

Computationally Efficient Learning of Large Scale Dynamical Systems: A Koopman Theoretic Approach*

1st Sinha, Subhrajit

Biological Science Facility

Pacific Northwest National Laboratory

Richland, USA

subhrajit.sinha@pnnl.gov

2nd Nandanoori, Sai Pushpak

Control and Optimization Group

Pacific Northwest National Laboratory

Richland, USA

saipushpak.n@pnnl.gov

3rd Yeung, Enoch

Mechanical Engineering Department

University of California, Santa Barbara

Santa Barbara, USA

eyeung@ucsb.edu

Abstract—In recent years there has been a considerable drive towards data-driven analysis, discovery and control of dynamical systems. To this end, operator theoretic methods, namely, Koopman operator methods have gained a lot of interest. In general, the Koopman operator is obtained as a solution to a least-squares problem, and as such, the Koopman operator can be expressed as a closed-form solution that involves the computation of Moore-Penrose inverse of a matrix. For high dimensional systems and also if the size of the obtained data-set is large, the computation of the Moore-Penrose inverse becomes computationally challenging. In this paper, we provide an algorithm for computing the Koopman operator for high dimensional systems in a time-efficient manner. We further demonstrate the efficacy of the proposed approach on two different systems, namely a network of coupled oscillators (with state-space dimension up to 2500) and IEEE 68 bus system (with state-space dimension 204 and up to 24,000 time-points).

I. INTRODUCTION

Dynamical systems theory dates back to Newton when he came up with the laws of motion. Since then, it has flourished as a branch of mathematics and physics, with applications to a plethora of scientific and engineering disciplines. In recent times, with advancements in computational and data handling capacity, data-driven analysis of dynamical systems has gained a real impetus.

In the realm of data-driven analysis of dynamical systems, one of the most popular methods is based on operator theoretic methods, namely, the Koopman and Perron-Frobenius (P-F) operators [1]. Given a dynamical system, P-F and Koopman operators are typically infinite-dimensional operators which govern the evolution of distributions (or measures) and functions, respectively, under the system dynamics [1]–[6]. Though these operators are infinite-dimensional, one major advantage is the fact that these are linear operators and hence even if the underlying system is nonlinear, one obtains a linear system, albeit infinite-dimensional [1]. It is the linearity property that has driven researchers to explore these operator theoretic ideas for data-driven analysis of dynamical systems.

This work was supported partially by a Defense Advanced Research Projects Agency (DARPA) Grant No. DEAC0576RL01830 and Institute of Collaborative Biotechnologies Grant.

Data-driven discovery and control of dynamical systems, using the P-F and Koopman operators involves the construction of finite-dimensional approximations of these operators. Computation of P-F operators involves set-theoretic methods [7]–[9] and is typically computationally expensive. Finite-dimensional approximation of Koopman operator, on the other hand, is computed by studying the evolution of *observables* and usually involves finding a solution to a least-squares problem and hence is computationally much less expensive. Moreover, a Koopman operator can be trained on only a few trajectories and thus, one does not require a large number of initial conditions. Hence, the Koopman operator is more popular for data-driven analysis of dynamical systems. This has resulted in a huge drive in the development of Koopman operator methods for dynamical systems analysis and control [2]–[7], [10]–[24].

Among the different methods for computation of the finite-dimensional approximation of the Koopman operator, the most popular methods are Dynamic Mode Decomposition (DMD) [4] and Extended Dynamic Mode Decomposition (EDMD) [14]. Typically the trajectories are lifted to a higher dimensional feature space and the finite-dimensional Koopman operator is obtained as a solution of a norm minimization problem on the feature space and as such when the norm considered is the 2-norm, the Koopman operator can be computed directly using the Moore-Penrose inverse. However, for high dimensional systems computation of the Moore-Penrose and hence the Koopman operator becomes computationally challenging. In particular, the standard algorithms for computation of the Moore-Penrose inverse involves Singular Value Decomposition (SVD) and the computation is of the order $O(n^3)$. Moreover, if the number of data-points is large, it poses similar challenges.

