terminates in finite iterations and provide an upper bound on its number.

Proposition 3 (Finite-Time Termination of RFB-EDMD): Under Assumption 1, Algorithm 1 terminates in at most N_d iterations, where N_d is the dimension of the search space (equivalently, the dimension of the original dictionary).

Proof: We reason by contradiction. Suppose that Algorithm 1 does not terminate in the first N_d iterations. As a result, it does not execute Step 12 or Step 17 in the first N_d iterations. Consequently, the conditions in Step 10 and Step 15 do not hold and we have $\lambda_{\max,i} > \epsilon^2$ and $S_i \neq \emptyset$ for all $i \in \{1, \dots, N_d\}$. Moreover, the algorithm executes Step 14 and Steps 19-21 in the first N_d iterations.

Based on Step 14, we have $\dim(S_i) < \#\text{cols}(M_{C,i})$. Hence, using the definition of $M_{C,i}$, $K_{F,i}$, $K_{B,i}$, A_i , B_i , C_i , and V_i in Steps 6-8 and Steps 19-21, for all $i \in \{1, \dots, N_d\}$, one can write

$$\dim(S_i) < \#\text{cols}(A_{i-1}) = \#\text{cols}(C_{i-1}) = \#\text{cols}(V_{i-1}). \quad (16)$$

Using the previous equation in conjunction with the fact that $\#\text{cols}(V_j) = \dim(S_j)$ for all $j \in \mathbb{N}$ (cf. Step 19), we have

$$\dim(S_{i+1}) \leq \dim(S_i) - 1, \quad \forall i \in \{1, \dots, N_d\}.$$

Moreover, using the previous inequality N_d times, in conjunction with equation (16) for $i = 1$, and noting that $\#\text{cols}(V_0) = N_d$, one can deduce that

$$\dim(S_{N_d}) < 1.$$

Hence, $S_{N_d} = \emptyset$ (note that the case $S_{N_d} = \{\mathbf{0}\}$ cannot happen since the eigenvectors are always non-zero) and the condition in Step 15 holds, which contradicts the hypothesis that the algorithm does not terminate by executing Step 17 in the first N_d iterations. \blacksquare

The next result provides the basic properties of RFB-EDMD's internal matrices, which we will use frequently in our analysis.

Theorem 2 (Properties of RFB-EDMD's Internal Matrices): Let Algorithm 1 terminate after L iterations. Then, under Assumption 1, for any arbitrary $\epsilon \in [0, 1]$,

(a) $C_i^T C_i = I$ for all $i \in \{0, \dots, L-1\}$;

- (b) $C_L = \mathbf{0}_{N_d \times 1}$ or $C_L^T C_L = I$;
- (c) $\mathcal{R}(C_i) \subseteq \mathcal{R}(C_{i-1})$ for all $i \in \{1, \dots, L\}$;
- (d) $A_i = D(X)C_i$, $B_i = D(Y)C_i$ for all $i \in \{0, \dots, L-1\}$;
- (e) A_i and B_i have full column rank for all $i \in \{0, \dots, L-1\}$.

Proof: (a) The case for $i = 0$ trivially holds (cf. Step 3). Since the algorithm terminates at iteration L , it does not execute Steps 17 in the first $L-1$ iterations. As a result, the condition in Step 15 does not hold in the first $L-1$ iterations and $S_i \neq \emptyset$, $\forall i \in \{1, \dots, L-1\}$. Consequently, based on Step 19, V_i is well defined and using the definition of the basis function,

$$V_i^T V_i = I, \quad \forall i \in \{1, \dots, L-1\}. \quad (17)$$

Moreover, noting the iterative definition of C_i at Step 20 and using the fact that $C_0 = I_{N_d}$, one can write

$$C_i = V_1 V_2 \dots V_i, \quad \forall i \in \{1, \dots, L-1\}. \quad (18)$$

Equations (17)-(18) imply that $C_i^T C_i = I$, for all $i \in \{1, \dots, L-1\}$.

(b) Since the algorithm terminates at iteration L , it either executes Steps 11-12 or Steps 16-17 in the last iteration. In the former case, $C_L = C_{L-1}$, cf. Step 11, and $C_L^T C_L = I$ based on part (a). In the latter case, $C_L = \mathbf{0}_{N_d \times 1}$ trivially holds based on Step 16.

(c) We first prove the result for $i \in \{1, \dots, L-1\}$. The algorithm executes Step 20 in the first $L-1$ iterations since it does not terminate until iteration L . Hence, $C_i = C_{i-1} V_i$ and consequently $\mathcal{R}(C_i) \subseteq \mathcal{R}(C_{i-1})$ for all $i \in \{1, \dots, L-1\}$.

Next, we prove the identity for $i = L$. Since the algorithm terminates at iteration L , it either executes Steps 11-12 leading to $\mathcal{R}(C_L) = \mathcal{R}(C_{L-1})$ or it executes Steps 16-17 leading to $C_L = \mathbf{0}_{N_d \times 1}$. In both cases, the identity $\mathcal{R}(C_L) \subseteq \mathcal{R}(C_{L-1})$ holds.

(d) The result holds trivially for $i = 0$ based on Step 3. Hence, we focus on the cases where $i \neq 0$. Based on the iterative definition of A_i and B_i at Step 21, one can write

$$A_i = A_0 V_1 V_2 \dots V_i,$$

$$B_i = B_0 V_1 V_2 \dots V_i,$$

for all $i \in \{1, \dots, L-1\}$. This, in conjunction with (18) and the fact that $A_0 = D(X)$, $B_0 = D(Y)$ (cf. Step 3), implies that $A_i = D(X)C_i$ and $B_i = D(Y)C_i$ for all $i \in \{1, \dots, L-1\}$.

(e) Using part (d), the statement directly follows from Assumption 1 and the fact that C_i 's have full column rank for all $i \in \{0, \dots, L-1\}$ (cf. part (a)). \blacksquare

Remark 7 (Time Complexity of RFB-EDMD Algorithm): Considering that the time complexity of basic scalar operations are of order $O(1)$, and assuming the generic case where $N \gg N_d$, the most time consuming steps in Algorithm 1 are Steps 6-7, which can be solved using truncated Singular Value Decomposition with complexity $O(N N_d^2)$ (cf. [97]). Since the algorithm terminates after at most N_d iterations (cf. Proposition 3), this yields the total time complexity of $O(N N_d^3)$, which means the algorithm is efficient given the linear complexity in size of data N and the fact that,

in practice, we usually have $N \gg N_d$. It is also worth mentioning that the complexity could be further reduced by using fast linear algebraic methods [98]. \square

V. RFB-EDMD IDENTIFIES KOOPMAN EIGENFUNCTIONS AND APPROXIMATE INVARIANT SUBSPACES

Here we show that the RFB-EDMD algorithm solves Problem 1. The next result shows that the identified subspace by RFB-EDMD satisfies (11); therefore, solving Problem 1(a).

Theorem 3 (RFB-EDMD Bounds the Invariance Proximity and Koopman Prediction Error): *Given the RFB-EDMD output $C_\epsilon^{FB} \neq \mathbf{0}$ and dictionary D_ϵ defined in (13)-(14) with parameter $\epsilon \in [0, 1]$, define the Forward and Backward EDMD matrices on the RFB-EDMD subspace as*

$$\begin{aligned}\tilde{K}_F &= \text{EDMD}(D_\epsilon, X, Y) = D_\epsilon(X)^\dagger D_\epsilon(Y), \\ \tilde{K}_B &= \text{EDMD}(D_\epsilon, Y, X) = D_\epsilon(Y)^\dagger D_\epsilon(X).\end{aligned}$$

Let $M_C = I - \tilde{K}_F \tilde{K}_B$ be the associated consistency matrix and $\mathfrak{P}_{\mathcal{K}_f, \epsilon}$ the EDMD predictor for \mathcal{K}_f defined in (15). Then, $\mathcal{I}_C = \lambda_{\max}(M_C) \leq \epsilon^2$. Consequently,

$$\frac{\|\mathcal{K}f - \mathfrak{P}_{\mathcal{K}_f, \epsilon}\|_{L_2(\mu_X)}}{\|\mathcal{K}f\|_{L_2(\mu_X)}} \leq \epsilon,$$

for all $f \in \mathcal{L}_\epsilon = \text{span}(D_\epsilon)$ with $\|\mathcal{K}f\|_{L_2(\mu_X)} \neq 0$.

Proof: By Proposition 3, the algorithm terminates after a finite number of iterations, say L . There are two ways in which this can happen: either from Steps 11-12 or Steps 16-17. Since $C_\epsilon^{FB} = C_L \neq \mathbf{0}$ by hypothesis, we deduce that the algorithm executes Steps 11-12 upon termination at iteration L . As a result, the condition in Step 10 holds at iteration L and we have $\lambda_{\max, L} \leq \epsilon^2$. Hence, based on the definition of $\lambda_{\max, L}$, one can write

$$\lambda_{\max}(M_{C,L}) = \lambda_{\max}(I - K_{F,L} K_{B,L}) \leq \epsilon^2, \quad (19)$$

where $K_{F,L}$, $K_{B,L}$, and $M_{C,L}$ are the internal matrices of Algorithm 1 at iteration L defined in Steps 6-8. Noting that by definition $C_\epsilon^{FB} = C_L$ and based on Step 11 at iteration L , we have $C_L = C_{L-1}$ leading to the conclusion that $C_\epsilon^{FB} = C_{L-1}$. In addition, one can use Theorem 2(d) and the definition of D_ϵ in (14) to deduce that $A_{L-1} = D_\epsilon(X)$ and $B_{L-1} = D_\epsilon(Y)$. Consequently,

$$\tilde{K}_F = K_{F,L}, \quad \tilde{K}_B = K_{B,L}, \quad M_C = M_{C,L}.$$

Hence, based on the previous equations and (19), we deduce $\mathcal{I}_C = \lambda_{\max}(M_C) \leq \epsilon^2$. The rest of the statement then follows directly from Theorem 1. \blacksquare

The next result shows that at each iteration RFB-EDMD retains all Koopman eigenfunctions in the search space, paving the way for solving Problem 1(b).

