Thai Journal of Mathematics : 222-256
Special Issue: Annual Meeting in Mathematics 2018
http://thaijmath.in.cmu.ac.th
Online ISSN 1686-0209

Model Order Reduction for Sine-Gordon Equation Using POD and DEIM

Norapon Sukuntee[†] **and Saifon Chaturantabut**[‡],[1]

[†]Department of Mathematics and Statistics, Thammasat University
e-mail : nsukuntee@gmail.com
[‡]Department of Mathematics and Statistics, Thammasat University
e-mail : saifon@mathstat.sci.tu.ac.th

Abstract : This work applies model reduction techniques for efficiently approximating the solution of the sin-Gordon equation. The proper orthogonal decomposition (POD) is employed to construct a low dimensional basis that can accurately capture the dynamics of the solution space. This POD basis is used with the Galerkin projection to obtain a reduced-order model that is much smaller than the original discretized system. However, the effectiveness of the POD-Galerkin approach is limited to the linear part of the system. The discrete empirical interpolation method (DEIM) is then applied to further reduce the computational complexity of the nonlinear term. This work investigates the effect of using different amount of snapshots from coarse discretization for constructing the POD basis, as well as demonstrates the applicability of the POD-DEIM approach on predicting solution of the parametrized sine-Gordon equation. The POD-DEIM solutions are shown to be accurate and can be solved with much less computational time and memory storage when compared to the high-dimensional discretized sine-Gordon equation.

Keywords : Model order reduction; Sine-Gordon equation; Differential equations; Proper orthogonal decomposition; Discrete empirical interpolation method

2000 Mathematics Subject Classification : 65F30; 3540; 34A45 (2000 MSC)

[0] This research was supported by the Thammsat University Research Scholar, Contract No. 2/19/2561

[1] Corresponding author email: saifon@mathstat.sci.tu.ac.th

1 Introduction

The sine-Gordon equation is a nonlinear hyperbolic partial equation involving the d'Alembert operator [30] and the sine of the unknown function. The equation was known in the nineteenth century in the course of study of various problems. The equation grew greatly in importance in the 1970s when it was realized that it led to solitons [18]: a self-reinforcing solitary wave package that maintains its shape while it propagates at a constant velocity. The sine-Gordon equation appears in many physical applications, including relativistic field theory [7], Josephson junction [15] and mechanical transmission line [29]. Many numerical methods for finding approximate solution of the sine-Gordon equation have received considerable attention in the literature [33, 34, 4, 19]. These methods usually require a large memory storage with long computational time and hence make it inefficient during computation. Model reduction techniques can be used to avoid this problem.

Projection-based model reduction approaches are commonly used in various applications. Ding, Lili and Kong and Qin [12] in 2017 presented two robust modal reduction bases, namely multi-model method (MM) and modal strain energy by first-order correction method (MSEC) to reduce the order of viscoelastic systems with multiple damping models. MM and MSEC are used to build a projection basis representing the complex eigensolutions of structural system. In [3], Bonotto, Cenedese and Bettini proposed a model order reduction method via Krylov subspace projection for applications in the field of computational electromagnetics. The block Arnoldi with modified Gram-Schmidt algorithm is employed to construct the basis for Krylov subspace. Model order reduction of glucose-insulin homeostasis using empirical Gramians and balanced truncation is discussed in [39]. Other projection-based model reduction includes Arnoldi-based model order reduction for linear systems with inhomogeneous initial conditions [35], data-driven operator inference for nonintrusive projection-based model reduction [31] and dynamic model reduction using data-driven Loewner-framework applied to thermally morphing structures [32]. For the sine-Gordon equation, Afkham and Hesthaven [25] studied projection-based model reduction approaches to construct a reduced-order model that preserves the symplectic symmetry of dissipative Hamiltonian systems. The basis used in [25] is generated from a greedy algorithm. In this work, we focus on an optimal basis in the least-squares sense computed from proper orthogonal decomposition (POD).

POD is also known by many different names, depending on the field of application. Principal component analysis (PCA) is probably the oldest, generally traced back more than a century to Pearson [13]. Other equivalent methods include the Karhunen-Loève transform [24], the Hotelling transform [17] and empirical orthogonal function (EOF) analysis [42]. POD has been applied in various fields. Karaszen, Akkoyunlu and Uzunca [20] used POD to derive a reduced-order model for the nonlinear Schrödinger equation. In fluid mechanics, POD is applied as a reduce order method for flow around an oscillating cylinder [22] and the upper tropical Pacific ocean [6]. In [16], a reduced-order optimal control methodology mainly based on

POD is proposed to optimize the operation of wells in water flooding reservoir. The authors of [8] presented a reduced-order model for wall shear stress in abdominal aortic aneurysms by using POD. Not only is POD used for model order reduction for differential equations, but also its concept can be extended to missing data reconstruction. In particular, The POD basis has been used for data reconstruction in the least-square sense for an inverse material characterization [40] and aerodynamic problems [28].

In this work, we consider a finite difference discretization of partial differential equations in a space variable, which will become a system of nonlinear ordinary differential equations. The system generally can be defined on a large domain that requires many spatial grid points. When we solve this full-order discretized system directly, the computation can lead to long simulation time and require large storage. To overcome this difficulty, the POD technique is used to construct a subspace spanned by a basis with much smaller dimension. Galerkin projection is then used to project the full-order system to the low-dimensional POD system. By using the POD-Galerkin approach for linear dynamical system, the solution is generally obtained with less computational effort while sacrificing small amount of accuracy compared with the solution from full-order system. However, the effective dimension reduction of the POD-Galerkin approach is limited to the linearity of the system. If the system has nonlinear properties, additional model reduction technique has to be applied to overcome this problem.

In the case of nonlinear dynamical systems, several methods are used for reducing the computational complexity. Mirgolbabaei and Echekki [26] presented kernel principal component analysis (KPCA), a nonlinear alternative to classical principal component analysis (PCA) for combustion composition space. The authors of [23] proposed a novel learning framework called quasi-curvature local linear projection (QLLP). This framework first selects small landmarks from original data to obtain the low-dimensional coordinates in quasi-curvature locally linear embedding (QLLE) and then adopts extreme learning machine (ELM) to learn the explicit mapping function from original data to low-dimensional coordinates for nonlinear dimensionality reduction. Furthermore, an enhanced 3D data transfer method for fluidstructure interface by isometric mapping (ISOMAP) nonlinear space dimension reduction is discussed in [43]. In general setting of large-scaled nonlinear differential equations, discrete empirical interpolation method (DEIM) is an efficient way to handle this problem. DEIM [9] was developed from the empirical interpolation method (EIM), which was first introduced by Barrault, Maday, Nguyen and Patera [1] in 2004. The DEIM procedure constructs a selected interpolation indices that indicate an interpolation-based projection to provide a nearly optimal subspace approximation to the nonlinear term. Consequently, the complexity in evaluating the nonlinear term becomes proportional to a small number of selected spatial indices.

This work applies the DEIM procedure with the POD technique to further reduce the complexity of nonlinearity. The POD-DEIM approach has been used for reducing computational complexity of nonlinear dynamical systems in various applications. In 2017, Dehghan and Abbaszadeh [11] presented a combination of POD-DEIM and meshless local RBF-DQ approach for the application in preventing of groundwater contamination. Yang and Veneziani [41] studied an efficient estimation of cardiac conductivities via POD-DEIM model order reduction. In [36], Stanko, Boyce and Yeh applied a nonlinear model reduction of unconfined groundwater flow using POD and DEIM. In addition, [5] employed POD-DEIM approach for efficient reduction of a dynamic 2D catalytic reactor model. The POD-DEIM is also used for computational EMG model [27], strain-softening viscoplasticity [14] and four dimensional variational data assimilation [37]. In this work, the POD-DEIM technique is applied to approximate solution of the sine-Gordon equation. As a result of this study, we will illustrate POD and POD-DEIM approximations compared with the full-order solution in the numerical experiments.

This work is arranged as follows. In Section 2, a brief overview of the methodology is given. In Section 3, we describe an application of the POD-DEIM approach on the sine-Gordon equation. Section 4 demonstrates the efficiency of the POD-DEIM approach using three numerical experiments. First, the accuracy and the efficiency in decreasing the storage and simulation time are shown through the sine-Gordon equation with a fixed parameter value. Next, the effect of using different amount of snapshots from coarse discretization to generate POD basis is investigated. Finally, the POD-DEIM approach is shown to accurately construct different reduced systems for various parameter values by using only one basis set for each POD and DEIM approximations. Section 5 provides a conclusion and discusses important issues presented in this work.