In this paper, we provide a new method for computing the Koopman operator for high dimensional systems (and/or large data sets) in a time-efficient manner. In particular, the method is based on Cholesky decomposition [25] so that the dimension of the matrix being inverted is reduced. Furthermore, if parallel processing capabilities is utilised, the computation

arXiv:2007.00835v1 [eess.SY] 2 Jul 2020

time is reduced further, thus making this algorithm for data-driven identification of dynamical systems using Koopman operator easily scalable. To demonstrate the efficacy of the proposed approach, we consider two different dynamical systems wherein one example we increase the dimension of the system (keeping the number of data points constant) and compare the computation times, while in the other example we consider a high dimensional system and consider different sizes of the data-set.

The paper is organized as follows. In section II, we briefly review the concepts of P-F and Koopman operators, followed by the EDMD algorithm for the finite-dimensional approximation of the Koopman operator in section III. In section IV, we describe the algorithm for the efficient computation of the Koopman operator. In section V, we demonstrate the efficacy of the method by computing the Koopman operators for two high-dimensional systems, namely, a network of linear oscillators and IEEE 68 bus system. The paper is finally concluded in section VI.

II. PRELIMINARIES

Consider a discrete-time dynamical system

$$x_{t+1} = T(x_t) \quad (1)$$

where $T : X \subseteq \mathbb{R}^N \rightarrow X$ is assumed to be at least \mathcal{C}^1 . Associated with the dynamical system (1) is the Borel- σ algebra $\mathcal{B}(X)$ on X and the vector space $\mathcal{M}(X)$ of bounded complex valued measures on X . With this, two linear operators, namely, Perron-Frobenius (P-F) and Koopman operator, can be defined as follows [1]:

Definition 1: The P-F operator $\mathbb{P} : \mathcal{M}(X) \rightarrow \mathcal{M}(X)$ is given by

$$[\mathbb{P}\mu](A) = \int_X \delta_{T(x)}(A) d\mu(x) = \mu(T^{-1}(A))$$

$\delta_{T(x)}(A)$ is stochastic transition function which measure the probability that point x will reach the set A in one time step under the system mapping T .

Definition 2: Given any $h \in \mathcal{F}(X)$, the Koopman operator $\mathbb{U} : \mathcal{F}(X) \rightarrow \mathcal{F}(X)$ is defined as $[\mathbb{U}h](x) = h(T(x))$, where $\mathcal{F}(X)$ is the space of functions (observables), defined on X , invariant under the action of the Koopman operator.

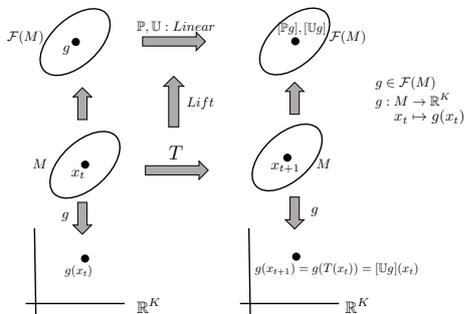


Fig. 1. Schematic of the P-F and Koopman operators.

Both the Perron-Frobenius and the Koopman operators are linear operators, even if the underlying system is nonlinear. But while analysis is made tractable by linearity, the trade-off is that these operators are typically infinite-dimensional. In particular, the P-F operator and Koopman operator often will lift a dynamical system from a finite-dimensional space to generate an infinite-dimensional linear system.

III. DATA-DRIVEN DISCOVERY OF DYNAMICAL SYSTEMS

The P-F and Koopman operators are adjoint to each other, however, since the Koopman operator governs the evolution of functions under the system dynamics (instead of densities as in the case of P-F operators) it is more suitable for data-driven analysis and discovery of dynamical systems.

A. Extended Dynamic Mode Decomposition

In this subsection, we briefly describe the popular Extended Dynamic Mode Decomposition (EDMD) algorithm for the computation of the Koopman operator from time-series data [14].