Theorem 4 (RFB-EDMD Internally Retains all Koopman Eigenfunctions in Search Space): *Let ϕ be a Koopman eigenfunction with eigenvalue $\lambda \in \mathbb{C} \setminus \{0\}$ contained in the span of the original dictionary, i.e., $\phi \in \mathcal{S} = \text{span}(D)$, denoted as $\phi(\cdot) = D(\cdot)v$ with $v \in \mathbb{C}^{N_d} \setminus \{\mathbf{0}\}$. Under Assumption 1, let L be the termination iteration of*

Algorithm 1 with $\epsilon \in [0, 1]$. Then for all $i \in \{0, \dots, L\}$, there exists a complex vector w_i with appropriate size such that $v = C_i w_i$, where C_i is the internal matrix of Algorithm 1 at iteration i .

Proof: We prove the result by induction. For $i = 0$, $C_0 = I_{N_d}$, cf. Step 3, and $w_0 = v$ trivially satisfies the sought property. Now, suppose that the result holds for $i \in \{0, \dots, L-1\}$, i.e., there exist w_i such that $v = C_i w_i$ for $i \in \{0, \dots, L-1\}$, and we prove the same for $i+1$. If the algorithm at iteration $i+1$ executes Step 11, then we have $w_{i+1} = w_i$ and the sought property holds based on the hypothesis of the induction (note that this only happens at the last iteration $i+1 = L$ following from Steps 11-12). Now, let us consider the case where the algorithm does not execute Step 11 at iteration $i+1$. In this case, the condition in Step 10 does not hold and we have

$$\lambda_{\max, i} > \epsilon^2 \geq 0. \quad (20)$$

By definition of the Koopman eigenfunctions, we have

$$K\phi(x) = \phi \circ T(x) = \lambda\phi(x), \quad \forall x \in \mathcal{M}.$$

This identity combined with the property of data mentioned in (6) and the fact that $\phi(\cdot) = D(\cdot)v$ leads to

$$D(Y)v = \lambda D(X)v.$$

Moreover, based on the hypothesis of the induction, we have $v = C_i w_i$. Therefore,

$$D(Y)C_i w_i = \lambda D(X)C_i w_i.$$

This together with Theorem 2(d) and the fact that $\lambda \neq 0$ leads to

$$\begin{aligned}B_i w_i &= \lambda A_i w_i, \\ A_i w_i &= \lambda^{-1} B_i w_i.\end{aligned} \quad (21)$$

As a result,

$$\begin{aligned}K_{F,i+1} K_{B,i+1} w_i &= A_i^\dagger B_i B_i^\dagger A_i w_i = \lambda^{-1} A_i^\dagger B_i B_i^\dagger B_i w_i \\ &= \lambda^{-1} A_i^\dagger B_i w_i = A_i^\dagger A_i w_i,\end{aligned}$$

where we have used (21), Steps 6-7 of the algorithm and the fact that $B_i B_i^\dagger B_i = B_i$. Using this equation and Step 8, we have

$$M_{C,i+1} w_i = (I - A_i^\dagger A_i) w_i = 0,$$

where the last equality holds since $A_i^\dagger A_i = I$ based on Theorem 2(e). Hence, w_i is an eigenvector of $M_{C,i+1}$ with eigenvalue 0. Therefore, based on (20) and Step 14, we have $w_i \in S_{i+1}$. Consequently, based on Step 19, there exists a vector, which we refer to as w_{i+1} , such that $w_i = V_{i+1} w_{i+1}$. This identity in conjunction with the definition of w_i as $v = C_i w_i$ and Step 20 implies

$$v = C_i w_i = C_i V_{i+1} w_{i+1} = C_{i+1} w_{i+1},$$

leading to $v \in \mathcal{R}(C_{i+1})$, concluding the proof. \blacksquare

Theorem 4 shows that RFB-EDMD iteratively prunes the search space in a way that does not remove critical

information containing exact eigenfunctions. We build on this result to demonstrate that RFB-EDMD solves Problem 1(b).

Theorem 5 (RFB-EDMD Retains all Koopman Eigenfunctions in Search Space): Let $\phi \in \mathcal{S} = \text{span}(D)$ be a Koopman eigenfunction with eigenvalue $\lambda \in \mathbb{C} \setminus \{0\}$ contained in the span of the original dictionary. Under Assumption 1, for $\epsilon \in [0, 1]$, let D_ϵ be the dictionary identified by the RFB-EDMD algorithm in (14). Then $\phi \in \text{span}(D_\epsilon) = \mathcal{L}_\epsilon$.

Proof: By hypothesis, $\phi(\cdot) = D(\cdot)v$ with $v \in \mathbb{C}^{N_d} \setminus \{\mathbf{0}\}$. Let L be the termination iteration of Algorithm 1. By Theorem 4, $v \in \mathcal{R}(C_i)$ for all $i \in \{0, \dots, L\}$. Hence, $C_i \neq \mathbf{0}$ for all $i \in \{0, \dots, L\}$. As a result, the algorithm at iteration L terminates by executing Steps 11-12 (and not Steps 16-17). Hence, by Theorem 4 and equation (13), we have $v \in \mathcal{R}(C_\epsilon^{FB}) = \mathcal{R}(C_L)$. Therefore, from (14), we conclude $\phi \in \text{span}(D_\epsilon)$. ■

VI. ACCURACY HIERARCHY ON SUBSPACES

Here, we show that the solution of RFB-EDMD with respect to the parameter $\epsilon \in [0, 1]$ creates a hierarchy of nested subspaces with different values of invariance proximity. Our first step towards establishing this result is to show the subspaces identified by RFB-EDMD are monotone with respect to the parameter ϵ .

Theorem 6 (Monotonicity of Identified Subspaces by RFB-EDMD with Respect to ϵ): Let $0 \leq \epsilon_1 \leq \epsilon_2 \leq 1$ and

$$\begin{aligned} C_{\epsilon_1}^{FB} &= \text{RFB-EDMD}(D(X), D(Y), \epsilon_1), \\ C_{\epsilon_2}^{FB} &= \text{RFB-EDMD}(D(X), D(Y), \epsilon_2). \end{aligned}$$

Then, $\mathcal{R}(C_{\epsilon_1}^{FB}) \subseteq \mathcal{R}(C_{\epsilon_2}^{FB})$ and $\text{span}(D_{\epsilon_1}) \subseteq \text{span}(D_{\epsilon_2})$, where $D_{\epsilon_1}(\cdot) = D(\cdot)C_{\epsilon_1}^{FB}$ and $D_{\epsilon_2}(\cdot) = D(\cdot)C_{\epsilon_2}^{FB}$ are the corresponding RFB-EDMD dictionaries, cf. (14).

Proof: We start the proof by noting that only Steps 10-12 of Algorithm 1 depend on the parameter ϵ . In fact, as long as the condition in Step 10 does not hold, the parameter ϵ does not play any role in the evolution of Algorithm 1.

Let L_{ϵ_2} be the termination iteration of the algorithm with parameter ϵ_2 (cf. Proposition 3). Upon termination, the algorithm either executes Steps 16-17 (case (a)) or Steps 11-12 (case (b)). We consider these cases below.

Case (a): Since the algorithm with ϵ_2 executes Steps 16-17 at iteration L_{ϵ_2} , the condition in Step 10 never holds through the evolution of the algorithm (otherwise, it would terminate by executing Steps 11-12). Hence, $\lambda_{\max,i} > \epsilon_2^2$ for all $i \in \{1, \dots, L_{\epsilon_2}\}$. As a result, $\lambda_{\max,i} > \epsilon_1^2$ for all $i \in \{1, \dots, L_{\epsilon_2}\}$, and the algorithms with parameters ϵ_1 and ϵ_2 have identical evolution until the end (when they both terminate by executing Step 17) and consequently, $C_{\epsilon_1}^{FB} = C_{\epsilon_2}^{FB} = \mathbf{0}$. Moreover, $\text{span}(D_{\epsilon_1}) = \text{span}(D_{\epsilon_2})$ trivially holds.

Case (b): Since the algorithm with ϵ_2 terminates by executing Steps 11-12, the condition in Step 10 holds at iteration L_{ϵ_2} and consequently, $\lambda_{\max,L_{\epsilon_2}} \leq \epsilon_2^2$. We further

divide this case into two subcases (i) $\lambda_{\max,L_{\epsilon_2}} \leq \epsilon_1^2$ and (ii) $\lambda_{\max,L_{\epsilon_2}} > \epsilon_1^2$.

(i) $\lambda_{\max,L_{\epsilon_2}} \leq \epsilon_1^2$: in this case the algorithm with ϵ_1 also terminates (by executing Steps 11-12) after L_{ϵ_2} iterations with identical output as the algorithm with ϵ_2 . Hence, $\mathcal{R}(C_{\epsilon_1}^{FB}) = \mathcal{R}(C_{\epsilon_2}^{FB})$ and $\text{span}(D_{\epsilon_1}) = \text{span}(D_{\epsilon_2})$.

(ii) $\lambda_{\max,L_{\epsilon_2}} > \epsilon_1^2$: this is the only case when the algorithms with ϵ_1 and ϵ_2 diverge. At iteration L_{ϵ_2} , the algorithm with parameter ϵ_1 will move on to Step 14 and further prune the subspace. Then, the algorithm either terminates at iteration L_{ϵ_2} by executing Steps 16-17 leading to $C_{\epsilon_1}^{FB} = \mathbf{0}$, which satisfies the result trivially, or the algorithm does not execute Steps 16-17 at iteration L_{ϵ_2} and moves on to iteration $L_{\epsilon_2} + 1$. In this case, L_{ϵ_1} , the termination iteration of algorithm with parameter ϵ_1 , satisfies $L_{\epsilon_1} > L_{\epsilon_2}$. Noting that in the first L_{ϵ_2} iterations, the algorithms with parameters ϵ_1 and ϵ_2 have identical evolution (and identical internal matrices) prior to the termination of the algorithm with ϵ_2 , we invoke Theorem 2(c) to conclude that $\mathcal{R}(C_{\epsilon_1}^{FB}) \subseteq \mathcal{R}(C_{\epsilon_2}^{FB})$ and consequently, $\text{span}(D_{\epsilon_1}) \subseteq \text{span}(D_{\epsilon_2})$. ■

Next, we study the behavior of RFB-EDMD with extreme values of $\epsilon \in [0, 1]$, starting with $\epsilon = 1$.