2 Methodology

We consider a finite difference discretization of a nonlinear hyperbolic partial differential equation in a spatial domain, which becomes a system of nonlinear second order ordinary differential equations of the form

$$\frac{d^2}{dt^2}\mathbf{u}(t) = \mathbf{A}\mathbf{u}(t) + \mathbf{F}(t, \mathbf{u}(t)), \tag{2.1}$$

with some appropriate initial conditions, where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is a constant matrix, $\mathbf{F} : \mathcal{D}\times\mathbb{R}^n \to \mathbb{R}^n$ is a nonlinear vector-valued function, and $\mathbf{u} : \mathcal{D} \to \mathbb{R}^n$ is the state variable with $\mathcal{D} \subset \mathbb{R}$. The dimension n of the problem is the number of spatial grid points used in the discretization, which generally can be very large to obtain accurate numerical solution. As a result, it can be very expensive to solve this system. We will apply model reduction techniques to decrease the computational complexity.

Projection-based techniques are commonly used for constructing a reduced-order system with much smaller dimension than that of the original system. In this work, the Galerkin projection will be used with a low-dimensional basis to construct a reduced-order system. In particular, let $\mathbf{V}_k = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k] \in \mathbb{R}^{n\times k}$ be a matrix whose columns are vectors in the reduced basis space and $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$ is a set of orthonormal vectors, i.e., $\mathbf{v}_i^T\mathbf{v}_j = \begin{cases} 0 & , i \neq j \\ 1 & , i = j \end{cases}$ for all $i, j = 1, 2, \ldots, k$. The solution of the original system can be approximated by $\mathbf{u}(t) \approx \mathbf{V}_k\tilde{\mathbf{u}}(t)$, where $\tilde{\mathbf{u}} : \mathcal{D} \to \mathbb{R}^k$. By replacing $\mathbf{V}_k\tilde{\mathbf{u}}(t)$ in (2.1) and applying the Galerkin projection onto a reduced subspace span$\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$, we obtain the following reduced-order system

$$\frac{d^2}{dt^2}\tilde{\mathbf{u}}(t) = \tilde{\mathbf{A}}\tilde{\mathbf{u}}(t) + \mathbf{V}_k^T\mathbf{F}(t, \mathbf{V}_k\tilde{\mathbf{u}}(t)), \tag{2.2}$$

where $\tilde{\mathbf{A}} = \mathbf{V}_k^T\mathbf{A}\mathbf{V}_k \in \mathbb{R}^{k\times k}$. The choice of the reduced basis clearly affects the accuracy of the approximation. Several techniques for constructing a set of reduced basis can be used here. In this work, we apply the proper orthogonal decomposition(POD) for obtaining a reduced basis that is optimal in the sense that a certain approximation error involving the snapshots is minimized. Hence, a reduced space spanned by the POD basis frequently gives an excellent low-dimensional approximation. The POD technique is described in Section. 2.1.

2.1 Proper orthogonal decomposition (POD)

POD is a method for constructing a low-dimensional representation of a subspace in Hilbert space. It efficiently extracts the basis elements that contain dominant characteristics of the space. POD can be obtained from the left singular vectors of the singular value decomposition (SVD) in Euclidean space. The POD basis in Euclidean space can be specified formally as follows.

Consider a set of snapshots $\{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m\} \subset \mathbb{R}^n$, where snapshots are samples of trajectories. Suppose that $\mathcal{U} = \text{span}\{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m\}$ with $\dim \mathcal{U} = r_u$. A POD basis of dimension $k < r_u$ is a set of orthonormal vectors whose linear span is the best approximation of the space $\mathcal{U}$, i.e. it solves the following minimization problem

$$\min_{\{\phi_i\}_{i=1}^k} \sum_{j=1}^m \|\mathbf{u}_j - \sum_{i=1}^k (\mathbf{u}_j^T\phi_i)\phi_i\|_2^2, \tag{2.3}$$

with constrains $\phi_i^T\phi_j = \delta_{ij} = \left\{\begin{array}{ll} 0 & , i \neq j \\ 1 & , i = j \end{array}\right.$ for $i, j = 1, 2, \ldots, k$. It is well known that the solution to (2.3) is provided by the set of left singular vectors of the snapshot matrix $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m] \in \mathbb{R}^{n\times m}$. In particular, suppose the SVD of $\mathbf{U}$ is

$$\mathbf{U} = \mathbf{V}\Sigma\mathbf{W}^T, \tag{2.4}$$

where $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{r_u}] \in \mathbb{R}^{n\times r_u}$ and $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{r_u}] \in \mathbb{R}^{m\times r_u}$ are orthogonal matrices and $\Sigma = \text{diag}(\sigma_1, \sigma_2, \ldots, \sigma_{r_u}) \in \mathbb{R}^{r_u\times r_u}$ with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{r_u} > 0$. Note that $\mathbf{V}$ is called the left singular matrix of $\mathbf{U}$ and $\mathbf{W}$ is called the right singular matrix of $\mathbf{U}$. The columns of $\mathbf{V}$ and $\mathbf{W}$ are called the left singular vectors and the right singular vectors of $\mathbf{U}$, respectively. The diagonal entries σ_i for all $i = 1, 2, \ldots, r_u$ of Σ are knowns as the singular values of $\mathbf{U}$. Then the optimal solution of (2.3) or the POD basis is $\{\mathbf{v}_i\}_{i=1}^k$. The minimum error for approximating the snapshots is given by

$$\sum_{j=1}^{m} \|\mathbf{u}_j - \sum_{i=1}^{k} (\mathbf{u}_j^T \mathbf{v}_i)\mathbf{v}_i\|_2^2 = \sum_{i=k+1}^{r_u} \sigma_i^2. \tag{2.5}$$

Thus, we employ the set $\{\mathbf{v}_i\}_{i=1}^k$ or the first k columns of $\mathbf{V}$ to construct a matrix $\mathbf{V}_k$, which will be used to project the original system to a low dimensional subspace as described in (2.2). More details on POD can be found in [21].

However, the effective dimension reduction of the POD-Galerkin approach is usually restricted to the linear term. In (2.2), computing the nonlinear term still has complexity that depends on n, the dimension of the original full-order system. In particular, to compute

$$\hat{\mathbf{F}}(t, \tilde{\mathbf{u}}(t)) := \underbrace{\mathbf{V}_k^T}_{k\times n} \underbrace{\mathbf{F}(t, \mathbf{V}_k \tilde{\mathbf{u}}(t))}_{n\times 1}, \tag{2.6}$$

it is required to perform matrix-vector multiplication with complexity $\mathcal{O}(kn)$. Solving the system (2.2) is still costly since the evaluation of the nonlinear term requires the full computation. Therefore, we employ another efficient technique to overcome the complexity problem that occurs on the nonlinear term as discussed in Section. 2.2.

2.2 Discrete empirical interpolation method (DEIM)

DEIM [9] is an efficient approach to reduce the complexity for evaluating the nonlinear term. To illustrate this issue, we consider again the nonlinearity in (2.6)

$$\hat{\mathbf{F}}(t, \tilde{\mathbf{u}}(t)) := \underbrace{\mathbf{V}_k^T}_{k\times n} \underbrace{\mathbf{F}(t, \mathbf{V}_k \tilde{\mathbf{u}}(t))}_{n\times 1}.$$