Let

$$X_p = [x_1, x_2, \dots, x_M], \quad X_f = [y_1, y_2, \dots, y_M] \quad (2)$$

where $x_i \in X$ and $y_i \in X$, be snapshots of data obtained from a simulation of a system $x \mapsto T(x)$ or from an experiment. The two pair of data sets are assumed to be two consecutive snapshots i.e., $y_i = T(x_i)$. Let $\mathcal{D} = \{\psi_1, \psi_2, \dots, \psi_K\}$ be the set of dictionary functions or observables, where $\psi : X \rightarrow \mathbb{C}$. Let $\mathcal{G}_{\mathcal{D}}$ denote the span of \mathcal{D} such that $\mathcal{G}_{\mathcal{D}} \subset \mathcal{G}$, where $\mathcal{G} = L_2(X, \mathcal{B}, \mu)$. Define vector valued function $\Psi : X \rightarrow \mathbb{C}^K$

$$\Psi(x) := [\psi_1(x) \quad \psi_2(x) \quad \dots \quad \psi_K(x)]^T \quad (3)$$

Any function $\phi, \hat{\phi} \in \mathcal{G}_{\mathcal{D}}$ can be written as

$$\phi = \sum_{k=1}^K a_k \psi_k = \Psi^T a, \quad \hat{\phi} = \sum_{k=1}^K \hat{a}_k \psi_k = \Psi^T \hat{a} \quad (4)$$

for some set of coefficients $a, \hat{a} \in \mathbb{C}^K$. Let

$$\hat{\phi}(x) = [\mathbb{U}\phi](x) + r,$$

where r is a residual function that appears because $\mathcal{G}_{\mathcal{D}}$ is not necessarily invariant to the action of the Koopman operator. Let \mathbf{K} be the finite dimensional approximation of the Koopman operator. Then the matrix \mathbf{K} is obtained as a solution of least square problem as follows

$$\min_{\mathbf{K}} \|\mathbf{K}Y_p - Y_f\|_2 \quad (5)$$

where

$$Y_p = \Psi(X_p) = [\Psi(x_1), \Psi(x_2), \dots, \Psi(x_M)] \quad (6)$$

$$Y_f = \Psi(X_f) = [\Psi(y_1), \Psi(y_2), \dots, \Psi(y_M)],$$

with $\mathbf{K} \in \mathbb{C}^{K \times K}$. The optimization problem (5) can be solved using a multitude of techniques like batch gradient descent, stochastic gradient descent or can be solved explicitly to obtain the following solution for the matrix \mathbf{K}

$$\mathbf{K}_{EDMD} = Y_f Y_p^\dagger \quad (7)$$

where \mathbf{G}^\dagger is the Moore-Penrose of matrix \mathbf{G} . DMD is a special case of EDMD algorithm with $\Psi(x) = x$.

IV. EFFICIENT COMPUTATION OF KOOPMAN OPERATOR

Since the Koopman operator is obtained as a solution of a least-squares problem, one can solve it directly by computing the Moore-Penrose inverse of a suitable matrix. The Moore-Penrose inverse of a $p \times q$ matrix M is a unique $q \times p$ matrix M^\dagger , such that

$$MM^\dagger M = M; \quad M^\dagger MM^\dagger = M^\dagger.$$

When M is of full rank the Moore-Penrose inverse is the usual pseudo-inverse $M^\dagger = (M^\top M)^{-1}M^\top$. However, when M is rank deficient, the computation of M^\dagger is not straightforward and there are several algorithms to compute the Moore-Penrose inverse [26], with the Singular Value Decomposition (SVD) technique being one of the most commonly used algorithms. However, the computational time of these algorithms grows rapidly with the size of the matrix M .

With this said, the EDMD algorithm usually suffers from the curse of dimensionality. The key feature of the EDMD algorithm is to choose a set of dictionary functions \mathcal{D} , such that the dictionary functions are rich enough to approximate the Koopman invariant subspaces. To achieve this, usually, the cardinality of the set \mathcal{D} is large and hence the dimension of the feature space is large, even if the dimension of the underlying system is small. As such, the number of dictionary functions increases rapidly with the dimension of the underlying system. Hence, the computation of the Koopman operator becomes slow and is thus computationally inefficient. Furthermore, often the matrices Y_p and Y_f , as defined in (6), are rank deficient, especially if most of the data points x_i are close to an attractor set. Hence, computing the Koopman operator using the existing algorithms becomes quite complex.

