A Computationally Optimal Randomized Proper Orthogonal Decomposition Technique

Dan Yu, Suman Chakravorty

***Abstract*—In this paper, we consider the model reduction problem of large-scale systems, such as systems obtained through the discretization of partial differential equations. We propose a computationally optimal randomized proper orthogonal decomposition (RPOD*) technique to obtain the reduced order model by perturbing the primal and adjoint system using Gaussian white noise. We show that the computations required by the RPOD* algorithm is orders of magnitude lower while its performance is much better than the BPOD output projection algorithm. We also relate the RPOD* algorithm to Krylov subspace methods and show that it constitutes a fundamentally different approach to computational linear algebra problems.**

***Keywords*—Model Reduction, Proper Orthogonal Decomposition (POD), Krylov subspace methods, Randomization Algorithm.**

I. Introduction

In this paper, we are interested in the model reduction of large scale systems such as those governed by partial differential equations (PDE). The dimension of the system is large due to the discretization of the PDEs. For instance, consider the atmospheric dispersion of an air pollutant [1]. The emission of contaminants on the ground level is shown in Fig. 1 with four point sources labeled from S1 to S4.

This is a three dimensional problem, and after discretizing the PDE, the dimension of the system is 10^6. Therefore, we are interested in constructing a reduced order model (ROM) that can capture the input/output characteristics of the large model such that this ROM can then be used by a filtering algorithm for updating the states of the field, such as the Kalman filter. Also, the actuators and sensors could be placed anywhere in this field, which leads to a model reduction problem of a large-scale system with a large number of inputs/outputs.

In the past few decades, two model reduction techniques, singular value decomposition (SVD)-based methods and Krylov subspace methods have been studied in detail [2]. Surveys on model reduction methods can be found in, e.g., [3], [4].

For large-scale systems, Krylov subspace methods [2], [5] which compute subspace bases using the Lanczos [6] and Arnoldi [7] procedures iteratively have been widely used. Compared with SVD-based methods, the computational and storage cost using Krylov subspace methods are reduced, while

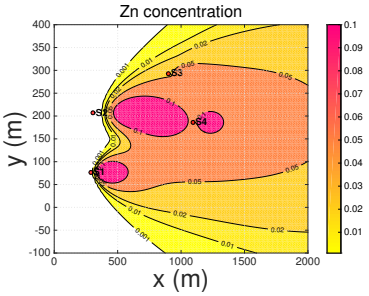


Fig. 1. Air pollutant problem

in general, important properties of the original system such as stability and passivity are not preserved [3], [5].

Among SVD-based methods, balanced proper orthogonal decomposition (BPOD) [8], [9] was proposed as an approximate version of balanced truncation [10]. Balancing transformations are constructed using impulse responses of both the primal and adjoint system, and hence, the most controllable and observable modes can be kept in the ROM. In 1978, Kung [11] presented a new model reduction algorithm in conjunction with the SVD technique, and the eigensystem realization algorithm (ERA) [12] was developed based on this technique. BPOD is equivalent to ERA procedure [13], and forms the Hankel matrix using the primal and adjoint system simulations as opposed to the input-output data as in ERA.

The primary drawback of BPOD and ERA is that for a large scale system, such as that obtained by discretizing a PDE, with a large number of inputs/outputs, the computational burden incurred is very high. There are two main parts to the computation: the first is to collect datasets from computationally expensive primal and adjoint simulations in order to generate the Hankel matrix. The second part is to solve the SVD problem for the resulting Hankel matrix.

To reduce the computational cost, various methods have been proposed in literature. For example, [9] proposed a BPOD output projection method to address the problem when the number of outputs is large, albeit the method is still faced with a very high computational burden when both the numbers of inputs and outputs are large. In [14], a method is proposed to reduce the number of snapshots, however, the primary problem regarding large number of inputs/outputs remains the same. More recently, [15] proposes a tangential interpolation-based ERA (TERA) method, which projects the impulse responses of original system onto suitably chosen directions, and constructs a projected Hankel matrix. In [16], a CUR-based ERA method constructs a sub-Hankel matrix by searching over columns and

D. Yu is with the Department of Aerospace Engineering, Texas A&M University, College Station, TX, 77840 USA e-mail: (yudan198811@hotmail.com).

S. Chakravorty is with the Department of Aerospace Engineering, Texas A&M University, College Station, TX 77840 USA e-mail: (schakrav@tamu.edu)

rows of the full Hankel matrix. Similarities and differences between the above mentioned algorithms and the RPOD* algorithm, are discussed in detail in Section V-B.

Recently, randomized algorithms, such as random sampling algorithms and random projection algorithms have received a lot of attentions in computer science [17], [18] and control community [19]. For a large scale matrix H, random sampling algorithms construct a rank k approximation matrix $\hat{H}$ by choosing and rescaling some columns of H according to certain sampling probabilities [17], [20], so the errors are bounded with high probability. In random projection method [18], a Gaussian test matrix Ω is generated, and the orthonormal matrix Q is constructed by performing a QR factorization of the matrix product $H\Omega$. The large matrix H is projected on to Q such that the error is bounded with high probability.

We had introduced an RPOD algorithm in [21] that randomly chooses a subset of the input/output trajectories. A sub-Hankel matrix is constructed using these sampled trajectories, which is then used to form a ROM in the usual BPOD fashion. The major problem associated with this algorithm is that different choices of sampling algorithms would lead to different lower bounds, and the choice of a good sampling algorithm other than the uniform distribution is unclear. In this paper, we propose the RPOD* algorithm where we perturb the primal and adjoint system with Gaussian white noise, and we show that the Markov parameters of the ROM constructed using RPOD* are close to Markov parameters of the full order system. The main contribution of RPOD* method in model reduction is that only one primal and one adjoint simulation are needed, and the construction of the full Hankel matrix is avoided, and hence, it is computational orders of magnitudes less expensive when compared to BPOD/ERA algorithms for a large-scale system with a large number of inputs/outputs. Fundamentally, the RPOD* constructs a randomized version of the standard Krylov subspace, which is orders of magnitude less time and memory consuming than Krylov subspace methods, while finding the orthonormal basis for the randomized subspace reduces to a very small symmetric Eigenvalue problem. Owing to the central importance of Krylov subspace methods in computational Linear Algebra, we believe that the RPOD* represents a fundamentally different approach to solving such problems that is orders of magnitude more efficient, while at the same time being much more accurate, than standard Krylov subspace methods.

The rest of the paper is organized as follows. In Section II, the problem is formulated, and in Section III, we introduce the partial realization problem and review the BPOD algorithm. Then we illustrate in a simplified fashion how to relate the BPOD ROM to the controllable and observable modes of the system. The RPOD* algorithm is proposed in Section IV, and the formal proofs and results are shown. Also, we discuss some implementation problems in this section. In Section V, we compare the RPOD* algorithm with related algorithms. In Section VI, we provide computational results comparing the RPOD* with BPOD output projection for a three dimensional atmospheric dispersion problem.

II. Problem Formulation

Consider a stable discrete-time linear input-output system:

$$x(k+1) = Ax(k) + Bu(k), y(k) = Cx(k), \tag{1}$$

where $x(k) \in \Re^N$, $u(k) \in \Re^p$, $y(k) \in \Re^q$ are the states, inputs, and outputs at discrete time instant k respectively, $k = 1, 2, \cdots$. Assume that A, B, C matrices are real.