Let $\{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_m\} \subset \mathbb{R}^n$ be the set of the nonlinear snapshots $\mathbf{f}_j = \mathbf{F}(t_j, \mathbf{u}(t_j))$ for all $j = 1, 2, \ldots, m$, where $\mathbf{u}(t_j)$ is already computed from (2.1). Suppose that $\mathcal{F} = \operatorname{span}\{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_m\}$ with $\dim(\mathcal{F}) = r_f$. We denote the nonlinear snapshot matrix with $\bar{\mathbf{F}} = [\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_m] \in \mathbb{R}^{n\times m}$. The SVD is then used on $\bar{\mathbf{F}}$ to find the POD basis of rank $l < r_f$ of the nonlinear term. In particular, assume that the SVD of $\bar{\mathbf{F}}$ is $\bar{\mathbf{F}} = \bar{\mathbf{V}}\bar{\Sigma}\bar{\mathbf{W}}^T$, where $\bar{\mathbf{V}} = [\bar{\mathbf{v}}_1, \bar{\mathbf{v}}_2, \ldots, \bar{\mathbf{v}}_{r_f}] \in \mathbb{R}^{n\times r_f}$, $\bar{\mathbf{W}} = [\bar{\mathbf{w}}_1, \bar{\mathbf{w}}_2, \ldots, \bar{\mathbf{w}}_{r_f}] \in \mathbb{R}^{m\times r_f}$ and $\bar{\Sigma} = \operatorname{diag}(\bar{\sigma}_1, \bar{\sigma}_2, \ldots, \bar{\sigma}_{r_f}) \in \mathbb{R}^{r_f\times r_f}$. Thus, the POD basis of rank l of the nonlinear term is the first l columns of the matrix $\bar{\mathbf{V}}$, denoted by $\bar{\mathbf{V}}_l$. Then the nonlinear function $\mathbf{F}(t, \mathbf{V}_k\tilde{\mathbf{u}}(t))$ can be approximated by a subspace spanned by the basis $\{\bar{\mathbf{v}}_1, \bar{\mathbf{v}}_2, \ldots, \bar{\mathbf{v}}_l\}$, which is of the form

$$\mathbf{F}(t, \mathbf{V}_k \tilde{\mathbf{u}}(t)) \approx \bar{\mathbf{V}}_l \mathbf{c}(t), \tag{2.7}$$

where $\mathbf{c} : \mathcal{D} \to \mathbb{R}^l$ and $\mathbf{c}(t)$ is the corresponding coefficient vector at the time $t \in \mathcal{D}$. The DEIM technique is applied here to specify $\mathbf{c}(t)$ by selecting the l rows of (2.7).

Let $\mathbf{P}$ be a matrix used in the interpolation defined as $\mathbf{P} = [\mathbf{e}_{\wp_1}, \mathbf{e}_{\wp_1}, \ldots, \mathbf{e}_{\wp_l}] \in \mathbb{R}^{n\times l}$, where $\mathbf{e}_{\wp_i} = [0, \ldots, 0, 1, 0, \ldots, 0]^T$ is the $\wp_i$ column of the identity matrix $\mathbf{I}_n \in \mathbb{R}^{n\times n}$ for all $i = 1, 2, \ldots, l$. By multiplying $\mathbf{P}^T$ both sides of Eq. (2.7), the selection of components in the nonlinear term is made as follows

$$\mathbf{P}^T\mathbf{F}(t, \mathbf{V}_k\tilde{\mathbf{u}}(t)) \approx \underbrace{\mathbf{P}^T\bar{\mathbf{V}}_l}_{l\times l}\,\mathbf{c}(t). \tag{2.8}$$

Assume that $\mathbf{P}^T\bar{\mathbf{V}}_l$ is a nonsingular matrix. Then $\mathbf{c}(t)$ can be determined uniquely as $\mathbf{c}(t) \approx (\mathbf{P}^T\bar{\mathbf{V}}_l)^{-1}\mathbf{P}^T\mathbf{F}(t, \mathbf{V}_k\tilde{\mathbf{u}}(t))$. As a result, the final approximation of (2.7) becomes

$$\mathbf{F}(t, \mathbf{V}_k\tilde{\mathbf{u}}(t)) \approx \bar{\mathbf{V}}_l(\mathbf{P}^T\bar{\mathbf{V}}_l)^{-1}\mathbf{P}^T\mathbf{F}(t, \mathbf{V}_k\tilde{\mathbf{u}}(t)). \tag{2.9}$$

The interpolation indices $\wp_1, \wp_2, \ldots, \wp_l$ are generated by the DEIM algorithm shown in Algorithm 1.

Algorithm 1 DEIM

```
Input: $l < r_f$, $\{\bar{\mathbf{v}}\}_{j=1}^{r_f} = \{\bar{\mathbf{v}}_1, \bar{\mathbf{v}}_2, \ldots, \bar{\mathbf{v}}_{r_f}\} \subset \mathbb{R}^n$
Output: $\mathbf{P} \in \mathbb{R}^{n\times l}$, $\vec{\wp}_l = [\wp_1, \wp_2, \ldots, \wp_l]^T \in \mathbb{R}^l$
$[|\rho|, \wp_1] = \max\{|\bar{\mathbf{v}}_1|\}$
$\bar{\mathbf{V}} = [\bar{\mathbf{v}}_1]$, $\bar{\mathbf{P}} = [\mathbf{e}_{\wp_1}]$, $\vec{\wp} = [\wp_1]$
for $j = 2 : r_f$ do
    $\mathbf{c} = (\bar{\mathbf{P}}^T\bar{\mathbf{V}})^{-1}\bar{\mathbf{P}}^T\bar{\mathbf{v}}_j$
    $\mathbf{r} = \bar{\mathbf{v}}_j - \bar{\mathbf{V}}\mathbf{c}$
    $[|\rho|, \wp_j] = \max\{|\mathbf{r}|\}$
    $\bar{\mathbf{V}} \leftarrow [\bar{\mathbf{V}}\ \ \bar{\mathbf{v}}_j]$, $\bar{\mathbf{P}} \leftarrow [\bar{\mathbf{P}}\ \ \mathbf{e}_{\wp_j}]$, $\vec{\wp} \leftarrow \begin{bmatrix} \vec{\wp} \\ \wp_j \end{bmatrix}$
end for
$\mathbf{P} = \bar{\mathbf{P}}(:, 1 : l)$, $\vec{\wp}_l = \vec{\wp}(1 : l)$
```

From Algorithm 1, the procedure constructs a set of indices inductively on the input basis. The process starts from selecting the first interpolation index $\wp_1 \in \{1, 2, \ldots, n\}$ corresponding to the first input basis $\bar{\mathbf{v}}_1$ entry which has the largest magnitude. The remaining indices $\wp_j$ for $j = 2, 3, \ldots, l$ are selected from the entry of the residual $\mathbf{r} = \bar{\mathbf{v}}_j - \bar{\mathbf{V}}\mathbf{c}$ with the largest magnitude. The linear independence of the input basis $\{\bar{\mathbf{v}}_j\}_{j=1}^{l}$ guarantees that in each iteration, $\mathbf{r}$ is a nonzero vector and thus ρ is also nonzero. This implies that $\mathbf{P}^T\bar{\mathbf{V}}_l$ is always nonsingular. Hence, the DEIM procedure is well-defined. This also implies that the interpolation indices $\{\wp_j\}_{j=1}^{r_f}$ are nonrepeated. The output matrix $\mathbf{P}$ is employed to construct a low-dimensional approximation of the nonlinear term. Then the POD technique described in Section. 2.1 is used in conjunction with the DEIM technique to construct a reduced-order system that is completely independent of the full dimension as shown below

$$\frac{d^2}{dt^2}\tilde{\mathbf{u}}(t) = \underbrace{\tilde{\mathbf{A}}}_{k\times k}\underbrace{\tilde{\mathbf{u}}(t)}_{k\times 1} + \underbrace{\mathbf{V}_k^T\bar{\mathbf{V}}_l}_{k\times l}\underbrace{(\mathbf{P}^T\bar{\mathbf{V}}_l)^{-1}}_{l\times l}\underbrace{\mathbf{P}^T\mathbf{F}(t,\mathbf{V}_k\tilde{\mathbf{u}}(t))}_{l\times 1}. \tag{2.10}$$

An error bound between the nonlinear function, $\mathbf{F}(t,\mathbf{u}(t))$ and its DEIM approximation, $\mathbf{F}^{DEIM}(t,\mathbf{u}(t)) := \bar{\mathbf{V}}_l(\mathbf{P}^T\bar{\mathbf{V}}_l)^{-1}\mathbf{P}^T\mathbf{F}(t,\mathbf{u}(t))$ [9] was given by

$$\|\mathbf{F} - \mathbf{F}^{DEIM}\|_2 \le \mathcal{C}\|(\mathbf{I} - \bar{\mathbf{V}}_l\bar{\mathbf{V}}_l^T)\mathbf{F}\|_2, \tag{2.11}$$

where $\mathcal{C} = \|(\mathbf{P}^T\bar{\mathbf{V}}_l)^{-1}\|_2$. To increase the accuracy of the DEIM approximation, the value of $\|(\mathbf{I} - \bar{\mathbf{V}}_l\bar{\mathbf{V}}_l^T)\mathbf{F}\|_2$ should be decreased while the value of $\mathcal{C} = \|(\mathbf{P}^T\bar{\mathbf{V}}_l)^{-1}\|_2$ is still bounded. By increasing the number of left singular vectors in $\bar{\mathbf{V}}_l$, the value of $\|(\mathbf{I} - \bar{\mathbf{V}}_l\bar{\mathbf{V}}_l^T)\mathbf{F}\|_2$ becomes smaller. However, the DEIM algorithm described in Algorithm 1 computes the interpolation matrix $\mathbf{P}$ that can be prevented the growth of the term $\mathcal{C} = \|(\mathbf{P}^T\bar{\mathbf{V}}_l)^{-1}\|_2$ according to [10]. Hence, the accuracy can be improved by increasing more left singular vectors in $\bar{\mathbf{V}}_l$.