A. The algorithm

Let $M^\top M$ be a $q \times q$ symmetric matrix so that its rank is $r \leq q$. Hence, there exists a unique upper triangular matrix U with $(q - r)$ zero rows such that

$$U^\top U = M^\top M.$$

Hence, removal of the zero rows results in a matrix $L \in \mathbb{R}^{r \times q}$ with full rank r . Moreover,

$$U^\top U = M^\top M = LL^\top.$$

With this we have

Theorem 3:

$$M^\dagger = L(L^\top L)^{-1}(L^\top L)^{-1}L^\top M^\top. \quad (8)$$

Proof. From [27], we have for a product matrix XY ,

$$(XY)^\dagger = Y^\top (X^\top X Y Y^\top)^\dagger X^\top.$$

Hence, when Y is the identity matrix, we have

$$X^\dagger = (X^\top X)^\dagger X^\top.$$

Moreover, if $B = X^\top$ and $X \in \mathbb{R}^{q \times r}$ is of rank r , then

$$(X X^\top)^\dagger = X(X^\top X)^{-1}(X^\top X)^{-1}X^\top.$$

Hence,

$$(M^\top M)^\dagger = (LL^\top)^\dagger = L(L^\top L)^{-1}(L^\top L)^{-1}L^\top. \quad (9)$$

Using, $X^\dagger = (X^\top X)^\dagger X^\top$, we have

$$\begin{aligned} M^\dagger &= (M^\top M)^\dagger M^\top \\ &= L(L^\top L)^{-1}(L^\top L)^{-1}L^\top M^\top. \end{aligned} \quad (10)$$

■

Using the above theorem, the computation of Koopman operator can be summarized as follows:

Algorithm 1 Efficient Computation of Koopman Operator

- 1) From the obtained data set form the sets X_p and X_f as given in (2).
 - 2) Choose the dictionary functions $\Psi(x)$.
 - 3) Compute Y_p and Y_f as given by (6).
 - 4) Using the Cholesky decomposition of Y_p , compute the Moore-Penrose inverse (Y_p^\dagger) of Y_p by the formula (10).
 - 5) Compute the Koopman Operator as $\mathbf{K} = Y_f Y_p^\dagger$.
-

B. Computational cost

The above theorem provides an algorithm of finding the Moore-Penrose inverse of a rank deficient non-square matrix. Computationally, there are two steps that are computationally expensive, namely, computing full rank Cholesky decomposition of $M^\top M$ and computing the inverse of $L^\top L$. On a serial processor, these computations are of order $O(q^3)$ and $O(r^3)$, respectively. However, if one utilizes parallel processing, then the complexity is of order $O(q)$ [28]. Furthermore, even when using serial processing, inverting a $r \times r$ matrix is computationally less expensive than inverting a $q \times q$ matrix, where $r < q$. Thus, even in serial processing, the above algorithm is efficient and when parallel processing is available, the complexity of the inversion step is of order $O(\log r)$ [29].

V. SIMULATION RESULTS

In this section, we demonstrate the efficacy of the proposed approach over SVD based method for computation of the Koopman operator for a high-dimensional system. In one of the examples, namely a network of coupled oscillators, we vary the number of oscillators in the network, while keeping the number of time-points of the data constant, while in the power network example, we vary the number of data points, while keeping the number of states of the system constant. The rationale behind this approach is the fact that while computing the Koopman operator, the size of the matrix whose inverse is to be computed can become large because of one or both of two reasons, namely, the number of states is large or the number of data points is large. We address both the cases in the following simulations. All the simulations were performed in MATLAB R2019a on an Apple Macbook Pro with 2.3 GHz Intel Core i5 processor and 8 GB 2133 MHz LPDDR3 RAM.

A. Network of Coupled Oscillators

Consider a network of coupled linear oscillators given by

$$\ddot{\theta}_k = -\mathcal{L}_k\theta - d\dot{\theta}_k, \quad k = 1, \dots, N \quad (11)$$

where θ_k is the angular position of the k^{th} oscillator, N is the number of oscillators, \mathcal{L}_k is the k^{th} row of the Laplacian \mathcal{L} and d is the damping coefficient. In the simulations, the damping coefficient has been set to 0.4 for all the oscillators and the data was collected with a sampling time of $\delta t = 0.01$ seconds. Furthermore, we assume a ring topology of the network as shown in Fig. 2.