The adjoint system is defined as:

$$z(k+1) = A'z(k) + C'v(k), w(k) = B'z(k), \tag{2}$$

where $z(k) \in \Re^N$, $v(k) \in \Re^q$, $w(k) \in \Re^p$ are the states, inputs, and outputs of the adjoint system at time k respectively. Denote $(.)'$ as the conjugate transpose of $(.)$.

Definition 1. The Markov parameters of the system are defined as $h_k = CA^{k-1}B, k = 1, \cdots$.

Assumption 1. Assume that A is stable and diagonalizable.

Under Assumption 1, let,

$$A = V\Lambda U' = \begin{pmatrix} V_1 & \cdots & V_N \end{pmatrix} \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_N \end{pmatrix} \begin{pmatrix} U_1' \\ \vdots \\ U_N' \end{pmatrix}, \tag{3}$$

where λ_i is the i^{th} eigenvalue, and (V_i, U_i) is the corresponding right and left eigenvector, $i = 1, 2, \cdots N$.

Definition 2. A mode (λ_i, V_i, U_i) is not controllable if $U_i'B = 0$ and is not observable if $CV_i = 0$, where (λ_i, V_i, U_i) is the i^{th} eigenvalue-eigenvector pair.

We partition the eigenvalues and eigenvectors (Λ, V, U) into four parts:

$$A = \begin{pmatrix} V_{co}' \\ V_{c\bar{o}}' \\ V_{\bar{c}o}' \\ V_{\bar{c}\bar{o}}' \end{pmatrix}' \begin{pmatrix} \Lambda_{co} & & & \\ & \Lambda_{c\bar{o}} & & \\ & & \Lambda_{\bar{c}o} & \\ & & & \Lambda_{\bar{c}\bar{o}} \end{pmatrix} \begin{pmatrix} U_{co}' \\ U_{c\bar{o}}' \\ U_{\bar{c}o}' \\ U_{\bar{c}\bar{o}}' \end{pmatrix}, \tag{4}$$

where $(\Lambda_{co}, V_{co}, U_{co})$ are the controllable and observable modes, $(\Lambda_{c\bar{o}}, V_{c\bar{o}}, U_{c\bar{o}})$ are the controllable but unobservable modes, $(\Lambda_{\bar{c}o}, V_{\bar{c}o}, U_{\bar{c}o})$ are the uncontrollable but observable modes, and $(\Lambda_{\bar{c}\bar{o}}, V_{\bar{c}\bar{o}}, U_{\bar{c}\bar{o}})$ are the uncontrollable and unobservable modes.

In this paper, we consider the model reduction problem for large-scale systems with a large number of inputs/outputs. The goal is to construct a ROM such that given the same input $u(k)$, the outputs of the ROM $\hat{y}(k)$ are close to the outputs of the full order system $y(k)$, i.e., $y(k) - \hat{y}(k)$ is small, $k = 1, 2, \cdots$.

Denote the number of the controllable and observable modes is exactly l throughout the paper. First, we summarize all the assumptions made in this paper.

A1. A is stable and diagonalizable.

A2. $l \ll N$.

A3a. $U_{\bar{c}o}'B = 0, U_{\bar{c}\bar{o}}'B = 0, CV_{c\bar{o}} = 0, CV_{\bar{c}\bar{o}} = 0$.

A3b. $U_{\bar{c}o}'B = \epsilon C_1, U_{\bar{c}\bar{o}}'B = \epsilon C_2, CV_{c\bar{o}} = \epsilon C_3, CV_{\bar{c}\bar{o}} = \epsilon C_4$, where ϵ is a small number, C_1, C_2, C_3, C_4 are constant matrices. And if $\|U_{co}'B\| = O(\|C_5\|), \|CV_{co}\| = O(\|C_5\|)$, then $\|C_1\|, \|C_2\|, \|C_3\|, \|C_4\| = O(\|C_5\|)$, where C_5 is a constant matrix, $\|.\|$ denotes the spectral 2-norm of (.).

The structure of the paper is summarized in Figure 2.

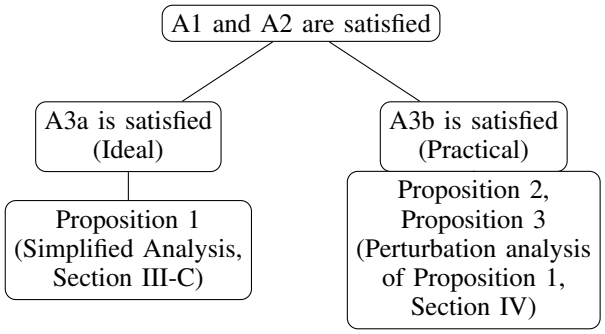


Fig. 2. Structure of the Paper

Discussion on Assumptions. For most of practical applications we consider, A1 is satisfied. A2 needs to be satisfied for the system to have a ROM, this assumption is typically satisfied for a large-scale system. It should be noticed that from Definition 2, assumption A3a is the statement of controllability/observability of different modes of the system. However, in practice, $U'_{\bar{c}o}B, CV_{c\bar{o}}$ are never exactly zero, and hence, in assumption A3b, we assume that $\|U'_{\bar{c}o}B\| \propto \epsilon, \|CV_{\bar{c}o}\| \propto \epsilon$, where ϵ is small.

III. Partial Realization Problem

In this section, first, we introduce the partial realization problem, and then we briefly review the BPOD algorithm, an approach used to solve the partial realization problem. A simplified analysis is provided, which is critical to understand the intuition behind the proposed RPOD* algorithm in Section IV.

A. Preliminaries

Denote the finite reachability matrix $\mathcal{R}(A,B)$ and observability matrix $\mathcal{O}(A,C)$, respectively, as

$$\begin{aligned}\mathcal{R}(A,B) &= [B, AB, \cdots, A^{s-1}B] \in \Re^{N\times ps},\\ \mathcal{O}(A,C) &= [C', A'C', \cdots, (A')^{s-1}C']' \in \Re^{qs\times N},\end{aligned} \quad (5)$$

where s is a finite number. The span of the columns of the matrix $\mathcal{R}(A,B)$ is denoted as span col $\mathcal{R}(A,B)$, and is referred to as a Krylov subspace.

Given a finite sequence of Markov parameters, the partial realization problem is defined as follows.

Definition 3 [2], [22]. Given a finite sequence of Markov parameters $h_k, k = 1, \cdots, 2s$, the partial realization problem consists of finding a positive integer N and constant matrices (A,B,C) such that $h_k = CA^{k-1}B, A \in \Re^{N\times N}, B \in \Re^{N\times p}, C \in \Re^{q\times N}, k = 1, \cdots, 2s$. Then (A, B, C) is called a partial realization of the sequence h_k.

Partial Realization by two-sided projection [2], [23]. Let biorthogonal bases $\Phi \in \Re^{N\times l}$, $\Psi \in \Re^{N\times l}$ be full column rank matrices, where $\Phi'\Psi = I$, and:

$$\begin{aligned}&\text{span col } \Psi = \text{span col } \mathcal{R}(A,B),\\ &\text{span col } \Phi = \text{span col } \mathcal{O}(A,C)',\end{aligned} \quad (6)$$

then the ROM $\hat{A} = \Phi'A\Psi, \hat{B} = \Phi'B, \hat{C} = C\Psi$ is a partial realization of (A,B,C), and matches $2s$ Markov parameters.