In the next section, the POD-DEIM approach will be applied to the full-order discretized system of the sine-Gordon equation.

3 An application of POD and DEIM on the sine-Gordon equation

The sine-Gordon equation plays an important role in many mathematical physics applications such as quantum field theory, Josephson junction and mechanical transmission line. The equation is of the form

$$u_{tt} = u_{xx} - \sin(u), \qquad (x,t) \in [0,L]\times[0,T], \tag{3.1}$$

with the initial conditions: $u(x,0) = f(x)$ and $u_t(x,0) = g(x)$, and the boundary conditions $u(0,t) = h_0(t)$ and $u(L,t) = h_L(t)$.

Finite difference is used to discretize this problem here. We assume that $n+2$ spatial grid points $0 = x_0 < x_1 < \cdots < x_{n+1} = L$ are distributed uniformly with step size $\Delta x = \frac{L}{(n+1)}$. Similarly, the time domain is partitioned uniformly by $0 = t_0 < t_1 < \cdots < t_m = T$ with step size $\Delta t = \frac{T}{m}$. Define the approximation of $u(x,t)$ at (x_i,t_j) by u_{ij} ,i.e., $u_{ij} \approx u(x_i,t_j)$. The second order central difference is used for the spatial discretization to obtain the following matrix form of differential equation

$$\frac{d^2}{dt^2}\mathbf{u}(t) = \frac{1}{\Delta x^2}[\mathbf{A}\mathbf{u}(t) + \mathbf{b}(t)] - \sin(\mathbf{u}(t)), \tag{3.2}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is a constant matrix from finite difference discretization, $\mathbf{b}(t_j) = \mathbf{b}_j = [u(0,t_j),\ldots,0,\ldots,u(L,t_j)]^T \in \mathbb{R}^n$ is a vector computed from the boundary conditions, and $\mathbf{u}_j$ is the approximate solution of $\mathbf{u}(t_j) = [u(x_1,t_j), u(x_2,t_j),$

$\ldots, u(x_n, t_j)]^T \in \mathbb{R}^n$ at the time $t_j = j\Delta t$. Then, we applied the second order central difference discretization with the semi-implicit method, which gives the following update formula:

$$\mathbf{u}_{j+1} \quad = \quad \mathbf{B}\Big[\lambda[\mathbf{b}_{j+1} - (\Delta x^2)\sin(\mathbf{u}_j)] + 2\mathbf{u}_j - \mathbf{u}_{j-1}\Big], \tag{3.3}$$

where $\mathbf{B} = (\mathbf{I}_n - \lambda\mathbf{A})^{-1}$. The formula (3.3) requires to use solutions from the previous two steps, $\mathbf{u}_j$ and $\mathbf{u}_{j-1}$ for updating $\mathbf{u}_{j+1}$. Thus, the two initial conditions are then used to provide $\mathbf{u}_0$ and $\mathbf{u}_1$ for the first iteration. The full-order solutions are then computed and the computation is terminated at the time $T = m\Delta t$. The numerical solutions are denoted by $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m$ and will be called snapshots. Let $\mathbf{U}^s = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m] \in \mathbb{R}^{n\times m}$ be the snapshot matrix. We next apply model reduction technique described earlier to reduce the computational complexity of this full-order discretized sine-Gordon equation.

3.1 Reduced-order modelling using POD technique

Suppose $\mathbf{U}^s = \mathbf{V}^{pod}\Sigma^{pod}\mathbf{W}^{pod^T}$ is the SVD of $\mathbf{U}^s$ with $\text{rank}(\mathbf{U}^s) = \text{rank}(\mathbf{V}^{pod}) = r_{pod}$. Choose the first $k \in \{1, 2, \ldots, r_{pod}\}$ columns of $\mathbf{V}^{pod}$, denoted by $\mathbf{V}_k^{pod} \in \mathbb{R}^{n\times k}$ as the k-dimensional POD basis of the snapshot set $\{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m\}$. Thus, the solution can be approximated by the subspace spanned by the POD basis, $\mathbf{u}(t) \approx \mathbf{V}_k^{pod}\tilde{\mathbf{u}}(t)$, where $\tilde{\mathbf{u}} : [0, T] \to \mathbb{R}^k$. By substituting $\mathbf{u}(t) \approx \mathbf{V}_k^{pod}\tilde{\mathbf{u}}(t)$ into the full-order model (3.2) and applying the Galerkin projection as described in the previous section, we obtain the following POD reduced system

$$\frac{d^2}{dt^2}\tilde{\mathbf{u}}(t) = \frac{1}{\Delta x^2}[\tilde{\mathbf{A}}\tilde{\mathbf{u}}(t) + \mathbf{V}_k^{pod^T}\mathbf{b}(t)] - \mathbf{V}_k^{pod^T}\sin(\mathbf{V}_k^{pod}\tilde{\mathbf{u}}(t)), \tag{3.4}$$

where $\tilde{\mathbf{A}} = \mathbf{V}_k^{pod^T}\mathbf{A}\mathbf{V}_k^{pod}$. The corresponding update formula for the POD reduced system can be obtained by using the second order central difference on the time domain with semi-implicit time integration as shown below:

$$\tilde{\mathbf{u}}_{j+1} \quad = \quad \tilde{\mathbf{B}}\Big[\lambda[\tilde{\mathbf{b}}_{j+1} - (\Delta x^2)\underbrace{\mathbf{V}_k^{pod^T}\sin(\mathbf{V}_k^{pod}\tilde{\mathbf{u}}_j)]}_{\text{nonlinear term}} + 2\tilde{\mathbf{u}}_j - \tilde{\mathbf{u}}_{j-1}\Big], \tag{3.5}$$

where $\tilde{\mathbf{B}} = (\mathbf{I}_n - \lambda\tilde{\mathbf{A}})^{-1}$, $\tilde{\mathbf{b}}_j = \mathbf{V}_k^{pod^T}\mathbf{b}_j$ with $\tilde{\mathbf{u}}_0 = \mathbf{V}_k^{pod^T}\mathbf{u}_0$ and $\tilde{\mathbf{u}}_1 = \mathbf{V}_k^{pod^T}\mathbf{u}_1$ for updating at the first iteration. To obtain the POD approximation, the reduced variable $\tilde{\mathbf{u}}_j$ which obtained from (3.5) for each iteration is projected back to the solution space. Therefore, the POD approximate solution is $\mathbf{u}_j^{pod} = \mathbf{V}_k^{pod}\tilde{\mathbf{u}}_j$, for all $j = 2, 3, \ldots, m$.