Lemma 3 (RFB-EDMD with $\epsilon = 1$ Identifies the Search Space): Let $C_1^{FB} = \text{FB-EDMD}(D(X), D(Y), 1)$ and $D_1(\cdot) = D(\cdot)C_1^{FB}$. Then, $C_1^{FB} = I_{N_d}$ and $D_1 = D$.

Proof: Based on Lemma 2(c) and the fact that $\epsilon = 1$, one can deduce that at the first iteration of the algorithm the condition in Step 10 holds. Therefore, the algorithm terminates by executing Steps 11-12 and consequently, $C_1^{FB} = C_1 = C_0 = I_{N_d}$. ■

Next, we show that the RFB-EDMD with $\epsilon = 0$ finds a subspace on which the data-driven prediction of the Koopman operator's action on all functions is exact.

Lemma 4 (RFB-EDMD with $\epsilon = 0$ Leads to Subspace with Exact Prediction on Data): Let $C_0^{FB} = \text{FB-EDMD}(D(X), D(Y), 0)$ and $D_0(\cdot) = D(\cdot)C_0^{FB}$. Then, for all $f \in \text{span}(D_0)$ with $\|\mathcal{K}f\| \neq 0$, we have

$$\frac{\|\mathcal{K}f - \mathfrak{P}_{\mathcal{K}f,0}\|_{L_2(\mu_X)}}{\|\mathcal{K}f\|_{L_2(\mu_X)}} = 0,$$

where $\mathfrak{P}_{\mathcal{K}f,0}$ is the EDMD predictor with $\epsilon = 0$ for $\mathcal{K}f$ defined in (15). □

The proof directly follows from Theorem 3.

Remark 8 (Finite-Precision Machines): To use Lemma 4 with finite-precision machines, one should set the value of parameter ϵ to a positive number close to zero instead of exactly equal to zero. To account for round-off errors when the dimension of the search space is large (leading to many iterations), we suggest setting ϵ to be a few orders of magnitude larger than the computer's precision. □

Theorem 6 and Lemmas 3-4 naturally lead to a hierarchy of subspaces based on the choice of parameter $\epsilon \in [0, 1]$.

Remark 9 (Accuracy Hierarchy on Identified Subspaces): Let $\mathcal{L}_\epsilon = \text{span}(D_\epsilon) \subseteq \mathcal{S}$ be the identified subspace by RFB-EDMD given parameter $\epsilon \in [0, 1]$. Then, based on Theorem 6 and Lemmas 3-4, we have $\mathcal{L}_0 \subseteq \mathcal{L}_\epsilon \subseteq \mathcal{L}_1 = \mathcal{S}$

for all $\epsilon \in [0, 1]$. It is crucial to note that there are only finitely many (at most $\dim(S) + 1 = N_d + 1$) distinct subspaces in this hierarchy, all with different dimensions. In other words, if $\mathcal{L}_0 \neq \mathcal{L}_1$, there exist at most m distinct values¹⁰ $0 < \epsilon_1^* < \dots < \epsilon_m^* < 1$ with $m < \dim(S) = N_d$ such that $\mathcal{L}_0 \subsetneq \mathcal{L}_{\epsilon_1^*} \subsetneq \dots \subsetneq \mathcal{L}_{\epsilon_m^*} \subsetneq \mathcal{L}_1$. Therefore, with any value of $\epsilon \in [0, 1]$, RFB-EDMD algorithm identifies the largest subspace from the aforementioned hierarchy whose index (ϵ^*) is smaller than ϵ (cf. Figure 7). Note that based on Theorem 5, \mathcal{L}_0 contains all Koopman eigenfunctions contained in the original search space \mathcal{S} . Moreover, based on Lemma 4, all Koopman predictions on \mathcal{L}_0 are exact on the data. \square

Remark 10 (RFB-EDMD Versus Tunable Symmetric Subspace Decomposition (T-SSD) [84]): Both the RFB-EDMD and T-SSD algorithms, cf. [84], are algebraic searches with guarantees on accuracy and convergence. Moreover, RFB-EDMD has the same order of time complexity as the efficient version of T-SSD algorithm (cf. Remark 7 and [84, Remark 8.2]). However, it has three major differences: (i) while T-SSD relies on custom-made subroutines, RFB-EDMD uses well-established linear algebraic routines, e.g., least-squares, eigendecomposition, etc., for which there exist efficient and computationally robust methods in most software packages. Moreover, all these routines have variations that can directly be used on parallel processing hardware such as GPUs and CPU clusters; (ii) RFB-EDMD has a less aggressive pruning strategy that only cuts the dimension of the subspace by removing the direction along the worst-case errors (corresponding to the largest eigenvalue of the consistency matrix), while T-SSD utilizes a more aggressive pruning strategy that sometimes can over-prune, leading to subspaces with smaller dimension but the same level of accuracy compared to RFB-EDMD; (iii) the subspaces identified by RFB-EDMD enjoy the accuracy hierarchy pointed out in Remark 9 while the pruning in T-SSD can break the monotonicity with respect to the value of ϵ , see [84, Remark 6.7]. \square

VII. SIMULATION RESULTS

We illustrate the correctness and effectiveness of the proposed methods in four different examples. We start by a simple example for which we can find the invariant subspaces analytically to verify the theoretical properties of the RFB-EDMD algorithm.¹¹

A. DISCRETE-TIME NONLINEAR SYSTEM

Consider the two-dimensional nonlinear system defined by the map

$$\begin{aligned} x_1^+ &= 0.8x_1 \\ x_2^+ &= \sqrt{0.9x_2^2 + x_1 + 0.1}, \end{aligned} \quad (22)$$

¹⁰Note that m can be zero, leading to the hierarchy $\mathcal{L}_0 \subsetneq \mathcal{L}_1$.

¹¹The codes for the examples are available at <https://github.com/mhaseli/RFB-EDMD-Codes>

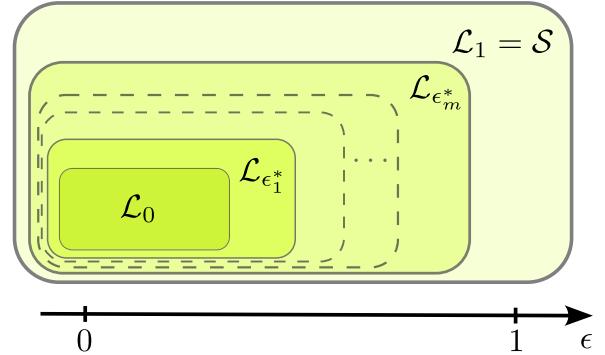


FIGURE 7. The accuracy hierarchy of subspaces contained in the search space \mathcal{S} . Given $\epsilon \in [0, 1]$, RFB-EDMD captures a member of the hierarchy with the largest index smaller than ϵ . Note that \mathcal{L}_0 contains all exact Koopman eigenfunctions contained in the original search space \mathcal{S} (cf. Theorem 5) and \mathcal{L}_1 equals to \mathcal{S} (cf. Lemma 3).

where the state $x = [x_1, x_2]^T$ belongs to $\mathcal{M} = [0, 2]^2$. To apply the RFB-EDMD algorithm, we uniformly sample $N = 1000$ points from the state space \mathcal{M} and push them through the dynamics one step in time to form snapshot matrices $X, Y \in \mathbb{R}^{N \times 2}$. Moreover, we choose a dictionary D spanning the space of all polynomials up to degree two such that the columns of $D(X)$ are orthonormal (cf. Remark 6). This can be done by starting from the dictionary of monomials $[1, x_1, x_2, x_1^2, x_1x_2, x_2^2]$, applying it on data in X , then doing a linear transformation on columns to make them mutually orthonormal.

As a first test for the RFB-EDMD algorithm, we apply it on the dictionary and data for different values of $\epsilon \in [0, 1]$. For comparison purposes, we also apply the efficient version of the Tunable Symmetric Subspace Decomposition (T-SSD) algorithm from [84] on the same dictionary, data, and values of ϵ . Figure 8 shows the dimension of the identified subspaces by RFB-EDMD, $\mathcal{L}_\epsilon = \text{span}(D_\epsilon)$ (where D_ϵ is defined in (14)), and the identified subspaces by the T-SSD algorithm versus the value of the accuracy parameter ϵ . Figure 8 clearly verifies the existence of the accuracy hierarchy on identified subspaces by RFB-EDMD. It also verifies the claim in Remark 10 that T-SSD has a more aggressive pruning. In this example, the accuracy hierarchy identified by RFB-EDMD contains only three nested subspaces with dimensions 6, 5, and 4 respectively. In addition, for $\epsilon = 1$ the algorithm identifies the original six-dimensional search space, which is in agreement with Lemma 3. It is also worth mentioning that both RFB-EDMD and T-SSD algorithms have matching behavior with extreme cases $\epsilon \approx 0$ and $\epsilon = 1$; however, their behavior is different within the interval $(0, 1)$ due to the aggressive pruning of T-SSD.

Given $\epsilon \approx 0$, cf. Remark 8, the RFB-EDMD algorithm identifies the 4-dimensional subspace $\text{span}\{1, x_1, x_1^2, x_2^2\}$. The reader can easily verify the invariance of this space under the Koopman operator.¹² The identified 4-dimensional

¹²It is sufficient to check that the action of the Koopman operator on the basis elements stays in the identified subspace.