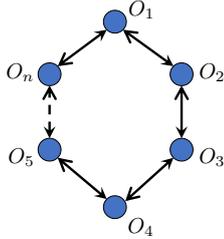


Fig. 2. Ring network of second order oscillators.

This network of oscillators has some similarities with the power network in the sense that each oscillator model follows closely with the second-order model of a synchronous generator and as they are connected over a network, it represents the interactions of the generators. The weighted Laplacian matrix shown in (11) captures the admittance matrix of the power network.

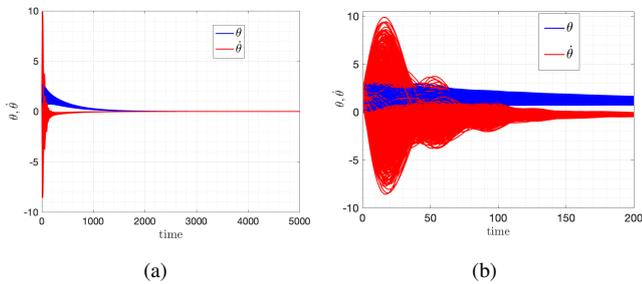


Fig. 3. (a) Time domain trajectories of the oscillator network with 1250 oscillators. (b) Transient trajectories of the oscillator network with 1250 oscillators.

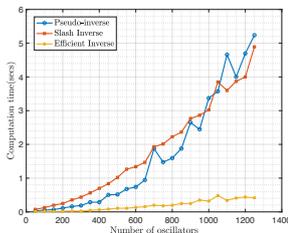


Fig. 4. Comparison of computation times of the Koopman operator for the oscillator network using different algorithms.

In the simulations, we vary the number of oscillators from 50 oscillators to 1250 oscillators, so that the dimension of the system varies from 100 to 2500. Furthermore, in all the simulations, data was collected for 5000 time steps. The time-domain trajectories of 1250 oscillators are shown in Fig. 3. Again, since the dynamical system is linear, we use linear dictionary functions, that is, $\Psi(x) = x$ and as such the dimension of the feature space is equal to the dimension of the underlying system.

With the 5000 data points, we compute the Koopman operator for the various networks and compare the computation times of the various algorithms. In particular, we compare the efficiency of our algorithm with the existing MATLAB commands of `pinv`, which uses SVD decomposition, and “\” (`backslash`), which uses QR decomposition with pivoting.

The computation times of the different algorithms for computing the Koopman operator is shown in Fig. 4. As can be seen from the figure, the computation time of the Koopman operator is greatly reduced by the proposed method. Moreover, the trend is almost linear, whereas the other algorithms scale in a nonlinear fashion with the size of the underlying system dimension. As such, the proposed method will scale much more efficiently with the dimension of the system.

The eigenvalues of the obtained Koopman operators (proposed method and `pinv` method) for 750 oscillators are shown in Fig. 5 and it can be seen that the dominant eigenvalues are similar using the two different techniques. As such, the long term behaviour, which is governed by the dominant eigenvalues, will be similar for both the Koopman operators. Note that the eigenvalue plots act as a validation of the obtained Koopman model.

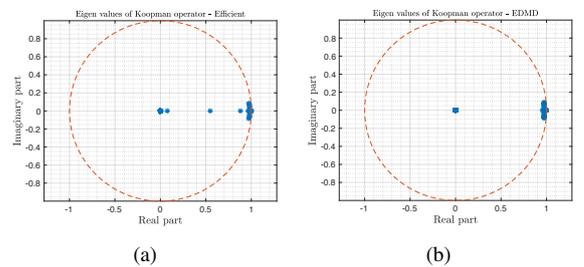


Fig. 5. (a) Eigenvalues of the Koopman operator when computed using proposed approach. (b) Eigenvalues of the Koopman operator computed using the SVD based inverse computation (`pinv` in MATLAB).