3.2 Complexity reduction on the nonlinear term

To apply DEIM, we first define $\mathbf{F}^n(t, \mathbf{u}(t)) = \sin(\mathbf{u}(t))$. Then we construct the nonlinear snapshot matrix $\bar{\mathbf{F}}^n = [\mathbf{F}^n(t_1, \mathbf{u}(t_1)), \mathbf{F}^n(t_2, \mathbf{u}(t_2)), \ldots, \mathbf{F}^n(t_m, \mathbf{u}(t_m))] \in$

$\mathbb{R}^{n\times m}$. Note that the evaluation of the nonlinear function at each time $t_j = j\Delta t$ for all $j = 1, 2, \ldots, m$ is a by-product which is obtained during the computation of the full-order model. As discussed in Section 2.2, we first compute the SVD of $\bar{\mathbf{F}}^n$. Assume $\bar{\mathbf{F}}^n = \mathbf{V}^{deim}\Sigma^{deim}\mathbf{W}^{deim^T}$ is the SVD of $\bar{\mathbf{F}}^n$ with $\text{rank}(\bar{\mathbf{F}}^n) = \text{rank}(\mathbf{V}^{deim}) = r_{deim}$. Choose the first $l \in \{1, 2, \ldots, r_{deim}\}$ columns of $\mathbf{V}^{deim}$, denoted by $\mathbf{V}_l^{deim} \in \mathbb{R}^{n\times l}$ as the l-dimensional POD basis of nonlinear snapshots. The POD basis of nonlinear snapshots is used as an input in the DEIM algorithm for generating the interpolation matrix $\mathbf{P} \in \mathbb{R}^{n\times l}$ as described in Algorithm 1. Thhe DEIM approximation of the nonlinear function is given by

$$\mathbf{F}^n(t, \mathbf{V}_k^{pod}\tilde{\mathbf{u}}(t)) \approx \Phi^{deim}\mathbf{P}^T\mathbf{F}^n(t, \mathbf{V}_k^{pod}\tilde{\mathbf{u}}(t)), \tag{3.6}$$

where $\Phi^{deim} = \mathbf{V}_l^{deim}(\mathbf{P}^T\mathbf{V}_l^{deim})^{-1}$ and the approximation of the projected nonlinear term in the POD reduced system becomes

$$\hat{\mathbf{F}}^n(t, \tilde{\mathbf{u}}(t)) \approx \mathbf{V}_k^{pod^T}\Phi^{deim}\mathbf{P}^T\mathbf{F}^n(t, \mathbf{V}_k^{pod}\tilde{\mathbf{u}}(t)). \tag{3.7}$$

By substituting (3.7) for the nonlinear term of (3.4), we obtain the following POD-DEIM reduced system:

$$\frac{d^2}{dt^2}\tilde{\mathbf{u}}(t) = \frac{1}{\Delta x^2}[\tilde{\mathbf{A}}\tilde{\mathbf{u}}(t) + \mathbf{V}_k^{pod^T}\mathbf{b}(t)] - \mathbf{V}_k^{pod^T}\Phi^{deim}\mathbf{P}^T\mathbf{F}^n(t, \mathbf{V}_k^{pod}\tilde{\mathbf{u}}(t)). \tag{3.8}$$

Finally, the model of the sine-Gordon equation can be solved entirely in the reduced dimension with no dependence on the original dimension by using the following iterative formula

$$\tilde{\mathbf{u}}_{j+1} = \tilde{\mathbf{B}}\left[\lambda[\tilde{\mathbf{b}}_{j+1} - (\Delta x^2)\mathbf{V}_k^{pod^T}\Phi^{deim}\mathbf{P}^T\sin(\mathbf{V}_k^{pod}\tilde{\mathbf{u}}_j)] + 2\tilde{\mathbf{u}}_j - \tilde{\mathbf{u}}_{j-1}\right], \tag{3.9}$$

with $\tilde{\mathbf{u}}_0 = \mathbf{V}_k^{pod^T}\mathbf{u}_0$ and $\tilde{\mathbf{u}}_1 = \mathbf{V}_k^{pod^T}\mathbf{u}_1$. As for the POD reduced system, the POD-DEIM approximate solution is given by $\mathbf{u}_j^{poddeim} = \mathbf{V}_k^{pod}\tilde{\mathbf{u}}_j$, for all $j = 2, 3, \ldots, m$.

In the next section, the numerical experiments compared between POD and POD-DEIM approximate solutions and the full-order solution are presented. These numerical results demonstrate the efficiency of the POD and POD-DEIM approaches through the substantial reduction in simulation time while maintaining accuracy of the approximate solution.

4 Numerical results

In this section, the numerical results from the POD-DEIM approach are analyzed in three different ways. In Section 4.1, both POD and DEIM are shown to be efficient in decreasing both computational memory storage and simulation time. In Section 4.2, the accuracy in approximating the solutions when the reduced models are constructed with different amount of snapshots is investigated. The POD-DEIM approach is also shown to accurately approximate the solutions that may not be corresponding to the time instances used in the snapshots for constructing the POD basis. In Section 4.3, the POD-DEIM approach is demonstrated to be applicable for parametrized sine-Gordon equation with different parameter values.

4.1 Numerical test I

This section considers the full-order model with spatial domain $[0, L]$, where $L = 10$ with dimension of spatial grid points $n = 499$ and time interval $[0, T]$, where $T = 10$ with number of time steps $m = 500$, which give $\Delta x = \frac{L}{n+1} = \frac{10}{500} = 0.02$ and $\Delta t = \frac{T}{m} = \frac{10}{500} = 0.02$. Here, the equation (3.1) is solved subject to the initial conditions

$$u(x, 0) \;=\; 0, \qquad u_t(x, 0) \;=\; 4\,\mathrm{sech}(x), \tag{4.1}$$

and the boundary conditions

$$u(0, t) \;=\; 4\arctan(\mathrm{sech}(0)t), \quad u(10, t) \;=\; 4\arctan(\mathrm{sech}(10)t). \tag{4.2}$$

Note that the sine-Gordon equation subjected to the initial conditions and the boundary conditions given above has the following exact solution [2]

$$u(x, t) = 4\arctan(\mathrm{sech}(x)t). \tag{4.3}$$

The result from the full-order system will be compared to both POD and POD-DEIM systems. To investigate accuracy, we first construct the solution matrix from the solution at every time step $j\Delta t$ for all $j = 0, 1, \ldots, m$ combined with the boundary conditions. The solution matrices of the full-order, POD and POD-DEIM models are denoted by $\mathbf{U}^{full}$, $\mathbf{U}^{pod}$ and $\mathbf{U}^{poddeim} \in \mathbb{R}^{(n+2)\times(m+1)}$, respectively. We define an error between the full-order solution and the POD approximate solution as

$$\mathrm{E}_{pod} = \sqrt{\sum_{j=0}^{m} \|\mathbf{u}_j - \mathbf{u}_j^{pod}\|_2^2} = \|\mathbf{U}^{full} - \mathbf{U}^{pod}\|_F. \tag{4.4}$$

In the same way, an error between the full-order solution and the POD-DEIM approximate solution is defined as

$$\mathrm{E}_{poddeim} = \sqrt{\sum_{j=0}^{m} \|\mathbf{u}_j - \mathbf{u}_j^{poddeim}\|_2^2} = \|\mathbf{U}^{full} - \mathbf{U}^{poddeim}\|_F. \tag{4.5}$$

The maximum error for matrix used in this analysis $|\cdot|_m : \mathbb{R}^{(n+2)\times(m+1)} \to \mathbb{R}$ is defined by

$$|\mathbf{U}|_m = \max_{i,j} |u_{i,j}|, \tag{4.6}$$

where $\mathbf{U} = [u_{i,j}] \in \mathbb{R}^{(n+2)\times(m+1)}$, which returns the maximum absolute value of elements in the matrix. Note that the first two columns at time t_0 and t_1 of the solution matrices are provided from the initial conditions (4.1).

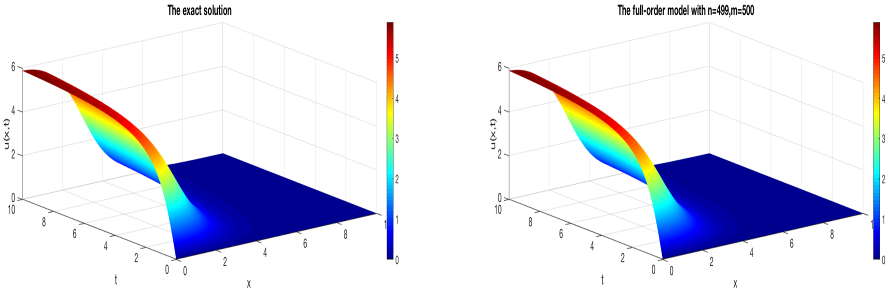

Figure 1: [Numerical test I] Left: The exact solution (4.3) of the sine-Gordon equation which is subjected the initial conditions (4.1). Right: The corresponding numerical solution.

The solutions $\mathbf{u}_j$ for all $j = 1, 2, \dots, m$ are arranged to construct a snapshot matrix. The SVD of snapshot matrix is then computed to determine the POD basis of snapshots. In Figure 2, the fast decay occurs around the first 90 singular values of 500 snapshots which implies that the first 90 left singular vectors can be used to represent the dominant characteristic of the whole set of snapshots.