The biorthogonal bases Φ and Ψ can be constructed using different approaches, such as Krylov subspace methods and balanced truncation methods. In the following, we briefly review the BPOD algorithm, which constructs $\mathcal{R}(A,B)$ and $\mathcal{O}(A,C)$ using primal and adjoint simulations and constructs the biorthogonal bases by solving a SVD problem. A simplified analysis is given to illustrate how transformation bases and Markov parameters of the ROM constructed using BPOD algorithm can be related to the controllable and observable modes $(\Lambda_{co}, V_{co}, U_{co})$ of the system matrix A.

B. BPOD Algorithm

Consider the stable linear system (1)-(2), let $B = [b_1, \cdots, b_p]$ where $b_i, i = 1, \cdots, p$ is the i^{th} column of B, and $C = [c'_1, \cdots, c'_q]'$, where $c_j, j = 1, \cdots, q$ is the j^{th} row of C. The BPOD algorithm [9] is summarized in Algorithm 1. For simplicity, we assume that in both the primal and adjoint simulations, we take s snapshots at discrete time $0, 1, \cdots, s-1$.

C. Simplified Analysis

First, we construct a modal BPOD ROM by projecting the BPOD bases (T_b, S_b) onto the ROM eigenspace as in Algorithm 2.

Under assumptions A1, A2, and A3a, we have the following result.

Proposition 1: Denote $(\hat{A}_b, \hat{B}_b, \hat{C}_b)$ as the modal ROM constructed using Algorithm 2, (Φ_b, Ψ_b) as BPOD modal bases. Then

$$\begin{aligned}\hat{A}_b &= \Phi_b A \Psi_b = \Lambda_{co},\\ \hat{B}_b &= \Phi_b B = U'_{co}B,\\ \hat{C}_b &= C\Psi_b = CV_{co},\end{aligned} \quad (15)$$

where $(\Lambda_{co}, U_{co}, V_{co})$ are the controllable and observable modes of the system, and $\hat{C}_b\hat{A}_b^{k-1}\hat{B}_b = CA^{k-1}B, k = 1, 2, \cdots$.

Proof: The snapshot ensemble X_b can be written as:

$$\begin{aligned}X_b &= [b_1, \cdots, b_p, \cdots, A^{s-1}b_1, \cdots, A^{s-1}b_p],\\ &= [B, AB, \cdots, A^{s-1}B] = \mathcal{R}(A,B).\end{aligned} \quad (16)$$

Similarly, the adjoint snapshot ensemble Z_b is:

$$Z_b = [C', A'C', \cdots, (A')^{s-1}C'] = \mathcal{O}(A,C)'. \quad (17)$$

Under assumptions A1 and A3a, and from (4), we have:

$$X_b = V_{co}\alpha^b_{co} + V_{c\bar{o}}\alpha^b_{c\bar{o}}, Z_b = U_{co}\beta^b_{co} + U_{\bar{c}o}\beta^b_{\bar{c}o}, \quad (18)$$

where $\alpha^b_{co}, \alpha^b_{c\bar{o}}, \beta^b_{co}, \beta^b_{\bar{c}o}$ are some coefficient matrices.

We can prove that

$$\begin{aligned}&\text{span col } V_{co} \subseteq \text{span col } \Psi_b = \text{span col } \mathcal{R}(A,B),\\ &\text{span col } U_{co} \subseteq \text{span col } \Phi'_b = \text{span col } \mathcal{O}(A,C)',\end{aligned} \quad (19)$$

Algorithm 1 BPOD Algorithm

1) Collect impulse responses X_b of the primal system (1): Use $b_i, i = 1, \cdots, p$ as initial conditions for (1) with $u(k) = 0, k \geq 0$. Take s snapshots at discrete times $0, 1, \cdots, s-1$, and

$$X_b = [x_1(0), \cdots, x_p(0), \cdots, x_1(s-1), \cdots, x_p(s-1)], \tag{7}$$

where $x_i(t)$ is the state snapshot at time instant t with b_i as the initial condition, $t = 0, 1, \cdots, s-1$ and $i = 1, 2, \cdots, p$, $ps \geq l$.

2) Collect impulse responses Z_b of the adjoint system (2): Use $c'_j, j = 1, \cdots, q$ as initial conditions for (2) with $v(k) = 0, k \geq 0$. Take s snapshots at time $0, 1, \cdots, s-1$, and

$$Z_b = [z_1(0), \cdots, z_q(0), \cdots, z_1(s-1), \cdots, z_q(s-1)], \tag{8}$$

where $z_j(t)$ is the state snapshot of the adjoint system at time t with c'_j as the initial condition, $t = 0, 1, \cdots, s-1$ and $j = 1, 2, \cdots, q$, and $qs \geq l$.

3) Construct Hankel matrix H_b, and solve the SVD problem,

$$H_b = Z'_b X_b = (L_b \quad L_1) \begin{pmatrix} \Sigma_b & 0 \\ 0 & \Sigma_1 \end{pmatrix} \begin{pmatrix} R'_b \\ R'_1 \end{pmatrix}, \tag{9}$$

where Σ_b contains the first l non-zero singular values, and (L_b, R_b) are the corresponding left and right singular vectors.

4) Construct BPOD bases:

$$T_b = X_b R_b \Sigma_b^{-1/2}, S_b = \Sigma_b^{-1/2} L'_b Z'_b, \tag{10}$$

5) The ROM is:

$$A_b = S_b A T_b, B_b = S_b B, C_b = C T_b. \tag{11}$$

and hence, BPOD modal ROM $(\hat{A}_b, \hat{B}_b, \hat{C}_b)$ is a partial realization of (A, B, C). The formal proof is shown in Appendix A. ∎

Discussion on Proposition 1. The Hankel matrix $H_b \in \Re^{qs \times ps}$ has rank l, where $l \ll N, ps, qs$. From the development of Proposition 1, we see that if there exists two matrices X_{opt}, Z_{opt}, such that span col $V_{co} \subseteq$ span col X_{opt} and span col $U_{co} \subseteq$ span col Z_{opt}, then the same modal ROM can be constructed. Moreover, only l snapshot may be enough to extract all the controllable and observable modes of the system.

IV. Computationally Optimal Randomized Proper Orthogonal Decomposition (RPOD*)

In this section, first, we define the computationally optimal snapshot ensemble and prove that snapshot ensembles generated by perturbing the primal/adjoint system with white noise are computationally optimal snapshot ensembles. Then we propose the RPOD* algorithm in Section IV-B. We discuss implementation issues of the algorithm in Section IV-C.

Algorithm 2 BPOD modal ROM Algorithm

1) Construct BPOD ROM (A_b, B_b, C_b) and BPOD bases (T_b, S_b) using BPOD Algorithm 1.
2) Solve the eigenvalue problem of A_b:

$$A_b = P_b \Lambda_b P_b^{-1}, \tag{12}$$

where Λ_b is a diagonal matrix contains the eigenvalues as diagonal entries, and P_b are the corresponding eigenvectors.

3) Construct BPOD modal bases:

$$\Phi_b = P_b^{-1} S'_b, \Psi_b = T_b P_b, \tag{13}$$

4) The modal ROM is:

$$\hat{A}_b = \Phi_b A \Psi_b, \hat{B}_b = \Phi_b B, \hat{C}_b = C \Psi_b. \tag{14}$$

A. Computationally Optimal Snapshot Ensemble

Definition 4. Under assumptions A1, A2 and A3a, a computationally optimal primal snapshot ensemble X_{opt} is an l-snapshot ensemble of rank l which can be written as

$$\underbrace{X_{opt}}_{N \times l} = \underbrace{V_{co}}_{N \times l} \underbrace{\alpha_{opt}}_{l \times l} + \underbrace{V_{c\bar{o}} \bar{\alpha}_{opt}}_{N \times l}, \tag{20}$$

where α_{opt} is a rank l constant matrix, and $\bar{\alpha}_{opt}$ is a constant matrix of suitable dimensions.