B. Discovery of Power Network

In this subsection, we demonstrate the proposed algorithm on the real-time PMU data. *GridSTAGE* [30] simulation platform developed by Pacific Northwest National Laboratory (PNNL) under the support of Department of Energy’s Advanced Grid Modeling program is leveraged to generate the synthetic data for a power network. *GridSTAGE* is developed based on Power System Toolbox (PST) [31] where nonlinear time-domain simulations can be generated for standard IEEE bus systems. *GridSTAGE* emulates PMU, SCADA sensors

and provides rich multivariate spatio-temporal network data. In particular, a IEEE 68 bus system is considered (Fig. 6) and random load changes were considered at random bus locations with small magnitudes to generate transient data for the power network. PMU data whose resolution is 50 measurements/second is generated and for the sake of convenience, we assume a PMU is available at each bus. PMU data contains nodal measurements such as voltage magnitude (in p.u.), voltage angle, frequency and rate of change of frequency along with branch measurements such as current magnitude and current angle.

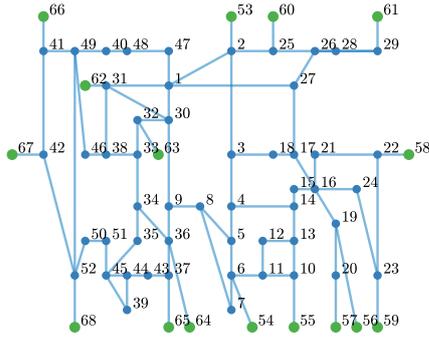


Fig. 6. IEEE 68 bus network topology. The green nodes correspond to the generators.

The high-level objective here is to identify the evolution of power system dynamics in a purely data-driven fashion applying Koopman operator theory. Specifically, we chose the nodal measurements of PMUs voltage magnitudes (V_m), voltage angles (V_a) and frequencies (f) at each bus to learn an equivalent representation of the power system dynamics in terms of a Koopman operator. Sample time-domain trajectories (data-set used to compute the different Koopman operators) of the variables are shown in Figs. 7 and 8.

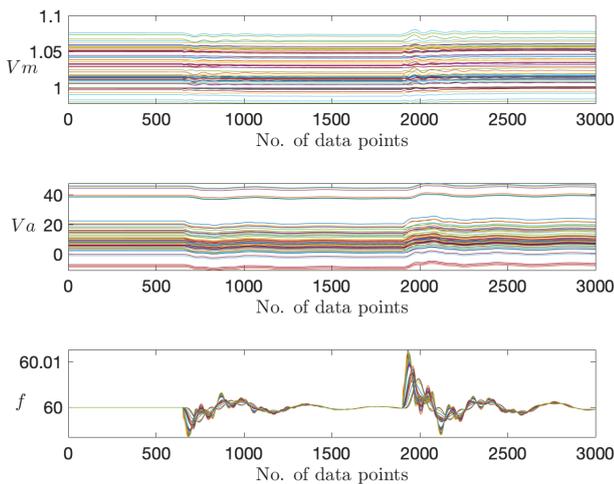


Fig. 7. Training data using PMU measurements for 3000 time steps.

As discussed in Eq. (7), it is seen that a pseudo-inverse needs to be computed to solve for the Koopman operator. As there are 68 buses and each bus has 3 states (voltage magnitude, voltage angle and frequency), the combined state space of the system is 204 (3×68). To demonstrate the proposed computation of the Koopman operator in a computationally efficient manner, we consider the PMU data for 8 seconds of simulation where there are at most 2 load changes every second. Each second of PMU data contains 3000 time points and hence the total length of PMU time-series data for 8 seconds is 24000. Now, 8 different Koopman operators are computed from 8 different data-sets. The size of the data set is varied from 3000 time steps to 24000 time steps in steps of 3000 time points.

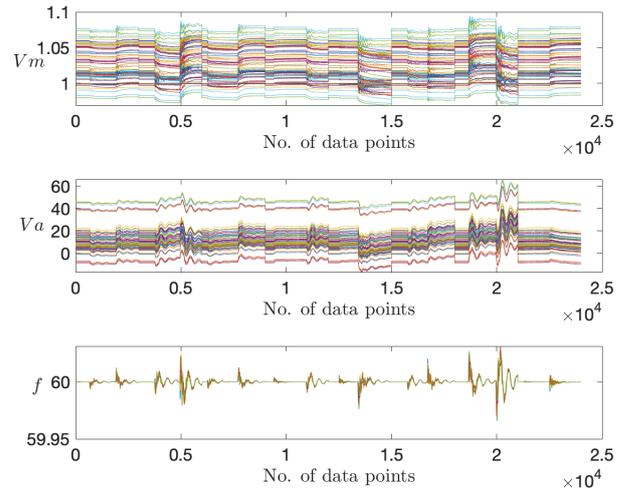


Fig. 8. Training data using PMU measurements for 24000 time steps.