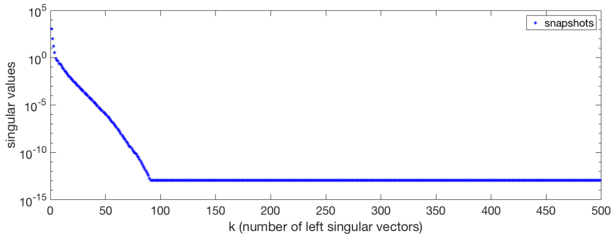

Figure 2: [Numerical test I] Singular values of 500 snapshots.

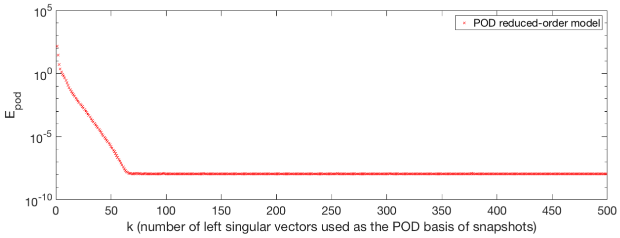

Figure 3: [Numerical test I] The number of left singular vectors used as the POD basis of snapshots is plotted against $\mathbf{E}_{pod}$ defined in (4.4) for $k = 1, 2, \dots, 499$.

In this section, the first $k = 20, 50, 60, 65, 70, 95, 100$ columns of the left singular matrix are chosen as the dimensions of different POD basis sets. The values of k will be considered as the POD reduced dimension. Here, the POD-Galerkin approach is applied as described in Section 3.1 for each reduced dimension. The POD approximate solution is obtained and compared with the full-order solution (Figure 1). The resulting solution of each POD reduced model and its corresponding absolute error for every component compared with the full-order solution are plotted as shown in Figure 4. Since we hardly see the difference between the full-order solution (Figure 1) and the POD approximate solution (Figure 4), we provide the maximum absolute error $|\mathbf{U}^{full} - \mathbf{U}^{pod}|_m$ defined in (4.6) and $\mathbf{E}_{pod}$ defined in (4.4) to demonstrate the accuracy as shown in Table 1.

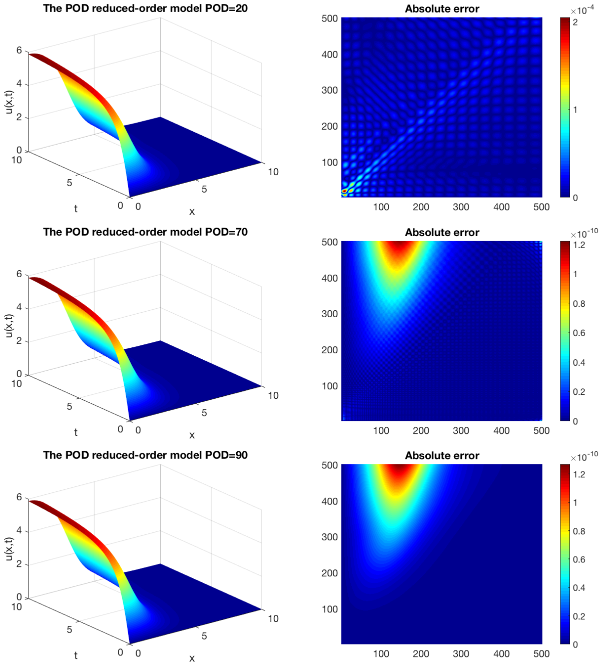

Figure 4: [Numerical test I] The POD approximation using different POD reduced dimensions $k = 20, 70, 90$ and the corresponding absolute error for every component compared with the full-order solution.

Figure 3 illustrates the decreasing of the error when the number of left singular vectors increase. This implies that larger number of left singular vectors can capture solution space more accurate. The decreasing of error reached a plateau when $k = 70$. As a result, $k = 70$ is chosen as an appropriate reduced dimension for POD. However, the CPU time can be further reduced by using the DEIM procedure, so that we fixed the POD reduced dimension $k = 70$ for constructing the POD-DEIM reduced-order model later.

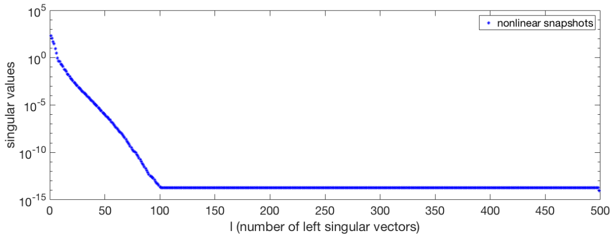

Figure 5: [Numerical test I] All nonzero singular values of 500 nonlinear snapshots.

For nonlinear snapshots, the decay of singular values is considered in a similar way as described before. Since the decreasing of singular values stops at $l = 100$ as shown in Figure 5, we will consider the DEIM reduced dimension when $l < 100$. In particular, we choose different DEIM reduced dimensions $l = 20, 40, 50, 60, 65, 75$ in this analysis (See Table 1). The fixed POD reduced dimension $k = 70$ is used in conjunction with these chosen DEIM reduced dimensions. The POD-DEIM reduced-order system is constructed as described in Section 3.2. In Figure 6, the resulting solution of POD-DEIM reduced model is shown for each reduced dimension in the same manner as for the case of POD reduced model.

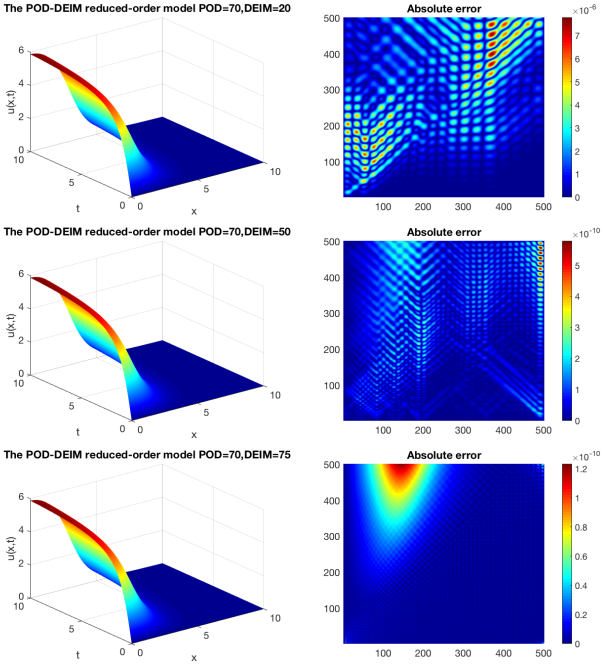

Figure 6: [Numerical test I] The POD-DEIM approximation for fixed POD reduced dimension $k = 70$ using different DEIM reduced dimensions $l = 20, 50, 75$ and the corresponding absolute error for every component compared with the full-order solution.

From Figure 7 (bottom), since there is no difference between the error of $k = 70$ and $k = 90$, then $k = 70$ should be used as the POD reduced dimension. To choose a proper DEIM reduced dimension, we consider the results in Table 1. Notice that, for $k = 70$, the decreasing of the error stops after using DEIM dimension $l = 50$. I.e., the POD-DEIM reduced systems with $k = 70$ and $l \geq 50$ have the same order of both errors defined in (4.6) and (4.5) as the POD reduced system with $k = 70$. Therefore, to obtain the most accurate approximation from the POD-DEIM approach with the minimum memory storage and CPU time, the smallest dimensions for POD and DEIM are $k = 70$ and $l = 50$, respectively.