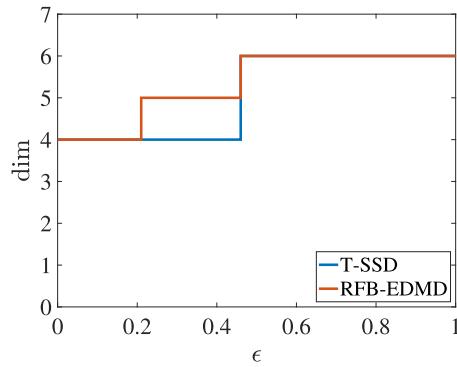


FIGURE 8. Dimension of identified subspaces by RFB-EDMD and T-SSD versus the value of the accuracy parameter $\epsilon \in [0, 1]$ for system (22).

TABLE 1. Identified eigenfunctions and eigenvalues of the Koopman operator associated with system (22).

Eigenfunction	Eigenvalue
$\phi_1(x) = 1$	$\lambda_1 = 1$
$\phi_2(x) = x_1$	$\lambda_2 = 0.8$
$\phi_3(x) = x_1^2$	$\lambda_3 = 0.64$
$\phi_4(x) = 1 - 10x_1 - x_2^2$	$\lambda_4 = 0.9$

subspace contains four Koopman eigenfunctions (up to machine precision) as illustrated in Table 1. Note that according to Theorem 5, the eigenfunctions in Table 1 are all the correct eigenfunctions in the original search space.

Next, to illustrate the accuracy of Koopman-based prediction on the state space, we consider the relative error of function predictions on individual points in the state space as follows

$$E_{\text{rel}}^{\epsilon}(x) = \frac{\|\mathcal{K}D_{\epsilon}(x) - \mathfrak{P}_{\mathcal{K}D_{\epsilon}(x)}\|_2}{\|\mathcal{K}D_{\epsilon}(x)\|_2} \times 100, \quad x \in \mathcal{M}, \quad (23)$$

where $D_{\epsilon}(\cdot) = [d_1^{\epsilon}(\cdot), \dots, d_{\dim \mathcal{L}_{\epsilon}}^{\epsilon}(\cdot)]$ is the identified dictionary by RFB-EDMD (cf. (14)). Moreover, $\mathcal{K}D_{\epsilon}$ and $\mathfrak{P}_{\mathcal{K}D_{\epsilon}}$ are defined in an element-wise manner as

$$\begin{aligned} \mathcal{K}D_{\epsilon} &= [\mathcal{K}d_1^{\epsilon}, \dots, \mathcal{K}d_{\dim \mathcal{L}_{\epsilon}}^{\epsilon}], \\ \mathfrak{P}_{\mathcal{K}D_{\epsilon}} &= [\mathfrak{P}_{\mathcal{K}d_1^{\epsilon}}, \dots, \mathfrak{P}_{\mathcal{K}d_{\dim \mathcal{L}_{\epsilon}}^{\epsilon}}], \end{aligned}$$

where we have used (15). Note that based on (15), one can write $\mathfrak{P}_{\mathcal{K}D_{\epsilon}}(\cdot) = D_{\epsilon}(\cdot)K_{\text{EDMD}, \epsilon}$, where $K_{\text{EDMD}, \epsilon} := \text{EDMD}(D_{\epsilon}, X, Y) = D_{\epsilon}(X)^{\dagger}D_{\epsilon}(Y)$. Figure 9 shows the relative error in (23) over the domain $[0, 2]^2$ for $\epsilon \approx 0$ and $\epsilon = 1$. It is evident that the error for $\epsilon \approx 0$ is equal to zero on the entire state space since the identified subspace is invariant under the Koopman operator. On the other hand, the error for the case with $\epsilon = 1$ (which is equivalent to applying EDMD on the search space according to Lemma 3) reaches 25% on some portions of the state space. It is worth mentioning that the evaluations points in Figure 9 are different from the data used for identification.

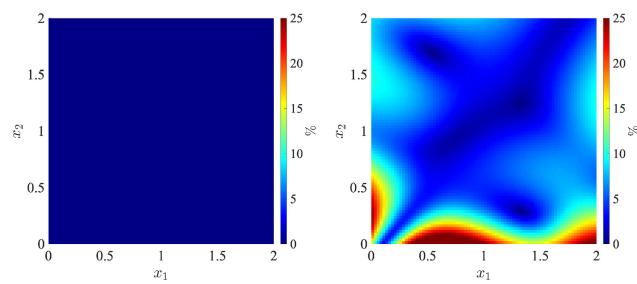


FIGURE 9. Relative error (23) of Koopman-based prediction of dictionaries identified by RFB-EDMD with $\epsilon \approx 0$ (left) and $\epsilon = 1$ (right) for system (22).

B. VAN DER POL OSCILLATOR REVISITED

Here we revisit the Van der Pol oscillator to illustrate the advantages of RFB-EDMD algorithm in addressing the challenges in Section III. To apply the RFB-EDMD algorithm, we use the data described in Example 1. The search space is also the 45-dimensional space of all polynomials up to degree 8. To alleviate the build up of round-off errors, we do a linear transformation on D_{pol} described in Example 1 to build the dictionary D such that the columns of $D(X)$ are orthonormal (cf. Remark 6).

We apply the RFB-EDMD and T-SSD algorithms on the dictionary and data for different values of $\epsilon \in [0, 1]$. Figure 10 shows the dimension of the identified subspaces by RFB-EDMD, $\mathcal{L}_{\epsilon} = \text{span}(D_{\epsilon})$ (where D_{ϵ} is defined in (14)), and the identified subspaces by T-SSD versus the value of the accuracy parameter ϵ . Figure 10 shows the accuracy hierarchy on identified subspaces by RFB-EDMD. Moreover, based on Figure 10, one can easily see that T-SSD has a more aggressive pruning and does not lead to an accuracy hierarchy (nested subspaces with respect to ϵ), since clearly the dimension of the identified subspaces is not monotonic with respect to ϵ . This validates the observations in Remark 10 regarding the differences between RFB-EDMD and T-SSD. For $\epsilon \approx 0$, cf. Remark 8, the identified subspace contains only constant functions, which is a trivial invariant subspace under the Koopman operator and does not provide any dynamical information. Therefore, one is required to allow for some error in the approximation to obtain information about the system's behavior.

To show the prediction accuracy of models built on the identified RFB-EDMD subspaces on the points in state space, we rely on the relative error (23). Figure 11 shows the error on the domain $[-3, 3]^2$ for different values of the accuracy parameter ϵ . It is clear that the prediction accuracy improves as one enforces a tighter invariance proximity by decreasing ϵ .

To show the capabilities of RFB-EDMD in accurately approximating Koopman eigenfunctions, we focus on the case with $\epsilon = 0.02$. In this case, RFB-EDMD identifies a 5-dimensional subspace containing five approximated eigenfunctions. The first eigenfunction $\phi_1(x) \equiv 1$ is a trivial Koopman eigenfunction with eigenvalue $\lambda_1 = 1$.

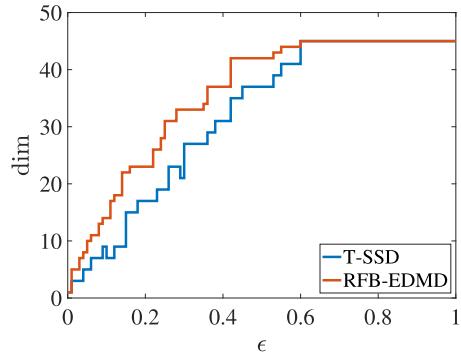


FIGURE 10. Dimension of identified subspaces by RFB-EDMD and T-SSD versus the value of the accuracy parameter $\epsilon \in [0, 1]$ for system (10).

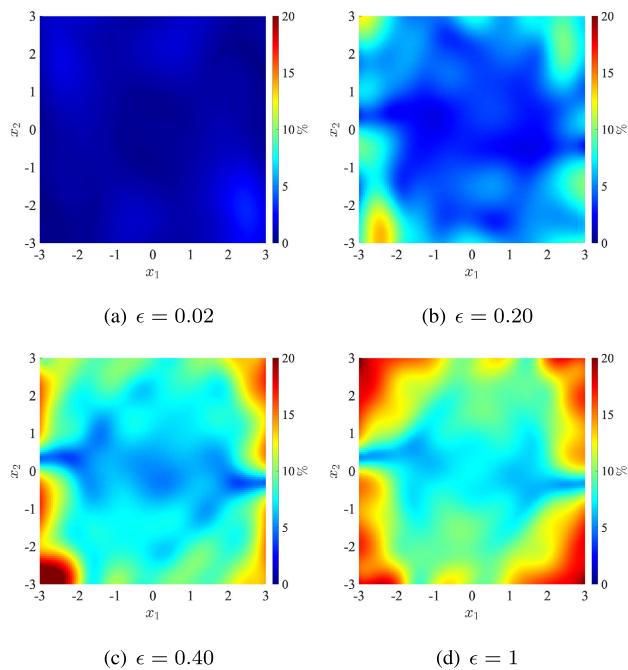


FIGURE 11. Relative error (23) of Koopman-based prediction of dictionaries identified by RFB-EDMD with various values of the accuracy parameter ϵ for system (10).

This is the only exact Koopman eigenfunction in the search space. Additionally, the algorithm approximates two pairs of complex-conjugate eigenfunctions. Figure 5 in Section III shows an eigenfunction from the first pair. Figure 12 shows an eigenfunction from the second pair, which capture information about the attractiveness of the limit cycle (note the parallelism with the phase portrait of the Van der Pol oscillator in Figure 1).