A similar definition suffices for the computationally optimal adjoint snapshot ensemble Z_{opt}.

Under assumptions A1, A2 and A3a, A is stable, and there exists a finite number s, such that $\|A^s\| \approx 0$, we have the following result.

Proposition 2: The computationally optimal primal snapshot ensemble X_r can be constructed by perturbing the primal system (1) with white noise $u(k)$, and collecting l snapshots at time $t_1, t_2, \cdots, t_l$, i.e., $X_r = [x(t_1), x(t_2), \cdots, x(t_l)]$, where $t_l \geq s$, $\|A^s\| \approx 0$, and l is the number of the controllable and observable modes of the system.

Proof: The state snapshot $x(k)$ at time k is:

$$x(k) = \sum_{i=0}^{k-1} A^i B u(k-i), k = 1, 2, \cdots, \tag{21}$$

where $u(k)$ is white noise.

The snapshot ensemble X_r consists of two parts, snapshots taken before time s and snapshots taken after s.

Consider the snapshot ensemble which takes s snapshots from time $k = 1$ to $k = s$, then from (21),

$$(x(1) \quad x(2) \quad \cdots \quad x(s)) = \underbrace{\begin{pmatrix} B & AB & \cdots & A^{s-1}B \end{pmatrix}}_{\mathcal{R}(A,B)} \times$$

$$\underbrace{\begin{pmatrix} u(1) & u(2) & \cdots & u(s-1) & u(s) \\ 0 & u(1) & \cdots & u(s-2) & u(s-1) \\ 0 & 0 & \cdots & u(s-3) & u(s-2) \\ \vdots & \vdots & \dots & \dots & \vdots \\ 0 & 0 & \cdots & 0 & u(1) \end{pmatrix}}_{\Omega}, \tag{22}$$

where $\mathcal{R}(A,B)$ is the observability matrix. Denote $\Omega = (\omega_1 \quad \omega_2 \cdots \quad \omega_s)$, where ω_i is the i^{th} column of Ω. The Ω matrix above has columns that are linearly independent since it is upper triangular.

The snapshot $x(k)$ taken after s can also be written as: $x(k) = \mathcal{R}(A,B)\omega_k, k > s$, where ω_k is a column vector which is also independent of the columns of Ω with probability 1.

Thus, RPOD* snapshot ensemble X_r can be written as:

$$X_r = \mathcal{R}(A,B)\left(\omega_{t_1}, \quad \cdots, \quad \omega_{t_l}\right) = \mathcal{R}(A,B)\Omega_1, \tag{23}$$

where Ω_1 has full column rank. Thus,

$$\begin{aligned} X_r &= \mathcal{R}(A,B)\Omega_1 = V_{co}\alpha_{co}^b\Omega_1 + V_{c\bar{o}}\alpha_{c\bar{o}}^b\Omega_1 \\ &= \underbrace{V_{co}}_{N\times l}\underbrace{\alpha_{co}}_{l\times l} + V_{c\bar{o}}\alpha_{c\bar{o}}, \end{aligned} \tag{24}$$

where rank $(\alpha_{co}) = l$. Hence, X_r is a computationally optimal snapshot ensemble. ■

B. RPOD* Algorithm

The RPOD* algorithm is summarized in Algorithm 3. Under assumptions A1, A2 and A3b, the following result holds.

Proposition 3: Denote (A_r, B_r, C_r) as the ROM constructed using RPOD* following Algorithm 3. If we keep the first l non-zero singular values in (27), then $\|C_rA_r^{k-1}B_r - CA^{k-1}B\| \propto O(\epsilon), k = 1, \cdots$, where ϵ is a small number defined in assumption A3b.

Proof: The proof of Proposition 3 uses perturbation theory [24], [25] to extend the proof of the idealized Proposition 1 such that A3b holds instead of A3a. Under assumption A3b, the actual snapshot ensembles can be written as:

$$X_r = V_{co}\alpha_{co} + V_{c\bar{o}}\alpha_{c\bar{o}} + V_{\bar{c}o}\delta\alpha_{\bar{c}o} + V_{\bar{c}\bar{o}}\delta\alpha_{\bar{c}\bar{o}}, \tag{32}$$

where $\delta\alpha_{\bar{c}o} = \epsilon\alpha_{\bar{c}o}$, $\delta\alpha_{\bar{c}\bar{o}} = \epsilon\alpha_{\bar{c}\bar{o}}$ and ϵ is small. Therefore, $\|V_{\bar{c}o}\delta\alpha_{\bar{c}o} + V_{\bar{c}\bar{o}}\delta\alpha_{\bar{c}\bar{o}}\| = O(\epsilon)$ are small perturbations of the ideal snapshot ensemble, and we can expect the ideal result to be perturbed by a small amount as well.

The formal proof is shown in Appendix B. ■

Corollary 1: ϵ is assumed to be a small number in assumption A3b, and ϵ can also be related to σ_{l+1} as follows.

$$\|C_rA_r^iB_r - CA^iB\| \propto O(\epsilon) \propto O(\sigma_{l+1}). \tag{33}$$

The proof is shown in Appendix C.

C. Implementation Issues

Here we give the insight into how to collect snapshot ensembles.

From the analysis in Section IV-A, we only need to collect l snapshots from primal and adjoint simulations. However, l is not known a priori, thus, in practice, we estimate the number of snapshots need to be collected as follows. Suppose we collect m snapshots. We start with a random guess $m \ll N$, or we can choose m from experience. For instance, in a fluid system with $N = 10^6$, l is usually $O(10) \sim O(10^2)$, and hence, we start with $m = O(10)$. Then we check the rank of $Z_r'X_r$. If $Z_r'X_r$ has full rank, i.e., rank $(Z_r'X_r) = m$, it is possible that we did not take enough snapshots, and we increase m, until rank $(Z_r'X_r) < m$.

Selecting snapshots varies for different problems and here, we only provide a heuristic approach. More specifically, we assume that m snapshots are taken at time steps $T, 2T \cdots, mT$, WLOG, and we discuss how to choose T. We require that $mT \geq s$, where $\|A^s\| \approx 0$. In Fig. 3, we show one simulation result comparing the accuracy of the ROMs using $T = 3, 5, 10, 20, 50$ for the atmospheric dispersion problem introduced in Section VI. Here, $s = 900$, and $m = 300$. The error is defined in (38), and we take the average of the error for each ROM. It can be seen that as T increases, the ROM is more accurate, while it takes longer time to generate the