EDMD algorithm with 1000 Gaussian radial basis functions (RBFs) is considered and hence the corresponding size of the Koopman operator is 1000×1000 . The computation times of the Koopman operators are shown in Fig. 9. As in the case of the oscillator network, it can be seen that the computation time of the Koopman operator using the proposed method is substantially lower than the SVD based (`pinv`) method. Moreover, the trend for the proposed approach is linear and as such, this method will scale much more efficiently compared to the existing method.

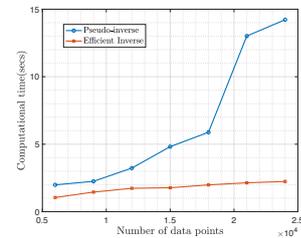


Fig. 9. Comparison of computation times of the Koopman operator for the power network with different algorithms.

Furthermore, in Fig. 10 we plot the eigenvalues of the two Koopman operators obtained by the two different methods and observe that the eigenspectrum is almost similar and thus verifying that the proposed method not only computes the Koopman operator in a time-efficient way, but it does compute the correct Koopman operator.

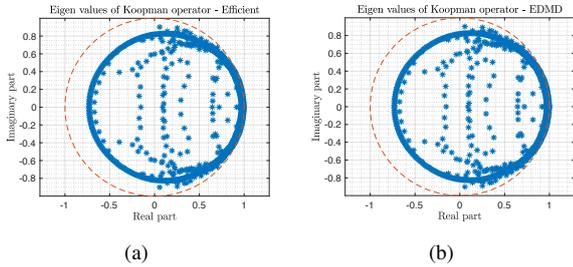


Fig. 10. (a) Eigenvalues of the Koopman operator when computed using proposed approach. (b) Eigenvalues of the Koopman operator computed using the SVD based inverse computation (`pinv` in MATLAB).

VI. CONCLUSIONS

In this paper, we provide an algorithm for time-efficient identification of large scale dynamical systems from time-series data. In particular, the algorithm efficiently computes the finite-dimensional approximation of the Koopman operator associated with a dynamical system. In general, the computation of the Koopman operator involves the computation of Moore-Penrose inverse of a matrix and if the state-space dimension is large, the computation of the inverse is computationally expensive. In this paper, we discuss a new algorithm based on Cholesky decomposition of the matrix to be inverted, which reduces the computation time significantly. The proposed approach's efficiency is further demonstrated on two different systems, namely, a network of coupled oscillators and the IEEE 68 bus system.

REFERENCES

- [1] A. Lasota and M. C. Mackey, *Chaos, Fractals, and Noise: Stochastic Aspects of Dynamics*. New York: Springer-Verlag, 1994.
- [2] U. Vaidya and P. G. Mehta, "Lyapunov measure for almost everywhere stability," *IEEE Transactions on Automatic Control*, vol. 53, no. 1, pp. 307–323, 2008.
- [3] I. Mezić and A. Banaszuk, "Comparison of systems with complex behavior," *Physica D*, vol. 197, pp. 101–133, 2004.
- [4] I. Mezić, "Spectral properties of dynamical systems, model reduction and decompositions," *Nonlinear Dynamics*, vol. 41, no. 1-3, pp. 309–325, 2005.
- [5] P. G. Mehta and U. Vaidya, "On stochastic analysis approaches for comparing dynamical systems," in *Proceeding of IEEE Conference on Decision and Control*, Spain, 2005, pp. 8082–8087.
- [6] M. Budisic, R. Mohr, and I. Mezić, "Applied koopmanism," *Chaos*, vol. 22, pp. 047510–32, 2012.
- [7] O. Junge and H. Osinga, "A set oriented approach to global optimal control," *ESAIM: Control, Optimisation and Calculus of Variations*, vol. 10, no. 2, pp. 259–270, 2004.
- [8] S. Sinha, U. Vaidya, and R. Rajaram, "Optimal placement of actuators and sensors for control of nonequilibrium dynamics," in *2013 European Control Conference (ECC)*. IEEE, 2013, pp. 1083–1088.
- [9] —, "Operator theoretic framework for optimal placement of sensors and actuators for control of nonequilibrium dynamics," *Journal of Mathematical Analysis and Applications*, vol. 440, no. 2, pp. 750–772, 2016.