C. DUFFING EQUATION

Consider the Duffing equation (see e.g., [72, Example 4.2])

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= -0.5x_2 + x_1 - x_1^3, \end{aligned} \quad (24)$$

where the state $x = [x_1, x_2]^T$ belongs to $\mathcal{M} = [-2, 2]^2$. We aim to identify Koopman eigenfunctions and invariant subspaces associated to the discretization of the system with

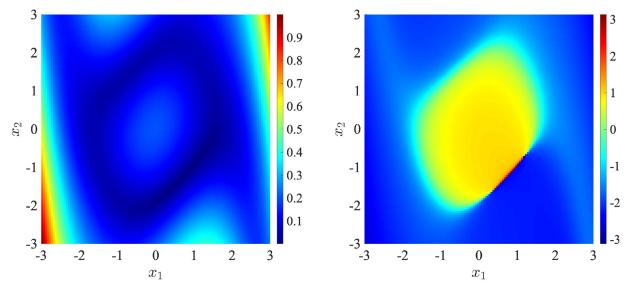


FIGURE 12. Absolute value (left) and phase (right) of the approximated eigenfunction with eigenvalue $\lambda = 0.9858 + 0.0080j$ identified by RFB-EDMD with $\epsilon = 0.02$ for system (10).

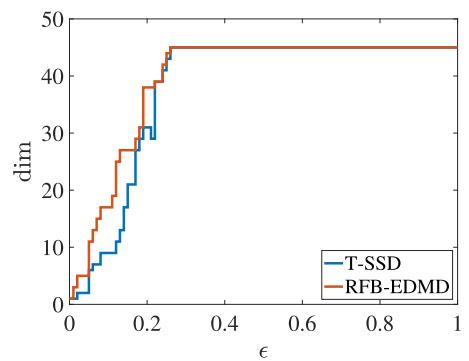


FIGURE 13. Dimension of identified subspaces by RFB-EDMD and T-SSD versus the value of the accuracy parameter $\epsilon \in [0, 1]$ for system (24).

time step $\Delta t = 2.5 \times 10^{-2}s$. To gather data, we simulate the system 5000 times with initial conditions uniformly sampled from state space \mathcal{M} . Each simulation lasts for 1s and the trajectories are sampled with timestep Δt . This procedure results in data snapshot matrices $X, Y \in \mathbb{R}^{N \times 2}$, with $N = 2 \times 10^5$. We use the same search space as the one in Section VII-B, with the dictionary D normalized according to Remark 6 to make the columns of $D(X)$ orthonormal.

We apply the RFB-EDMD algorithm on the dictionary and data for different values of $\epsilon \in [0, 1]$ and compare it with the T-SSD algorithm. Figure 13 shows the dimension of the identified RFB-EDMD subspaces $\mathcal{L}_\epsilon = \text{span}(D_\epsilon)$ (where D_ϵ is defined in (14)) versus the dimension of T-SSD subspaces for different values of the accuracy parameter ϵ . The plot clearly shows the accuracy hierarchy on the identified subspaces by RFB-EDMD and also confirms the advantages of RFB-EDMD over T-SSD as mentioned in Remark 10.

In order to illustrate the prediction accuracy of RFB-EDMD models on individual points in state space \mathcal{M} , we use the relative error (23). Figure 14 shows the relative error for different values of the accuracy parameter ϵ . Clearly, the prediction accuracy improves as one enforces a tighter invariance proximity by decreasing ϵ .

To illustrate the usefulness of RFB-EDMD in approximating eigenfunctions, we focus on the subspace identified with $\epsilon = 0.01$, which has dimension 3. The identified space contains one exact eigenfunction $\phi(x) \equiv 1$, which is a trivial Koopman eigenfunction. The subspace also contains two

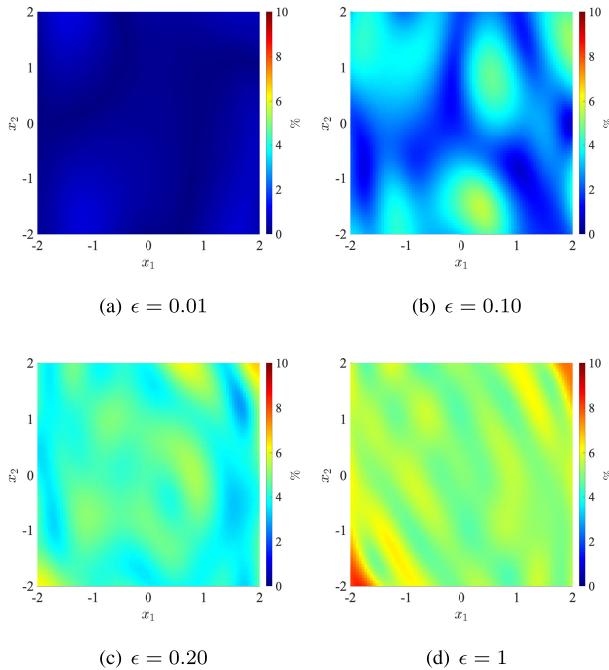


FIGURE 14. Relative error (23) of Koopman-based prediction of dictionaries identified by RFB-EDMD with various values of the accuracy parameter ϵ for system (24).

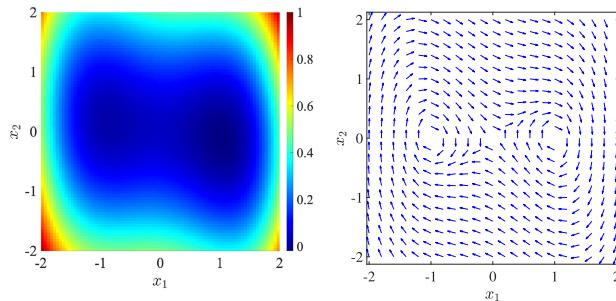


FIGURE 15. Approximated eigenfunction via RFB-EDMD with $\epsilon = 0.01$ with eigenvalue $\lambda = 0.9833$ (left) associated with system (24) along with corresponding vector field (right).

real-valued approximated eigenfunctions. Figure 15 shows an approximated eigenfunction along with the system's phase portrait to show how it captures the behavior of the vector field. Moreover, since the eigenvalue is inside the unit circle, by the temporal linearity of eigenfunctions (5), one can deduce that the system trajectories converge to the zero-level sets of the eigenfunction, which is in agreement with the fact that the system has attractive equilibria at $(-1, 0)$ and $(1, 0)$.

D. YEAST GLYCOLYSIS

Here, we turn our attention to a non-planar nonlinear system with a 7-dimensional state space describing yeast glycolysis [99], [100], [101],

$$\begin{aligned}\dot{x}_1 &= 2.5 - 100 \frac{x_1 x_6}{1 + (\frac{x_6}{0.52})^4}, \\ \dot{x}_2 &= 200 \frac{x_1 x_6}{1 + (\frac{x_6}{0.52})^4} - 6x_2(1 - x_5) - 12x_2x_5,\end{aligned}$$

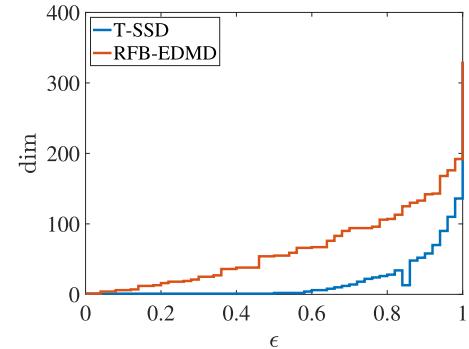


FIGURE 16. Dimension of identified subspaces by RFB-EDMD and T-SSD versus the value of the accuracy parameter $\epsilon \in [0, 1]$ for system (25).

$$\begin{aligned}\dot{x}_3 &= 6x_2(1 - x_5) - 16x_3(4 - x_6), \\ \dot{x}_4 &= 16x_3(4 - x_6) - 100x_4x_5 - 13(x_4 - x_7), \\ \dot{x}_5 &= 6x_2(1 - x_5) - 100x_4x_5 - 12x_2x_5, \\ \dot{x}_6 &= -200 \frac{x_1 x_6}{1 + (\frac{x_6}{0.52})^4} + 32x_3(4 - x_6) - 1.28x_6, \\ \dot{x}_7 &= 1.3(x_4 - x_7) - 1.8x_7,\end{aligned}\quad (25)$$

where $x = [x_1, \dots, x_7]^T$ is the state vector. These correspond to the equations and parameter values from [100]. Our aim is to use RFB-EDMD to approximate Koopman invariant subspaces associated with the system's discretization with timestep $\Delta t = 0.05$ s. To gather data, we simulate the system from 1000 initial conditions uniformly selected from $[0, 1]^7$. We run each simulation for a duration of 3s and sample each trajectory based on timestep Δt and form data snapshot matrices $X, Y \in \mathbb{R}^{N \times 7}$ with $N = 6 \times 10^4$. In addition, we consider the space of all polynomials up to degree 4 over the state space. This space has dimension 330. We use a dictionary D to span this subspace such that the columns of $D(X)$ are orthonormal, cf. Remark 6.

We apply the RFB-EDMD and T-SSD algorithms with different values of $\epsilon \in [0, 1]$. Figure 16 shows the accuracy hierarchy on the subspaces identified by RFB-EDMD (cf. Remark 9) while T-SSD does not lead to an accuracy hierarchy. Moreover, T-SSD significantly overprunes the subspace confirming the statements in Remark 10. Clearly, enforcing a higher level of accuracy by choosing a smaller value for ϵ leads to smaller subspaces for both algorithms.