Algorithm 3 RPOD* Algorithm

1) Perturb the primal system (1) with white noise $u(k)$, collect l snapshots at time $t_1, t_2, \cdots, t_l$, where $t_l \geq s$, l is the number of controllable and observable modes present in the snapshot ensembles and $\|A^s\| \approx 0$. Denote the snapshot ensemble X_r as:
$$X_r = \left(x(t_1) \quad x(t_2) \quad \cdots \quad x(t_l)\right). \tag{25}$$
2) Perturb the adjoint system (2) with white noise $v(k)$, collect l snapshots at time $t_1, t_2, \cdots, t_l$. Denote the adjoint snapshot ensemble Z_r as:
$$Z_r = \left(z(t_1) \quad z(t_2) \quad \cdots \quad z(t_l)\right). \tag{26}$$
3) Solve the SVD problem:
$$Z_r'X_r = \begin{pmatrix} L_r & L_o \end{pmatrix}\begin{pmatrix} \Sigma_r & 0 \\ 0 & \Sigma_o \end{pmatrix}\begin{pmatrix} R_r' \\ R_o' \end{pmatrix}, \tag{27}$$
where Σ_r contains the first l non-zero singular values $\sigma_1 \geq \sigma_2 \geq \cdots, \geq \sigma_l > 0$, (R_r, L_r) are the corresponding right and left singular vectors.
4) Construct bases:
$$T_r = X_rR_r\Sigma_r^{-1/2}, S_r = \Sigma_r^{-1/2}L_r'Z_r'. \tag{28}$$
5) Construct the ROM $\tilde{A}$, find the eigenvalues Λ_r and eigenvectors P_r of $\tilde{A}$.
$$\tilde{A} = S_rAT_r = P_r\Lambda_rP_r^{-1}, \tag{29}$$
6) Construct RPOD* bases:
$$\Phi_r = P_r^{-1}S_r, \Psi_r = T_rP_r. \tag{30}$$
7) The ROM is:
$$A_r = \Phi_rA\Psi_r, B_r = \Phi_rB, C_r = C\Psi_r \tag{31}$$

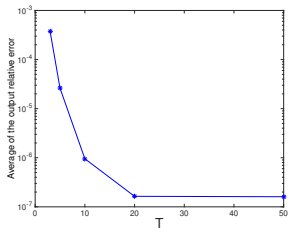


Fig. 3. Snapshot Selection: Effect of T

snapshots. Thus, this is a trade-off between the accuracy and computational efficiency.

V. Comparison with related Algorithms

Now we consider the RPOD* algorithm from a more general point of view.

The RPOD* snapshot ensemble can be written as

$$X_r = [B, AB, \cdots, A^{s-1}B]\Omega_1 \in \Re^{N\times l}, \tag{34}$$

where $\Omega_1 \in \Re^{ps\times l}$ is a full rank matrix. Therefore,

$$\text{span col } X_r = \text{span col } \mathcal{R}(A, B). \tag{35}$$

Note that this is the fundamental difference with all Krylov subspace methods. The construction of, and finding the bases of the Krylov subspaces is both time and memory expensive when the dimension of the system and the number of inputs/outputs are large. The RPOD* can be viewed as constructing a randomized version of the Krylov subspaces that is orders of magnitude less time and memory consuming than a typical Krylov subspace method, while solving a much smaller symmetric $l \times l$ Eigenvalue problem to find the orthonormal bases, such that:

$$\text{span col } \Psi_r = \text{span col } X_r. \tag{36}$$

Furthermore, owing to the central importance of Krylov subspace methods in computational linear algebra, we believe that the RPOD represents a fundamentally different approach to solving such problems that is orders of magnitude less time and memory consuming, while being much more accurate.*

A. Comparison with BPOD

As we mentioned in Section III-B, $p + q$ simulations are needed for BPOD algorithm. It is expensive to store $(p + q)s$ snapshots, and it is expensive to solve the resulting SVD problem due to the large size of the problem. For RPOD* algorithm, only (1 primal + 1 adjoint) simulations are needed, and only $2l$, where $l \ll s$ snapshots need to be stored. Also, it is easy to solve the resulting SVD problem.

Another practical problem with impulse responses snapshots is that snapshots after some time are dominated by very few slow modes. On the other hand, RPOD* trajectories are white noise forced, and all the modes are always present in all snapshots due to the persistent excitation of white noise terms [26]. Hence, RPOD* snapshots can be taken till time s, and be assured that all of the relevant modes will be captured.

B. Comparison with BPOD output projection, RPOD, TERA and CUR-based ERA

As we mentioned in Section I, various methods have been proposed to reduce the computational cost of solving SVD problem of the large Hankel matrix when the number of inputs/outputs is large. In Table I, the methods are divided into two groups: methods based on BPOD and methods based on ERA. Further, the methods are categorized based on the schemes to construct the reduced order Hankel matrix.

BPOD output projection [9] projects the outputs onto a smaller subspace with projection matrix $\Theta_o \in \Re^{q\times n_o}$, where n_o is the rank of the output projection, and constructs a projected adjoint system. TERA [15] projects the Markov parameters onto the smaller input and output subspaces using $W_1 \in \Re^{q\times l_1}, W_2 \in \Re^{p\times l_2}$ respectively, i.e., the projected Markov parameters $\hat{h}_k = W_1' h_k W_2 \in \Re^{l_1\times l_2}, k = 1, 2, \cdots$. RPOD [21] randomly chooses $\hat{p}$ columns and $\hat{q}$ rows of B, C and constructs the sub-Hankel matrix using these sampled trajectories. CUR-based ERA [16] constructs the full Hankel matrix H_b first, and then chooses l columns and rows of the full Hankel matrix H_b using some optimization algorithms.

Denote the Hankel matrix constructed using RPOD* as H_r, BPOD output projection as H_o, TERA as H_t, RPOD as H_r^s and CUR-based ERA as H_c. The constructed Hankel matrices can be viewed as projecting the full order Hankel matrix onto reduced order Hankel matrices with projection matrices as follows.

$$\begin{aligned}
H_r &= \Omega_2' H_b \Omega_1 \in \Re^{l\times l},\\
H_o &= \Theta' H_b \in \Re^{n_o s\times ps},\\
H_t &= \mathcal{W}_1' H_b \mathcal{W}_2 \in \Re^{l_1 s\times l_2 s},\\
H_r^s &= I_1 H_b I_2 \in \Re^{\hat{q}s\times \hat{p}s},\\
H_c &= I_3 H_b I_4 \in \Re^{l\times l},
\end{aligned} \tag{37}$$

where Θ is a diagonal matrix with Θ_o on diagonal entries, $\mathcal{W}_1, \mathcal{W}_2$ are diagonal matrices with W_1, W_2 on the diagonal entries respectively. $I_1 \in \Re^{\hat{q}s\times qs}$ is a 0-1 matrix, such that in each row, there is only one non-zero term, and $I_1(i, j) = 1, i = 1, \cdots, \hat{q}s$ indicate the j^{th} column of H_b is selected. I_2, I_3, I_4 are 0-1 matrices defined similarly.

Computational Complexity Analysis. Denote ΔT as the time to propagate the primal/adjoint system once. Both TERA and CUR-based ERA algorithm need to collect $2s$ Markov parameters which takes time $2sp\Delta T$. BPOD output projection algorithm and RPOD algorithm need $(p+n_o)$ simulations and $(\hat{p}+\hat{q})$ simulations respectively, and thus, the total computation time to generate snapshot ensembles is $(p+n_o)s\Delta T$ and $(\hat{p}+\hat{q})s\Delta T$. RPOD* needs 1 primal and 1 adjoint simulation till time t_l, where if we choose $t_l = s$, the total computation time to generate snapshot ensembles is $2s\Delta T$, which means that RPOD* computational cost is the same as BPOD in a single input single output system.