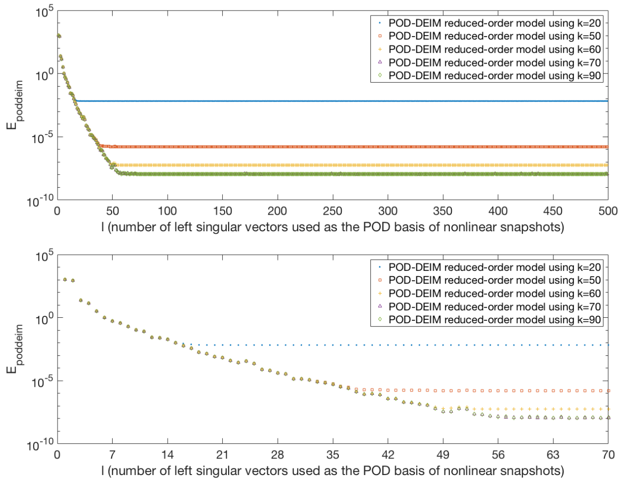

Figure 7: [Numerical test I] The number of left singular vectors used as the POD basis of nonlinear snapshots is plotted against $\mathbf{E}_{poddeim}$ defined in (4.5) for each $k = 20, 50, 60, 70, 90$ belong to $l = 1, 2, \ldots, 499$ (top) and $l = 1, 2, \ldots, 70$ (bottom).

	k	l	Scaled CPU time usage	$\lvert\mathbf{U}^{full} - \mathbf{U}^{pod}\rvert_m$ defined in (4.6)	$\mathbf{E}_{pod}$ defined in (4.4)
full-order	-	-	1	-	-
POD	20	-	0.0263223464	2.0497×10^{-4}	6.7361×10^{-3}
POD	50	-	0.0489276980	7.7867×10^{-8}	1.6956×10^{-6}
POD	60	-	0.0527012237	2.7650×10^{-9}	5.5912×10^{-8}
POD	65	-	0.0641294614	3.4197×10^{-10}	1.4452×10^{-8}
POD	70	-	0.0708348793	1.2223×10^{-10}	1.0977×10^{-8}
POD	90	-	0.0869485303	1.2693×10^{-10}	1.1793×10^{-8}
POD	100	-	0.0926408364	1.2515×10^{-10}	1.1569×10^{-8}
	k	l	Scaled CPU time usage	$\lvert\mathbf{U}^{full} - \mathbf{U}^{poddeim}\rvert_m$ defined in (4.6)	$\mathbf{E}_{poddeim}$ defined in (4.5)
full-order	-	-	1	-	-
POD-DEIM	70	20	0.0337271805	7.7991×10^{-6}	8.4465×10^{-4}
POD-DEIM	70	40	0.0357376960	9.8190×10^{-9}	9.1697×10^{-7}
POD-DEIM	70	50	0.0373398702	5.7891×10^{-10}	3.8234×10^{-8}
POD-DEIM	70	60	0.0386180176	1.3413×10^{-10}	1.2279×10^{-8}
POD-DEIM	70	65	0.0398271947	1.2461×10^{-10}	1.1126×10^{-8}
POD-DEIM	70	75	0.0401167150	1.2329×10^{-10}	1.1011×10^{-8}

Table 1: [Numerical test I] The POD and POD-DEIM reduced-order systems are shown the simulation time and compared its accuracy with the full-order system.

However, smaller dimensions of POD and DEIM can be used in practice for more efficient simulation time with certain acceptable trade off in accuracy. Note that the simulation time is averaged from 5 times computation of each case and this manner will be applied throughout the numerical results.

4.2 Numerical test II

We employ the full-order system which is defined as in previous section. The corresponding full-order solution is collected at the time $t = j\Delta t$ for $j = 1, 2, \ldots, m$, where $\Delta t = \frac{T}{m} = \frac{10}{500} = 0.02$. In this section, we construct POD basis from different amount of snapshots that are solved from the full-order system with different time-step sizes. Here, a smaller set of snapshots is obtained from computing the full-order system at the time $t = j\Delta t'$ for $j = 1, 2, \ldots, m'$, where $\Delta t' = \frac{T}{m'}$ with $m' < m$. I.e., the snapshots are computed from the full-order system with larger time step size than the one used in the reduced system, $\Delta t' > \Delta t$. In this section, the POD reduced dimension k will be variously considered. To compare the accuracy for each case of m', we will consider two criteria for selecting k,

(1) we will use the retained energy [6] of the snapshots set for unbiased choice of reduced dimension, which is defined as

$$I(e) = \frac{\sum_{i=1}^{e} \sigma_i}{\sum_{i=1}^{m'} \sigma_i}, \tag{4.7}$$

and choose $k = \text{argmin}\{e : I(e) \geq \gamma\}$, where $\gamma \in [0, 1]$ is the ratio of information captured by the subspace which is spanned by k-dimensional POD basis.

(2) we will use the fixed reduced dimension for every case of m' to maintain the computational complexity in the comparison.

In this numerical test, we construct the POD basis from different number of snapshots $m' = 200, 250, 300, 400$, i.e., the snapshots are taken from the numerical solution of the full-order system with time step $\Delta t' = 0.05, 0.04, 0.033, 0.025$.

	m'	$\Delta t'$	CPU time usage
Full-order	200	0.05	1.0399395420
	250	0.04	1.2768116360
	300	0.033	1.5967292390
	400	0.025	1.9375945300
	500	0.02	2.3625515580

Table 2: [Numerical test II] CPU time usage of the full-order system for each number of snapshots $m' = 200, 250, 300, 400, 500$.

As shown in Table 2, it is obvious that the simulation time becomes lower when we use larger step size $\Delta t'$, which will give smaller number of snapshots. This means that constructing the reduced-order model with fewer snapshots used less computational cost and memory storage. In Figure 8, the singular values are plotted for each case of different number of snapshots m'. We observe that the decreasing of singular values reached a plateau when $k \geq 85$ for every case as shown in Figure 8 (bottom). Therefore, the POD reduced dimension $k^* = 85$ will be used later in this numerical test for the POD-DEIM procedure. The retained energy defined in (4.7) is computed for different number of singular value k as shown in Figure 9. Notice that the retained energy approaches 1 very quickly when k is very small. As a result, using small γ, e.g. $\gamma = 0.5, 0.6, 0.8$, gives almost the same k for all cases of m'. This is equivalent to the comparison when the dimension k is fixed, which will be investigated later in this section. Therefore, we will use γ close to 1 so that we get different dimension k for different amount of m'.

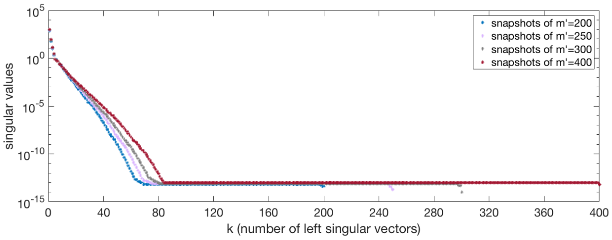

Figure 8: [Numerical test II] All nonzero singular values for each number of snapshot $m' = 200, 250, 300, 400$.

m'	$\Delta t'$	$\gamma = 0.99999$	$\gamma = 0.9999999$	$\gamma = 0.9999999999$
200	0.05	$k = 18$	$k = 29$	$k = 43$
250	0.04	$k = 19$	$k = 31$	$k = 46$
300	0.033	$k = 19$	$k = 32$	$k = 50$
400	0.025	$k = 20$	$k = 35$	$k = 55$

Table 3: [Numerical test II] The POD reduced dimensions corresponding to the ratio of captured information using $\gamma = 0.99999, 0.9999999, 0.9999999999$.

Particularly, the three specific choices of $\gamma = 0.99999, 0.9999999, 0.9999999999$ are used in this numerical test. The POD reduced dimensions are chosen corresponding to these choices for each number of snapshots as shown in Table 3. The resulting POD reduced-order models are used to approximate the solutions at each time $j\Delta t = j(0.02)$ for $j = 2, 3, \dots, 500$ and then compared with the full-order solution. The results in Table 4 demonstrate the effect of the number of snapshots on the accuracy of the reduced-order model. Notice that, for each γ close to 1 in Table 4, there is no significant difference in accuracy when using different m'. As γ gets closer to 1, the errors significantly decrease for all cases of m'. Notice also that, for a fixed values of γ, as the number of m' increases, the dimension k slightly increases and therefore it will have a little more numerical complexity. In the case of fixed $k = k^* = 85$, all cases of m' have the same order of accuracy, which implies that there is almost no effect of the number of snapshots on the approximation of the POD reduced system.