Unlike in the previous 2-dimensional examples for which we could plot the errors on the state space, in the current example we rely on statistical plots due to the higher dimensional nature of the system. To this end, we generate a test data set, by uniformly choosing 10^4 data points in the space $[0, 1]^7$, and apply the relative error (23) of Koopman-based predictors on identified subspaces by RFB-EDMD given different values of $\epsilon \in [0, 1]$. Figure 17 shows the histogram plots of errors over the test data set. For $\epsilon \approx 0$, cf. Remark 8, the algorithm identifies the maximal Koopman-invariant subspace in the search space, which has dimension one and contains all constant functions, therefore

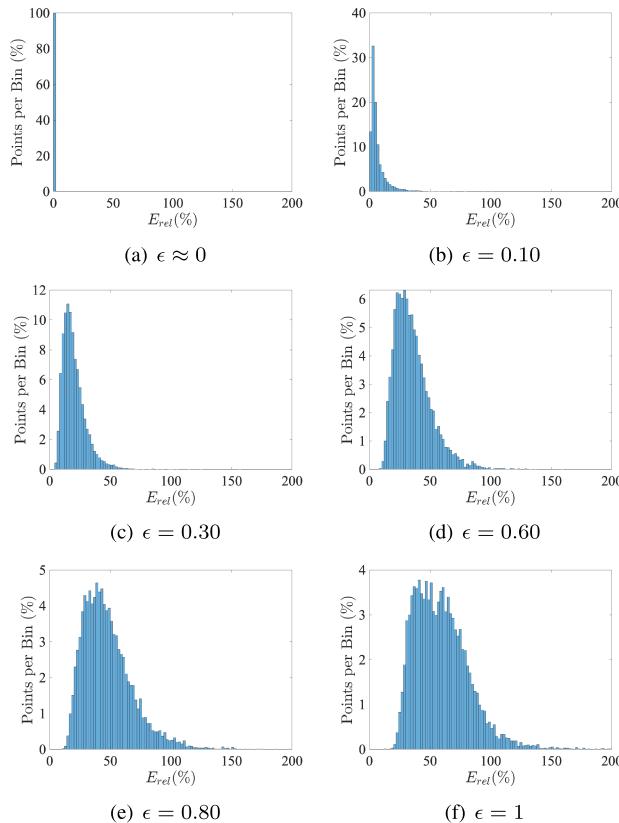


FIGURE 17. Histogram plot of relative error (23) of Koopman-based predictor over 10^4 points randomly selected from $[0, 1]^7$ associated with dictionaries identified with various values of the accuracy parameter ϵ for system (25).

the prediction on this space is exact. By increasing the value of ϵ , the dimension of the identified space increases while the prediction accuracy decreases, consistent with the accuracy hierarchy on identified subspaces. It should be noted that unlike the consistency index (cf. Theorem 1) which bounds the relative error in *function norms*, the error in (23) measures the error on *individual points*. Therefore, one can see in Figure 17 that the error in (23) can go above 100% while the consistency index is always in $[0, 1]$ (cf. Lemma 2).

VIII. CONCLUDING REMARKS AND OUTLOOK

We have introduced RFB-EDMD, an iterative algorithm that searches through a vector space of functions to find subspaces that are approximately invariant under the Koopman operator associated with a dynamical system. One can tune the accuracy of the approximation by choosing an accuracy parameter. We have provided a complete convergence and accuracy analysis for the RFB-EDMD algorithm and proved that it always captures correct Koopman eigenfunctions within the search space. We next discuss a few important questions that need to be addressed in the future.

A. CHOOSING THE SEARCH SPACE

The RFB-EDMD algorithm can search through any finite-dimensional vector space of functions as long as Assumption 1 holds. However, the identified subspaces

depend on the original search space. Naturally, the richer the search space, the more information one can extract. An important challenge is how to choose the search space based on information about the system to make sure that the space is rich enough to capture necessary information while not leading to large round-off errors as a result of ill-conditioned matrices often associated with some families of functions (e.g., polynomials).

B. CAPTURING THE SYSTEM'S FULL STATE

To completely predict the system's behavior, one needs to capture the full state of the system. In the RFB-EDMD algorithm, the accuracy level can be chosen such that the state observables are within the identified subspace or are close to it akin to the optimization method used in [87]. It is important to note that such finite-dimensional approximations have their own theoretical limitations [69] and often come at the cost of approximation errors that might accumulate in long-term predictions. Therefore, it is imperative to study the behavior of the models in multi-step prediction. Such analysis requires stronger assumptions about the underlying system, as compared to the mild rank condition, cf. Assumption 1, used here.

C. LEARNING THE BEST COORDINATES FOR REPRESENTATION

State is a fundamental system property and does not depend on a fixed coordinate. Generally, the choice of coordinates for representing the system's state is based on ease of representation, geometric/physical properties (position, momentum, etc), or the choice of sensors used for measurement. However, the aforementioned reasons do not apply to Koopman-based methods which work with function evolutions instead of directly describing the system's trajectories. This leads to the important question of how to choose/learn the appropriate coordinate system to capture the system behavior on low-dimensional vector spaces of functions. Although, previous research (e.g., [102]) provides a partial answer to this question for systems with specific attractors, to the best of our knowledge there is no definite *algorithmic* solution to choose appropriate coordinates for finite-dimensional Koopman-based approximations for *general* nonlinear dynamical systems.

D. FOCUSING ON FUNDAMENTAL SYSTEM PROPERTIES INSTEAD OF FULL STATE REPRESENTATION

As we mentioned above, there are limitations associated with finite-dimensional Koopman-based models that capture the system's full state. Interestingly, the full knowledge of the system is not required for many applications. For example, one might be interested in finding conservation laws and geometric constraints, which correspond to Koopman eigenfunctions with eigenvalue equal to one and do not require complete reconstruction of the system's state. Another important example in systems and control theory is

deciding the stability of attractors. Generally, one can address this problem through finding appropriate eigenfunctions [27] or building Lyapunov functions based on Koopman-operator theory [29], which again do not require reconstruction of the system's state.

ACKNOWLEDGMENT

The authors would like to thank William Sharpless for pointing out the yeast glycolysis system used in Section VII-D.

APPENDIX

Here, we provide the proof of Proposition 1.

Proof A1 (Proposition 1): For convenience, let $\mathcal{S} = \text{span}(D_1) = \text{span}(D_2)$. Note that as a result of Assumption 1, one can guarantee that the elements of D_1 and D_2 are linearly independent. Thus, D_1 and D_2 form bases for \mathcal{S} . Hence, there exists an invertible matrix $R \in \mathbb{R}^{N_d \times N_d}$ such that $D_1(\cdot) = D_2(\cdot)R$. Therefore, based on the definition of v_1 and v_2 , one can write

$$Rv_1 = v_2. \quad (26)$$

Moreover, based on [84, Lemma 7.1], we have

$$K_1 = R^{-1}K_2R. \quad (27)$$

As a result of the relationship between the dictionaries, and equations (26)-(27), one can write

$$\begin{aligned} \mathfrak{P}_{\mathcal{K}f,1}(\cdot) &= D_1(\cdot)K_1v_1 = (D_2(\cdot)R)(R^{-1}K_2R)(R^{-1}v_2) \\ &= D_2(\cdot)K_2v_2 = \mathfrak{P}_{\mathcal{K}f,2}(\cdot), \end{aligned}$$