It can be seen that RPOD* and CUR-based ERA construct the smallest projected Hankel matrices. Solving SVD problems using H_r, H_c takes time $O(l^3)$, while CUR-based ERA requires the storage of the full Hankel matrix, which is avoided in RPOD*. CUR-ERA also requires the solution of a large

TABLE I. COMPARISON OF RELATED METHODS

	BPOD based	ERA based
By Projection	BPOD output projection [9], RPOD*	Tangential interpolation-based ERA (TERA) [15]
By Sampling	RPOD [21]	CUR-factored Hankel Approximation [16]

combinatorial optimization problem to find the optimal set of inputs and outputs which can be very time consuming, and which is not required by RPOD*.

VI. COMPUTATIONAL RESULTS

In this section, we show the comparison of RPOD* with BPOD output projection for a three dimensional atmospheric dispersion problem to illustrate the proposed algorithm.

First, we define the error:

$$E_{time}(k) = \frac{\|y_{true}(k) - y_{rom}(k)\|}{\|u(k)\|}, \tag{38}$$

where $y_{true}(k)$ are the outputs of the full order system at time k, $y_{rom}(k)$ are the outputs of the reduced order system, and $u(k)$ are the inputs.

The frequency response for multiple input multiple output systems can be represented by plotting the maximum singular value of the transfer function matrix $\sigma_{max}(H(j\omega))$ as a function of frequency ω, where $\sigma_{max}(.)$ denotes the largest singular value of $(.)$. We define the frequency responses error as:

$$E_{fre}(j\omega) = |\sigma_{max}(H_{true}(j\omega) - H_{rom}(j\omega))|, \tag{39}$$

where $H_{true}(j\omega)$ is the transfer function of the full order system, and $H_{rom}(j\omega)$ is the transfer function of the ROM.

For the experiment below, there are design parameters that need to be chosen manually, such as the rank of the output projection, when to take the snapshots, and the size of the ROM. We have shown the results of the best selections of these parameters for BPOD output projection algorithm and RPOD* algorithm.

The three-dimensional advection-diffusion equation describing the contaminant transport in the atmosphere is:

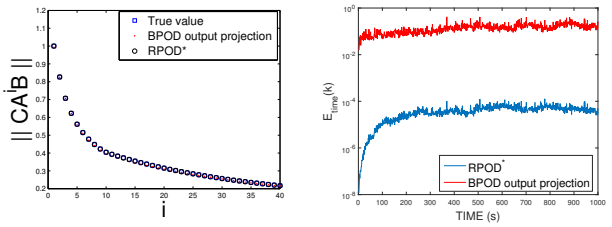
$$\frac{\partial c}{\partial t} + \nabla \cdot (c\vec{u}) = \nabla \cdot (K(\vec{X})\nabla c) + Q\delta(\vec{X} - \vec{X}_s), \tag{40}$$

where $c(\vec{X}, t)$ denotes the mass concentration at location $\vec{X} = (x, y, z)$, $\vec{X}_s = (x_s, y_s, z_s)$ denotes the location of point source, and Q is the contaminant emitted rate. The simulation domain $x \in [0, 2000](m), y \in [-100, 400](m), z \in [0, 50](m)$, and wind velocity $\vec{u} = (4m/s, 0, 0)$. ∇ denotes the gradient operator, and $K(\vec{X}) = diag(K_x(x), K_y(x), K_z(x))$ is a diagonal matrix whose entries are the turbulent eddy diffusivities.

The boundary conditions are:

$$c(0, y, z) = 0, c(\infty, y, z) = 0, c(x, \pm\infty, z) = 0,$$
$$c(x, y, \infty) = 0, K_z \frac{\partial c}{\partial z}(x, y, 0) = 0. \tag{41}$$

The system is discretized using finite difference method, and there are $100 \times 100 \times 10$ grid points which are equally spaced.



(a) Comparison of Markov Parameters (b) Comparison of Errors

Fig. 4. Atmospheric Dispersion Problem Time Responses

There are 10 point sources. We take the full field measurements (except the nodes on $x = 0, \infty$ and $y = \pm\infty$).

In this example, since the system dimension is $N = 10^5$, constructing the ROM with full field measurements using BPOD is computationally impossible, and thus, we only compare RPOD* algorithm with BPOD output projection algorithm.

For RPOD* algorithm, we collect snapshots sequentially. The system is perturbed by white noise with distribution $N(0, I_{10\times10})$. One primal simulation and one adjoint simulation are needed. We collect 400 equispaced snapshots from time $t \in [0, 4000]s$, where at time $s = 4000$, $\|A^s\| \approx 0$, and extract 380 modes. For BPOD output projection algorithm, we collect impulse responses from $p = 10$ primal simulations. We could not collect all impulse responses from $t \in [0, 4000]s$ due to the memory limits on the platform. Hence, for the best performance available in this example, we collect 300 equispaced snapshots from $t \in [0, 300]s$ for each primal simulation. The rank of the output projection is 100, and hence, 100 adjoint simulations are needed, and in the adjoint simulations, we collect 50 equispaced snapshots from $t \in [0, 50]s$. We can extract 250 modes.

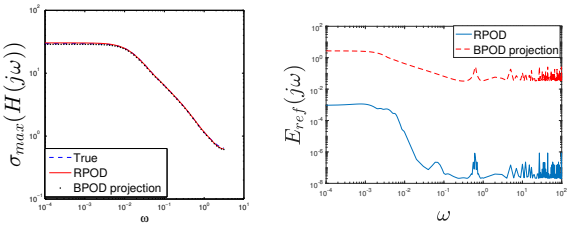
In Fig. 4(a), we compare Markov parameters of the ROM constructed using RPOD* and BPOD output projection with the full order system. Also, we perturb the system with random Gaussian noise, and compare errors in Fig. 4(b).

In Fig. 5(a), we compare frequency responses of the ROM constructed using RPOD* and BPOD output projection with the full order system. We can see that frequency responses of the ROMs are almost the same as the frequency responses of the full order system. In Fig. 5(b), we show the frequency responses errors.

Comparison of computational time is shown in Table II. It can be seen that for this example, construction of snapshot ensembles using RPOD* is faster than BPOD output projection, and the dominant computation cost is construction of $Z'X$, where RPOD* algorithm is almost 16500 times faster than BPOD output projection.

TABLE II. COMPARISON OF COMPUTATIONAL TIME

	Generate X	Generate Z	Construct $Z'X$	Solve SVD	Total Time
RPOD*	55.59s	56.23s	0.321s	0.4859s	112.6269s
output projection	30.461s	421.99s + 9.297s (projection)	5311s	9.118s	5781.856s



(a) Comparison of Frequency Responses (b) Comparison of Frequency Responses Errors

Fig. 5. Atmospheric Dispersion Problem Frequency Responses

All of the experiments reported in this paper were performed using Matlab 2013b on a Dell OptiPlex 9020, Intel(R) Core (TM) i7-4770CPU, 3.40GHz, 4GB RAM machine.

VII. CONCLUSION

In this paper, we have introduced a computationally optimal randomized POD procedure for extraction of ROMs for large scale systems such as those governed by PDEs. The ROM is constructed by perturbing the primal and adjoint system with Gaussian white noise, where the computational cost to construct snapshot ensembles is reduced when compared to perturbing the primal and adjoint system with impulses in BPOD/BPOD output projection algorithm. Also, it leads to a much smaller SVD problem, and an orders of magnitude reduction in the computation required for constructing ROMs over BPOD/ BPOD output projection procedure. The computational results show that the accuracy of RPOD* is much more accurate than BPOD output projection algorithm. We believe that the RPOD* represents a simple yet fundamentally different approach to Krylov subspace methods in computational linear algebra, and as such, its potential impact in large scale linear algebra problems might be highly significant.