- [10] M. Dellnitz and O. Junge, "On the approximation of complicated dynamical behavior," *SIAM Journal on Numerical Analysis*, vol. 36, pp. 491–515, 1999.
- [11] I. Mezić and A. Banaszuk, "Comparison of systems with complex behavior: spectral methods," in *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No.00CH37187)*, vol. 2, 2000, pp. 1224–1231 vol.2.
- [12] G. Froyland, "Extracting dynamical behaviour via Markov models," in *Nonlinear Dynamics and Statistics: Proceedings, Newton Institute, Cambridge, 1998*, A. Mees, Ed. Birkhauser, 2001, pp. 283–324.
- [13] M. Dellnitz, O. Junge, W. S. Koon, F. Lekien, M. Lo, J. E. Marsden, K. Padberg, R. Preis, S. D. Ross, and B. Thiere, "Transport in dynamical astronomy and multibody problems," *International Journal of Bifurcation and Chaos*, vol. 15, pp. 699–727, 2005.
- [14] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data-driven approximation of the koopman operator: Extending dynamic mode decomposition," *Journal of Nonlinear Science*, vol. 25, no. 6, pp. 1307–1346, 2015.
- [15] Y. Susuki and I. Mezić, "Nonlinear koopman modes and coherency identification of coupled swing dynamics," *IEEE Transactions on Power Systems*, vol. 26, no. 4, pp. 1894–1904, 2011.
- [16] E. Yeung, Z. Liu, and N. O. Hodas, "A koopman operator approach for computing and balancing gramians for discrete time nonlinear systems," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 337–344.
- [17] E. Yeung, S. Kundu, and N. Hodas, "Learning deep neural network representations for koopman operators of nonlinear dynamical systems," *arXiv preprint arXiv:1708.06850*, 2017.
- [18] S. Sinha, B. Huang, and U. Vaidya, "Robust approximation of koopman operator and prediction in random dynamical systems," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 5491–5496.
- [19] —, "On robust computation of koopman operator and prediction in random dynamical systems," *Journal of Nonlinear Science*, pp. 1–34, 2019.
- [20] S. Sinha, U. Vaidya, and E. Yeung, "On computation of koopman operator from sparse data," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 5519–5524.
- [21] S. Sinha, S. P. Nandanoori, and E. Yeung, "Koopman operator methods for global phase space exploration of equivariant dynamical systems," *arXiv preprint arXiv:2003.04870*, 2020.
- [22] —, "Online learning of dynamical systems: An operator theoretic approach," *arXiv preprint arXiv:1909.12520*, 2019.
- [23] —, "Data driven online learning of power system dynamics," *arXiv preprint arXiv:2003.05068*, 2020.
- [24] S. P. Nandanoori, S. Sinha, and E. Yeung, "Data-driven operator theoretic methods for global phase space learning," *arXiv preprint arXiv:1910.03011*, 2020.
- [25] P. Courrieu, "Fast computation of moore-penrose inverse matrices," *arXiv preprint arXiv:0804.4809*, 2008.
- [26] A. Ben-Israel and T. N. Greville, *Generalized inverses: theory and applications*. Springer Science & Business Media, 2003, vol. 15.
- [27] M. A. Rakha, "On the moore-penrose generalized inverse matrix," *Applied Mathematics and Computation*, vol. 158, no. 1, pp. 185–200, 2004.
- [28] V. Voevodin, "Voevodin vl. v," *Parallel computing. St. Petersburg: BHV-Petersburg*, 2002.
- [29] P. Courrieu, "Solving time of least square systems in sigma-pi unit networks," *arXiv preprint arXiv:0804.4808*, 2008.
- [30] "Gridstage: Spatiotemporal adversarial scenario generation framework," March 2020. [Online]. Available: <https://github.com/pnnl/GridSTAGE>
- [31] P. W. Sauer, M. A. Pai, and J. H. Chow, *Power system dynamics and stability: with synchrophasor measurement and power system toolbox*. John Wiley & Sons, 2017.