concluding the proof. \square

REFERENCES

- [1] B. O. Koopman, "Hamiltonian systems and transformation in Hilbert space," *Proc. Nat. Acad. Sci. USA*, vol. 17, no. 5, pp. 315–318, May 1931.
- [2] B. O. Koopman and J. V. Neumann, "Dynamical systems of continuous spectra," *Proc. Nat. Acad. Sci. USA*, vol. 18, no. 3, pp. 255–263, Mar. 1932.
- [3] I. Mezić, "Spectral properties of dynamical systems, model reduction and decompositions," *Nonlinear Dyn.*, vol. 41, nos. 1–3, pp. 309–325, Aug. 2005.
- [4] I. Mezić, "Spectrum of the Koopman operator, spectral expansions in functional spaces, and state-space geometry," *J. Nonlinear Sci.*, vol. 30, no. 5, pp. 2091–2145, Dec. 2019.
- [5] M. Budišić, R. Mohr, and I. Mezić, "Applied koopmanism," *Chaos*, vol. 22, no. 4, Dec. 2012, Art. no. 047510.
- [6] S. Peitz and S. Klus, "Koopman operator-based model reduction for switched-system control of PDEs," *Automatica*, vol. 106, pp. 184–191, Aug. 2019.
- [7] W. Zhang, Y.-C. Yu, and J.-S. Li, "Dynamics reconstruction and classification via Koopman features," *Data Mining Knowl. Discovery*, vol. 33, no. 6, pp. 1710–1735, Nov. 2019.
- [8] S. Klus, F. Nüske, S. Peitz, J.-H. Niemann, C. Clementi, and C. Schütte, "Data-driven approximation of the Koopman generator: Model reduction, system identification, and control," *Phys. D, Nonlinear Phenomena*, vol. 406, May 2020, Art. no. 132416.
- [9] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, "Spectral analysis of nonlinear flows," *J. Fluid Mech.*, vol. 641, pp. 115–127, Dec. 2009.
- [10] I. Mezić, "Analysis of fluid flows via spectral properties of the Koopman operator," *Annu. Rev. Fluid Mech.*, vol. 45, no. 1, pp. 357–378, Jan. 2013.
- [11] A. Dotto, D. Lengani, D. Simoni, and A. Taccchella, "Dynamic mode decomposition and Koopman spectral analysis of boundary layer separation-induced transition," *Phys. Fluids*, vol. 33, no. 10, Oct. 2021, Art. no. 104104.
- [12] A. Hasnain, N. Boddupalli, and E. Yeung, "Optimal reporter placement in sparsely measured genetic networks using the Koopman operator," in *Proc. IEEE 58th Conf. Decis. Control (CDC)*, Nice, France, Dec. 2019, pp. 19–24.
- [13] J. Harrison and E. Yeung, "Stability analysis of parameter varying genetic toggle switches using Koopman operators," *Mathematics*, vol. 9, no. 23, p. 3133, Dec. 2021.
- [14] S. Qian and C.-A. Chou, "A Koopman-operator-theoretical approach for anomaly recognition and detection of multi-variate EEG system," *Biomed. Signal Process. Control*, vol. 69, Aug. 2021, Art. no. 102911.
- [15] I. K. Gallos, D. Lehmburg, F. Dietrich, and C. Siettos, "Data-driven modelling of brain activity using neural networks, diffusion maps, and the Koopman operator," *Chaos: Interdiscipl. J. Nonlinear Sci.*, vol. 34, no. 1, Jan. 2024, Art. no. 013151.
- [16] Y. Susuki, I. Mezić, F. Raak, and T. Hikihara, "Applied Koopman operator theory for power systems technology," *Nonlinear Theory Its Appl., IEICE*, vol. 7, no. 4, pp. 430–459, 2016.
- [17] M. Netto and L. Mili, "A robust data-driven Koopman Kalman filter for power systems dynamic state estimation," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 7228–7237, Nov. 2018.
- [18] S. P. Nandamoori, S. Guan, S. Kundu, S. Pal, K. Agarwal, Y. Wu, and S. Choudhury, "Graph neural network and Koopman models for learning networked dynamics: A comparative study on power grid transients prediction," *IEEE Access*, vol. 10, pp. 32337–32349, 2022.
- [19] A. M. Avila and I. Mezić, "Data-driven analysis and forecasting of highway traffic dynamics," *Nature Commun.*, vol. 11, no. 1, p. 2090, Apr. 2020.
- [20] E. Ling, L. Zheng, L. J. Ratliff, and S. Coogan, "Koopman operator applications in signalized traffic systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 4, pp. 3214–3225, Apr. 2022.
- [21] S. Das, S. Mustavee, S. Agarwal, and S. Hasan, "Koopman-theoretic modeling of quasiperiodically driven systems: Example of signalized traffic corridor," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 53, no. 7, pp. 4466–4476, Jul. 2023.
- [22] R. Dubey, S. R. Samantaray, B. K. Panigrahi, and V. G. Venkoparao, "Koopman analysis based wide-area back-up protection and faulted line identification for series-compensated power network," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2634–2644, Sep. 2018.
- [23] M. Georgescu, S. Loire, D. Kasper, and I. Mezić, "Whole-building fault detection: A scalable approach using spectral methods," 2017, *arXiv:1703.07048*.
- [24] M. Bakhtiaridoust, M. Yadegar, N. Meskin, and M. Noorizadeh, "Model-free geometric fault detection and isolation for nonlinear systems using Koopman operator," *IEEE Access*, vol. 10, pp. 14835–14845, 2022.
- [25] S. Schlor, R. Strässer, and F. Allgöwer, "Koopman interpretation and analysis of a public-key cryptosystem: Diffie-hellman key exchange," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 984–990, 2023.
- [26] Z. Zeng, J. Liu, and Y. Yuan, "A generalized nyquist-Shannon sampling theorem using the Koopman operator," 2023, *arXiv:2303.01927*.
- [27] A. Mauroy and I. Mezić, "Global stability analysis using the eigenfunctions of the Koopman operator," *IEEE Trans. Autom. Control*, vol. 61, no. 11, pp. 3356–3369, Nov. 2016.
- [28] B. Yi and I. R. Manchester, "On the equivalence of contraction and Koopman approaches for nonlinear stability and control," *IEEE Trans. Autom. Control*, vol. 69, no. 7, pp. 4336–4351, Jul. 2024.
- [29] J. J. Bramburger and G. Fantuzzi, "Auxiliary functions as Koopman observables: Data-driven analysis of dynamical systems via polynomial optimization," *J. Nonlinear Sci.*, vol. 34, no. 1, p. 8, Feb. 2024.
- [30] S. A. Deka, A. M. Valle, and C. J. Tomlin, "Koopman-based neural Lyapunov functions for general attractors," in *Proc. IEEE Conf. Decision Control*, Cancun, Mexico, Dec. 2022, pp. 5123–5128.
- [31] S. A. Deka and D. V. Dimarogonas, "Supervised learning of Lyapunov functions using Laplace averages of approximate Koopman eigenfunctions," *IEEE Control Syst. Lett.*, vol. 7, pp. 3072–3077, 2023.
- [32] C. Mugisho Zagabe and A. Mauroy, "Uniform global stability of switched nonlinear systems in the Koopman operator framework," 2023, *arXiv:2301.05529*.
- [33] L. Zheng, X. Liu, Y. Xu, W. Hu, and C. Liu, "Data-driven estimation for region of attraction for transient stability using Koopman operator," *CSEE J. Power Energy Syst.*, vol. 9, no. 4, pp. 1405–1413, Jul. 2022.

[34] C. Garcia-Tenorio, D. Tellez-Castro, E. Mojica-Nava, and A. Vande Wouwer, "Evaluation of the regions of attraction of higher-dimensional hyperbolic systems using extended dynamic mode decomposition," *Automatica*, vol. 4, no. 1, pp. 57–77, Feb. 2023.

[35] A. Mauroy and A. Sootla, "Estimation of regions of attraction with formal certificates in a purely data-driven setting," in *Proc. 62nd IEEE Conf. Decis. Control (CDC)*, Singapore, Dec. 2023, pp. 4682–4687.

[36] Y. Meng, R. Zhou, and J. Liu, "Learning regions of attraction in unknown dynamical systems via zubov-koopman lifting: Regularities and convergence," 2023, *arXiv:2311.15119*.

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

[38] M. Blischke and J. P. Hespanha, "Learning switched Koopman models for control of entity-based systems," in *Proc. 62nd IEEE Conf. Decis. Control (CDC)*, Singapore, Dec. 2023, pp. 6006–6013.

[39] M. Haseli and J. Cortés, "Modeling nonlinear control systems via Koopman control family: Universal forms and subspace invariance proximity," 2023, *arXiv:2307.15368*.

[40] D. Gadginmath, V. Krishnan, and F. Pasqualetti, "Data-driven feedback linearization using the Koopman generator," 2022, *arXiv:2210.05046*.

[41] V. Zinage and E. Bakolas, "Neural Koopman Lyapunov control," *Neurocomputing*, vol. 527, pp. 174–183, Mar. 2023.

[42] B. Huang and U. Vaidya, "A convex approach to data-driven optimal control via Perron–Frobenius and Koopman operators," *IEEE Trans. Autom. Control*, vol. 67, no. 9, pp. 4778–4785, Sep. 2022.

[43] A. Sootla, A. Mauroy, and D. Ernst, "Optimal control formulation of pulse-based control using Koopman operator," *Automatica*, vol. 91, pp. 217–224, May 2018.

[44] M. E. Villanueva, C. N. Jones, and B. Houska, "Towards global optimal control via Koopman lifts," *Automatica*, vol. 132, Oct. 2021, Art. no. 109610.

[45] J. Hespanha and K. Camsari, "Markov chain Monte Carlo for koopman-based optimal control: Technical report," 2024, *arXiv:2405.01788*.

[46] R. Strässer, J. Berberich, and F. Allgöwer, "Robust data-driven control for nonlinear systems using the Koopman operator*," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 2257–2262, 2023.

[47] J. Deutscher, "Data-driven control of linear parabolic systems using Koopman eigenstructure assignment," 2024, *arXiv:2407.00432*.

[48] G. Mamakoukas, M. L. Castaño, X. Tan, and T. D. Murphey, "Derivative-based Koopman operators for real-time control of robotic systems," *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 2173–2192, Dec. 2021.

[49] L. Shi and K. Karydis, "Enhancement for robustness of Koopman operator-based data-driven mobile robotic systems," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 2503–2510.

[50] G. Mamakoukas, I. Abraham, and T. D. Murphey, "Learning stable models for prediction and control," *IEEE Trans. Robot.*, vol. 39, no. 3, pp. 2255–2275, Mar. 2023.

[51] H. H. Asada, "Global, unified representation of heterogenous robot dynamics using composition operators: A Koopman direct encoding method," *IEEE/ASME Trans. Mechatronics*, vol. 28, no. 5, pp. 2633–2644, May 2023.

[52] D. Bruder, X. Fu, R. B. Gillespie, C. D. Remy, and R. Vasudevan, "Data-driven control of soft robots using Koopman operator theory," *IEEE Trans. Robot.*, vol. 37, no. 3, pp. 948–961, Jun. 2021.

[53] D. A. Haggerty, M. J. Banks, E. Kamenar, A. B. Cao, P. C. Curtis, I. Mezić, and E. W. Hawkes, "Control of soft robots with inertial dynamics," *Sci. Robot.*, vol. 8, no. 81, Aug. 2023, Art. no. eadd6864.

[54] S. Xie and J. Ren, "Linearization of recurrent-neural-network-based models for predictive control of nano-positioning systems using data-driven Koopman operators," *IEEE Access*, vol. 8, pp. 147077–147088, 2020.

[55] M. Rahmani and S. Redkar, "Optimal control of a MEMS gyroscope based on the Koopman theory," *Int. J. Dyn. Control*, vol. 11, no. 5, pp. 2256–2264, Oct. 2023.

[56] C. Rodwell and P. Tallapragada, "Embodied hydrodynamic sensing and estimation using Koopman modes in an underwater environment," in *Proc. Amer. Control Conf. (ACC)*, Atlanta, Georgia, Jun. 2022, pp. 1632–1637.

[57] M. Mitjans, D. M. Levine, L. N. Awad, and R. Tron, "Koopman pose predictions for temporally consistent human walking estimations," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Kyoto, Japan, Oct. 2022, pp. 5257–5264.

[58] C. Folkestad, S. X. Wei, and J. W. Burdick, "KoopNet: Joint learning of Koopman bilinear models and function dictionaries with application to quadrotor trajectory tracking," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 1344–1350.

[59] W. A. Manzoor, S. Rawashdeh, and A. Mohammadi, "Koopman operator-based data-driven identification of tethered subsatellite deployment dynamics," *J. Aerosp. Eng.*, vol. 36, no. 4, Jul. 2023, Art. no. 04023021.

[60] C. Folkestad, Y. Chen, A. D. Ames, and J. W. Burdick, "Data-driven safety-critical control: Synthesizing control barrier functions with Koopman operators," *IEEE Control Syst. Lett.*, vol. 5, no. 6, pp. 2012–2017, Dec. 2021.

[61] M. Black and D. Panagou, "Safe control design for unknown nonlinear systems with koopman-based fixed-time identification," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 11369–11376, 2023.