APPENDIX A
PROOF OF PROPOSITION 1

Under assumption A3a, there are exactly l non-zero singular values in the resulting SVD problem, i.e.,

$$Z_b'X_b = L_b\Sigma_b R_b', \tag{42}$$

where $\Sigma_b \in \Re^{l\times l}$ are the l non-zero singular values and (L_b, R_b) are the corresponding left and right singular vectors. Consider the BPOD ROM:

$$\begin{aligned} A_b = S_bAT_b &= \Sigma_b^{-1/2}L_b'(Z_b'AX_b)R_b\Sigma_b^{-1/2} \\ &= \underbrace{\Sigma_b^{-1/2}L_b'(\beta_{co}^b)'}_{P_b}\Lambda_{co}\alpha_{co}^bR_b\Sigma_b^{-1/2}. \end{aligned} \tag{43}$$

It can be shown that Λ_{co} are the eigenvalues of A_b, and P_b are the eigenvectors. From (43),

$$\Lambda_{co} = \underbrace{(P_b^{-1}S_b)}_{\Phi_b}A\underbrace{(T_bP_b)}_{\Psi_b}. \tag{44}$$

It can be proved that Ψ_b, Φ_b are biorthogonal, and

$$\Psi_b = T_bP_b = V_{co} + V_{c\bar{o}}C_6, \tag{45}$$

where $C_6 = \alpha_{c\bar{o}}^bR_b\Sigma_b^{-1}L_b'(\beta_{co}^b)'$, and similarly,

$$\Phi_b = P_b^{-1}S_b = U_{co}' + C_7U_{\bar{c}o}', \tag{46}$$

where $C_7 = \alpha_{co}^bR_b\Sigma_b^{-1}L_b'(\beta_{\bar{c}o}^b)'$. Under assumption A3a, the modal BPOD ROM constructed using (Ψ_b, Φ_b) is:

$$\begin{aligned} \hat{A}_b &= \Phi_bA\Psi_b = \Lambda_{co}, \\ \hat{B}_b &= \Phi_b'B = U_{co}'B + C_7U_{\bar{c}o}'B = U_{co}'B, \\ \hat{C}_b &= C\Psi_b = CV_{co} + CV_{c\bar{o}}C_6 = CV_{co}. \end{aligned} \tag{47}$$

And hence, Markov parameters of the ROM are:

$$\hat{C}_b\hat{A}_b^{k-1}\hat{B}_b = CV_{co}\Lambda_{co}^{k-1}U_{co}'B = CA^{k-1}B, k = 1, 2, \cdots. \tag{48}$$

APPENDIX B
PROOF OF PROPOSITION 3

The adjoint snapshot ensemble Z_r can be written as:

$$Z_r = U_{co}\beta_{co} + U_{c\bar{o}}\delta\beta_{c\bar{o}} + U_{\bar{c}o}\beta_{\bar{c}o} + U_{\bar{c}\bar{o}}\delta\beta_{\bar{c}\bar{o}}, \tag{49}$$

where $\delta\beta_{c\bar{o}} = \epsilon\beta_{c\bar{o}}$, $\delta\beta_{\bar{c}\bar{o}} = \epsilon\beta_{\bar{c}\bar{o}}$, ϵ is defined in assumption A3b, $\beta_{c\bar{o}}, \beta_{\bar{c}\bar{o}}$ are coefficient matrices. From (32) and (49),

$$\begin{aligned} Z_r'X_r &= \beta_{co}'\alpha_{co} + \delta\beta_{c\bar{o}}'\alpha_{c\bar{o}} + \beta_{\bar{c}o}'\delta\alpha_{\bar{c}o} + \delta\beta_{\bar{c}\bar{o}}'\delta\alpha_{\bar{c}\bar{o}} \\ &= \beta_{co}'\alpha_{co} + \epsilon\underbrace{(\beta_{c\bar{o}}'\alpha_{c\bar{o}} + \beta_{\bar{c}o}'\alpha_{\bar{c}o})}_{E_1} + O(\epsilon^2), \\ &= \beta_{co}'\alpha_{co} + \epsilon E_1 + O(\epsilon^2). \end{aligned} \tag{50}$$

Denote $\bar{H}_r$ as the ideal Hankel matrix constructed with assumption A3a, and it can be proved that the true Hankel matrix H_r (assumption A3b is satisfied) can be viewed as adding a small perturbation of $\bar{H}_r$, i.e.,

$$\begin{aligned} H_r &= Z_r'X_r = \beta_{co}'\alpha_{co} + \epsilon E_1 = L_r\Sigma_rR_r' + L_o\Sigma_oR_o, \\ \bar{H}_r &= \beta_{co}'\alpha_{co} = \bar{L}_r\bar{\Sigma}_r\bar{R}_r' + \bar{L}_o\bar{\Sigma}_o\bar{R}_o', \end{aligned} \tag{51}$$

where $\Sigma_r \in \Re^{l\times l}$ contains the first l non-zeros singular values and L_r, R_r are the corresponding left and right singular vectors, $\Sigma_o \in \Re^{(n-l)\times(n-l)}$ contains the rest singular values, with left and right singular vectors L_o, R_o. Similarly, singular values and singular vectors of the ideal Hankl matrix are partitioned in the same way.

From the perturbation theory [24], [25], the perturbed singular values and singular vectors (Σ_r, L_r, R_r) are related to the singular values and singular vectors $(\bar{\Sigma}_r, \bar{L}_r, \bar{R}_r)$ as:

$$\begin{aligned}\Sigma_r &= \bar{\Sigma}_r + \epsilon E_2 + O(\epsilon^2),\\ L_r &= \bar{L}_r + \Delta L_r, R_r = \bar{R}_r + \Delta R_r,\end{aligned} \tag{52}$$

where $E_2, \Delta L_r, \Delta R_r$ are some matrices, and $\|\Delta L_r\|$, $\|\Delta R_r\| \propto O(\epsilon)$. E_2 is a diagonal matrix with diagonal elements e_i, where $e_i \propto O(\epsilon), i = 1, 2, \cdots l$.

Following from procedure in Appendix A, the RPOD* ROM

$$\begin{aligned}\tilde{A} &= S_r A T_r = \Sigma_r^{-1/2} L_r' Z_r' A X_r R_r \Sigma_r^{-1/2}\\ &= \Sigma_r^{-1/2} L_r' (\beta_{co}' \Lambda_{co} \alpha_{co} + \epsilon E_2) R_r \Sigma_r^{-1/2}.\end{aligned} \tag{53}$$