[62] M. Mazouchi and H. Modares, "Data-driven safe control via finite-time Koopman identifier," *Int. J. Control.*, vol. 97, no. 10, pp. 2284–2297, Oct. 2024.

[63] S. Bak, S. Bogomolov, P. S. Duggirala, A. R. Gerlach, and K. Potomkin, "Reachability of black-box nonlinear systems after Koopman operator linearization," *IFAC-PapersOnLine*, vol. 54, no. 5, pp. 253–258, 2021.

[64] B. Umathé, D. Tellez-Castro, and U. Vaidya, "Reachability analysis using spectrum of Koopman operator," *IEEE Control Syst. Lett.*, vol. 7, pp. 595–600, 2023.

[65] O. Thapliyal and I. Hwang, "Approximate reachability for Koopman systems using mixed monotonicity," *IEEE Access*, vol. 10, pp. 84754–84760, 2022.

[66] W. Sharpless, N. Shinde, M. Kim, Y. Tin Chow, and S. Herbert, "Koopman-hop hamilton-jacobi reachability and control," 2023, *arXiv:2303.11590*.

[67] H. Balmi, A. Aspeel, Z. Liu, and N. Ozay, "Koopman-inspired implicit backward reachable sets for unknown nonlinear systems," *IEEE Control Syst. Lett.*, vol. 7, pp. 2245–2250, 2023.

[68] P. Arathoon and M. D. Kvalheim, "Koopman embedding and super-linearization counterexamples with isolated equilibria," 2023, *arXiv:2306.15126*.

[69] Z. Liu, N. Ozay, and E. D. Sontag, "Properties of immersions for systems with multiple limit sets with implications to learning Koopman embeddings," 2023, *arXiv:2312.17045*.

[70] A. Mauroy, Y. Susuki, and I. Mezić, *Koopman Operator in Systems and Control*. New York, NY, USA: Springer, 2020.

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

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

[73] M. Korda and I. Mezić, "On convergence of extended dynamic mode decomposition to the Koopman operator," *J. Nonlinear Sci.*, vol. 28, no. 2, pp. 687–710, Apr. 2018.

[74] F. Nüske, S. Peitz, F. Philipp, M. Schaller, and K. Worthmann, "Finite-data error bounds for koopman-based prediction and control," *J. Nonlinear Sci.*, vol. 33, no. 1, p. 14, Feb. 2023.

[75] 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, Feb. 2016, Art. no. e0150171.

[76] M. Korda and I. Mezić, "Optimal construction of Koopman eigenfunctions for prediction and control," *IEEE Trans. Autom. Control*, vol. 65, no. 12, pp. 5114–5129, Dec. 2020.

[77] E. Kaiser, J. N. Kutz, and S. L. Brunton, "Data-driven discovery of Koopman eigenfunctions for control," *Mach. Learning: Sci. Technol.*, vol. 2, no. 3, Sep. 2021, Art. no. 035023.

[78] N. Takeishi, Y. Kawahara, and T. Yairi, "Learning Koopman invariant subspaces for dynamic mode decomposition," in *Proc. Conf. Neural Inf. Process. Syst.*, Jan. 2017, pp. 1130–1140.

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

[80] S. E. Otto and C. W. Rowley, "Linearly recurrent autoencoder networks for learning dynamics," *SIAM J. Appl. Dyn. Syst.*, vol. 18, no. 1, pp. 558–593, Jan. 2019.

[81] M. Sznajer, "A data driven, convex optimization approach to learning Koopman operators," in *Proc. Learn. Dyn. Control Conf.*, May 2021, pp. 436–446.

[82] M. Haseli and J. Cortés, “Learning Koopman eigenfunctions and invariant subspaces from data: Symmetric subspace decomposition,” *IEEE Trans. Autom. Control*, vol. 67, no. 7, pp. 3442–3457, Jul. 2022.

[83] M. Haseli and J. Cortés, “Parallel learning of Koopman eigenfunctions and invariant subspaces for accurate long-term prediction,” *IEEE Trans. Control Netw. Syst.*, vol. 8, no. 4, pp. 1833–1845, Dec. 2021.

[84] M. Haseli and J. Cortés, “Generalizing dynamic mode decomposition: Balancing accuracy and expressiveness in Koopman approximations,” *Automatica*, vol. 153, Jul. 2023, Art. no. 111001.

[85] Z. Drmač, I. Mezić, and R. Mohr, “Data driven modal decompositions: Analysis and enhancements,” *SIAM J. Scientific Comput.*, vol. 40, no. 4, pp. A2253–A2285, Jan. 2018.

[86] H. Zhang, S. T. M. Dawson, C. W. Rowley, E. A. Deem, and L. N. Cattafesta, “Evaluating the accuracy of the dynamic mode decomposition,” *J. Comput. Dyn.*, vol. 7, no. 1, pp. 35–56, Nov. 2019.

[87] S. Pan, N. Arnold-Medabali, and K. Duraisamy, “Sparsity-promoting algorithms for the discovery of informative Koopman-invariant subspaces,” *J. Fluid Mech.*, vol. 917, p. 18, Jun. 2021.

[88] M. J. Colbrook, L. J. Ayton, and M. Szőke, “Residual dynamic mode decomposition: Robust and verified koopmanism,” *J. Fluid Mech.*, vol. 955, p. A21, Jan. 2023.

[89] M. J. Colbrook and A. Townsend, “Rigorous data-driven computation of spectral properties of Koopman operators for dynamical systems,” *Commun. Pure Appl. Math.*, vol. 77, no. 1, pp. 221–283, Jan. 2024.

[90] 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,” *Experim. Fluids*, vol. 57, no. 3, p. 42, Feb. 2016.

[91] O. Azencot, W. Yin, and A. Bertozzi, “Consistent dynamic mode decomposition,” *SIAM J. Appl. Dyn. Syst.*, vol. 18, no. 3, pp. 1565–1585, Jan. 2019.

[92] L. Lortie, S. Dahdah, and J. R. Forbes, “Forward-backward extended DMD with an asymptotic stability constraint,” 2024, *arXiv:2403.10623*.

[93] O. Azencot, N. B. Erichson, V. Lin, and M. W. Mahoney, “Forecasting sequential data using consistent Koopman autoencoders,” in *Proc. Int. Conf. Mach. Learn.*, vol. 1, Jul. 2020, pp. 475–485.

[94] M. Haseli and J. Cortés, “Temporal forward–backward consistency, not residual error, measures the prediction accuracy of extended dynamic mode decomposition,” *IEEE Control Syst. Lett.*, vol. 7, pp. 649–654, 2023.

[95] S. Klus, P. Koltai, and C. Schütte, “On the numerical approximation of the perron-frobenius and Koopman operator,” *J. Comput. Dyn.*, vol. 3, no. 1, pp. 1–12, Sep. 2016.

[96] M. Haseli and J. Cortés, “Invariance proximity: Closed-form error bounds for finite-dimensional koopman-based models,” 2023, *arXiv:2311.13033*.

[97] X. Li, S. Wang, and Y. Cai, “Tutorial: Complexity analysis of singular value decomposition and its variants,” 2019, *arXiv:1906.12085*.

[98] J. Demmel, I. Dumitriu, and O. Holtz, “Fast linear algebra is stable,” *Numerische Math.*, vol. 108, no. 1, pp. 59–91, Oct. 2007.

[99] P. Ruoff, M. K. Christensen, J. Wolf, and R. Heinrich, “Temperature dependency and temperature compensation in a model of yeast glycolytic oscillations,” *Biophys. Chem.*, vol. 106, no. 2, pp. 179–192, Nov. 2003.

[100] B. C. Daniels and I. Nemenman, “Efficient inference of parsimonious phenomenological models of cellular dynamics using S-systems and alternating regression,” *PLoS ONE*, vol. 10, no. 3, Mar. 2015, Art. no. e0119821.

[101] E. Yeung, S. Kundu, and N. Hadas, “Learning deep neural network representations for Koopman operators of nonlinear dynamical systems,” in *Proc. Amer. Control Conf. (ACC)*, Jul. 2019, pp. 4832–4839.

[102] Y. Lan and I. Mezić, “Linearization in the large of nonlinear systems and Koopman operator spectrum,” *Phys. D, Nonlinear Phenomena*, vol. 242, no. 1, pp. 42–53, Jan. 2013.



MASIH HASELI received the B.Sc. and M.Sc. degrees in electrical engineering from the Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran, in 2013 and 2015, respectively, and the Ph.D. degree in engineering sciences (mechanical engineering) from the University of California San Diego, La Jolla, CA, USA, in 2022. He is currently a Postdoctoral Researcher with the Department of Mechanical and Aerospace Engineering, University of California San Diego. His research interests include system identification, nonlinear systems, network systems, data-driven modeling and control, and distributed and parallel computing. He was a recipient of the Bronze Medal of the 2014 Iran National Mathematics Competition and the Best Student Paper Award of the 2021 American Control Conference.



JORGE CORTÉS (Fellow, IEEE) received the Licenciatura degree in mathematics from Universidad de Zaragoza, Zaragoza, Spain, in 1997, and the Ph.D. degree in engineering mathematics from Universidad Carlos III de Madrid, Madrid, Spain, in 2001. He was a Postdoctoral Researcher with the University of Twente, Twente, The Netherlands, and the University of Illinois at Urbana–Champaign, Urbana, IL, USA. He was also an Assistant Professor with the Department of Applied Mathematics and Statistics, University of California at Santa Cruz, Santa Cruz, CA, USA, from 2004 to 2007. He is currently a Professor and the Cymer Corporation Endowed Chair of High-Performance Dynamic Systems Modeling and Control at the Department of Mechanical and Aerospace Engineering, University of California San Diego. His research interests include distributed control and optimization, network science, non-smooth analysis, reasoning and decision-making under uncertainty, network neuroscience, and multi-agent coordination in robotic, power, and transportation networks. He is a fellow of SIAM and IFAC.