And it can be proved that:

$$\tilde{A} = \bar{\Sigma}_r^{-1/2} \bar{L}_r' \beta_{co}' \Lambda_{co} \alpha_{co} \bar{R}_r \bar{\Sigma}_r^{-1/2} + \Delta_3 + O(\epsilon^2), \tag{54}$$

where Δ_3 is some matrix, and $\|\Delta_3\|_2 = k_3\epsilon$, k_3 is a constant. If we let

$$\bar{A} = \underbrace{\bar{\Sigma}_r^{-1/2} \bar{L}_r' \beta_{co}'}_{\bar{P}_r} \Lambda_{co} \underbrace{\alpha_{co} \bar{R}_r \bar{\Sigma}_r^{-1/2}}_{\bar{P}_r^{-1}}, \tag{55}$$

then

$$\tilde{A} = \bar{A} + \Delta_3 + O(\epsilon^2) = P_r \Lambda_r P_r^{-1}, \tag{56}$$

where $\|P_r - \bar{P}_r\| \leq \|\Delta_3\| = k_3\epsilon, \|\Lambda_r - \Lambda_{co}\| \leq \|\Delta_3\| = k_3\epsilon$. Thus,

$$\begin{aligned}\Psi_r &= T_r P_r = X_r \bar{R}_r \bar{\Sigma}_r^{-1/2} \bar{P}_r + \Delta_4,\\ \Phi_r &= P_r^{-1} S_r = \bar{P}_r^{-1} \bar{\Sigma}_r^{-1/2} \bar{L}_r' Z_r' + \Delta_5,\end{aligned} \tag{57}$$

where Δ_4, Δ_5 are some matrices and $\|\Delta_4\|, \|\Delta_5\| \propto O(\epsilon)$. Substitute (57) into ROM Markov parameters, and collect the small perturbation terms, we have:

$$C_r A_r^i B_r = C \Psi_r \Lambda_r^i \Phi_r B = C V_{co} \Lambda_{co}^i U_{co}' B' + \Delta, \tag{58}$$

where Δ is some matrix, and $\|\Delta\| \propto O(\epsilon)$.

For details of the proof, please see [27].

Appendix C
Proof of Corollary 1

For $\bar{\sigma}_i \in \bar{\Sigma}_o$ (zero singular values) with multiplicity $n - l$, the corresponding left and right singular vectors are $\bar{L}_o, \bar{R}_o$. The perturbed singular values $\sigma_i \in \Sigma_o$ and from [24]

$$\sigma_i = \epsilon \sqrt{\lambda_i(\bar{R}_o' E_1' \bar{L}_o \bar{L}_o' E_1 \bar{R}_o)}, i = 1, \cdots n - l. \tag{59}$$

Therefore,

$$\sigma_l = \bar{\sigma}_l + e_l\epsilon + O(\epsilon^2), \sigma_{l+1} = e_{l+1}\epsilon \propto O(\epsilon), \tag{60}$$

Thus,

$$\|C_r A_r^i B_r - C A^i B\| \propto O(\epsilon) \propto O(\sigma_{l+1}). \tag{61}$$

References

[1] J. M. Stockie, "The Mathematics of Atmospheric Dispersion Modeling," *SIAM Review*, vol. 53, No.2, pp. 349–372, 2011.

[2] A. Antoulas, *Approximation of Large Scale Dynamical Systems*. Philadelphia, PA: SIAM, 2005.

[3] U. Baur, P. Benner, and L. Feng, "Model Order Reduction for Linear and Nonlinear Systems: A System-Theoretic Perspective," *Arch. Computat. Methods Eng.*, vol. 21, pp. 331–358, 2014.

[4] P. Benner, S. Gugercin, and K. Willcox, "A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems," *SIAM Review*, vol. 57, no. 4, pp. 483–531, 2015.

[5] Z. Bai, "Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems," *Applied Numerical Mathematics*, vol. 43, pp. 9–44, 2002.

[6] P. Feldmann and R. W.Freund, "Efficient Linear Circuit Analysis by Padé Approximation via the Lanczos Process," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 14, no. 5, pp. 639–649, 1995.

[7] I. M. Jaimoukha and E. M. Kasenally, "Implicitly Restarted Krylov Subspace Methods for Stable Partial Realizations," *SIAM Journal on Matrix Analysis and Applications*, vol. 18, no. 3, pp. 633–652, 1997.

[8] K. Willcox and J. Peraire, "Balanced Model Reduction via the Proper Orthogonal Decomposition," *AIAA Journal*, vol. 40, pp. 2323–2330, 2002.

[9] C. W. Rowley, "Model reduction for fluids using balanced proper orthogonal decomposition," *International Journal of Bifurcation and Chaos*, vol. 15, pp. 997–1013, 2005.

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

[11] S. Kung, "A new identification method and model reduction algorithm via singular value decomposition," *12th Asilomar Conference on Circuits, Systems and Computers*, pp. 705–714, Nov. 1978.

[12] J.-N. Juang and R. S. Pappa, "An eigensystem realization algorithm for model parameter identification and model reduction," *Journal of Guidance, Control, and Dynamics*, vol. 8, No.5, pp. 620–627, 1985.

[13] Z. Ma, S. Ahuja, and C. W. Rowley, "Reduced order models for control of fluids using the eigensystem realization algorithm," *Theoret. Comput. Fluid Dyn.*, vol. 25, pp. 233–247, 2011.

[14] J. H. Tu and C. W. Rowley, "An improved algorithms for balanced pod through an analytic treatment of impulse response tails," *Journal of Computational Physics*, vol. 231, pp. 5317–5333, June, 2012.

[15] B. Kramer and S. Gugercin, "Tangential interpolation-based eigensystem realization algorithm for MIMO systems," *Mathematical and Computer Modelling of Dynamical Systems*, vol. 22, no. 4, pp. 282–306, 2016.

[16] B. Kramer and A. A. Gorodetsky, "System Identification via CUR-factored Hankel Approximation," ACDL Technical Report TR16-6, Tech. Rep., 2016.

[17] M. W. Mahnoey, "Randomized Algorithms for Matrices and Data," *Found. Trends Mach. Learn.*, vol. 3, no. 2, pp. 123–224, 2011.

[18] N. Halko, P. Martinsson, and J. Tropp, "Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions," *SIAM review*, vol. 52(2), pp. 217–288, 2011.

[19] G. Calafiore and M. Campi, "The scenario approach to robust control design," *IEEE Transactions on Automatic Control*, vol. 51, pp. 742–753, 2006.

[20] P. Drineas *et al.*, "Fast monte carlo algorithms for matrices ii: Computing a low rank approximation to a matrix," *SIAM J. Computing*, vol. 36, pp. 158–183, 2006.

[21] D. Yu and S. Chakravorty, "A Randomized Proper Orthogonal Decomposition Technique," in *Proceedings of American Control Conference*, 2015, pp. 1127–1142.

[22] A.M.King, U.B.Desai, and R.E.Skelton, "A Generalized Approach to q-Markov Covariance Equivalent Realizations for Discrete Systems," *Automatica*, vol. 24, no. 4, pp. 507–515, 1988.

[23] M. Bastug *et al.*, "Model Reduction by Nice Selections for Linear Switched Systems," *IEEE Transactions on Automatic Control*, vol. 61, no. 11, pp. 3422 – 3437, Nov. 2016.

[24] T. Soderstrom, "Perturbation results for singular values," Department of Information Technology at Uppsala University, Tech. Rep., 1999.

[25] B. Friedlander and B. Porat, "First-order perturbation analysis of singular vectors in singular value decomposition," *IEEE Transactions on Signal Processing*, vol. 56, Issue 7, pp. 3044–3049, 2008.

[26] Y. Zhu, "Chapter 3 - Identification Test Design and Data Pretreatment," in *Multivariable System Identification For Process Control*. Oxford: Pergamon, 2001, pp. 31 – 63.

[27] D. Yu and S. Chakravorty, "A Computationally Optimal Randomized Proper Orthogonal Decomposition Technique," arXiv: 1509.05705, Tech. Rep., 2015.