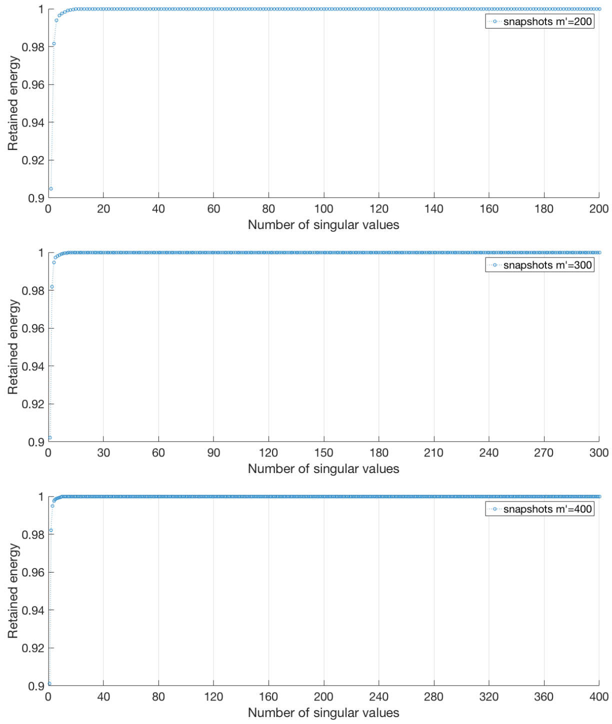

Figure 9: [Numerical test II] The retained energy defined in (4.7) for each number of snapshots $m' = 200, 300, 400$.

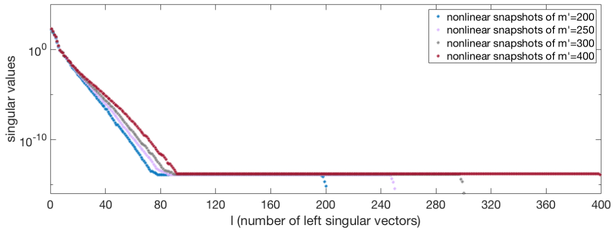

Figure 10: [Numerical test II] All nonzero singular values for each number of nonlinear snapshots $m' = 200, 250, 300, 400$.

For nonlinear snapshots, all nonzero singular values are plotted for different number of nonlinear snapshots m' in Figure 10. We notice that the decreasing of singular values stops at $l = 95$ for every case of m'. We therefore fixed $l = l^* = 95$ for the DEIM reduced dimension in this analysis. The retained energy for nonlinear snapshots is computed as shown in Figure 11. The same three choices of the ratio of information captured identify the reduced dimensions for DEIM as illustrated in Table 5. The fixed POD reduced dimension $k = k^* = 85$ is used in conjunction with these DEIM reduced dimensions. The results of the POD-DEIM reduced-order model are demonstrated for each case in Table 6. The accuracy improvement of the POD-DEIM approximation increasingly grows up for larger value of ratio of information captured on nonlinear snapshots. In the case when $l = l^* = 95$, the maximum absolute error and the error in Frobenius norm become smaller than the order $\mathcal{O}(10^{-9})$ and $\mathcal{O}(10^{-7})$, respectively for all cases of m'. That is, there is no significant difference in accuracy of the POD-DEIM reduced systems construct from various number of snapshots obtained from full-order systems with different step size $\Delta t'$. In addition, we use certain small dimensions for POD and DEIM to further reduce the CPU time when solving the POD-DEIM approximation for all cases of m'. Both POD and DEIM reduced dimensions are set to be the same, which are equal to $5, 10, 15, 20$. The POD-DEIM approximation from the case of smallest

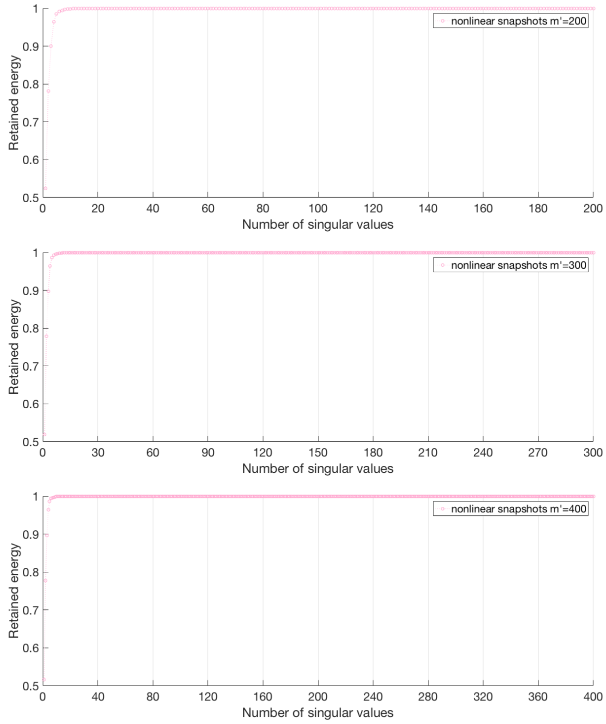

Figure 11: [Numerical test II] The retained energy defined in (4.7) for each number of nonlinear snapshots $m' = 200, 300, 400$.

number of snapshots $m' = 200$ are demonstrated in Figure 12. As illustrated in Table 7, when using $k = 20$ and $l = 20$, the maximum absolute error and the error in Frobenius norm become the order $\mathcal{O}(10^{-4})$ and $\mathcal{O}(10^{-2})$, respectively for every case of m'. These reduced-order model can give reasonable accurate approximation, while using very low memory storage and small simulation time. As a result, in practice, to efficiently reduce the simulation time and memory storage, it is possible to use small dimensions of POD and DEIM (e.g. $k = l = 20$) with certain acceptable trade off on accuracy.

m'	$\Delta t'$	$\gamma = 0.99999$	$\lvert\mathbf{U}^{full} - \mathbf{U}^{pod}\rvert_m$ defined in (4.6)	$\mathbf{E}_{pod}$ defined in (4.4)
200	0.05	$k = 18$	7.3666×10^{-4}	7.0872×10^{-2}
250	0.04	$k = 19$	7.3749×10^{-4}	7.1085×10^{-2}
300	0.033	$k = 19$	8.3870×10^{-4}	9.0115×10^{-2}
400	0.025	$k = 20$	5.0956×10^{-4}	5.2898×10^{-2}
m'	$\Delta t'$	$\gamma = 0.9999999$	$\lvert\mathbf{U}^{full} - \mathbf{U}^{pod}\rvert_m$ defined in (4.6)	$\mathbf{E}_{pod}$ defined in (4.4)
200	0.05	$k = 29$	2.1145×10^{-4}	1.2768×10^{-2}
250	0.04	$k = 31$	1.6986×10^{-4}	6.9289×10^{-3}
300	0.033	$k = 32$	1.1377×10^{-4}	4.2463×10^{-3}
400	0.025	$k = 35$	3.9855×10^{-5}	1.4465×10^{-3}
m'	$\Delta t'$	$\gamma = 0.9999999999$	$\lvert\mathbf{U}^{full} - \mathbf{U}^{pod}\rvert_m$ defined in (4.6)	$\mathbf{E}_{pod}$ defined in (4.4)
200	0.05	$k = 43$	1.5398×10^{-6}	6.6238×10^{-5}
250	0.04	$k = 46$	1.2912×10^{-6}	5.1901×10^{-5}
300	0.033	$k = 50$	1.4166×10^{-6}	5.9415×10^{-5}
400	0.025	$k = 55$	7.3285×10^{-7}	2.9622×10^{-5}
m'	$\Delta t'$	Fixed $k = k^*$	$\lvert\mathbf{U}^{full} - \mathbf{U}^{pod}\rvert_m$ defined in (4.6)	$\mathbf{E}_{pod}$ defined in (4.4)
200	0.05	$k = 85$	1.8069×10^{-10}	9.9349×10^{-8}
250	0.04		3.3215×10^{-10}	2.3069×10^{-8}
300	0.033		1.8854×10^{-10}	1.2474×10^{-8}
400	0.025		1.1379×10^{-10}	1.0195×10^{-8}

Table 4: [Numerical test II] Errors of approximate solutions from the POD reduced-order systems using different POD reduced dimensions follow from Table 3 and fixed $k = k^* = 85$ compared with the full-order system.

m'	$\Delta t'$	$\gamma = 0.99999$	$\gamma = 0.9999999$	$\gamma = 0.9999999999$
200	0.05	$l = 21$	$l = 31$	$l = 46$
250	0.04	$l = 22$	$l = 34$	$l = 50$
300	0.033	$l = 23$	$l = 35$	$l = 53$
400	0.025	$l = 24$	$l = 39$	$l = 59$

Table 5: [Numerical test II] The DEIM reduced dimensions corresponding to the ratio of captured information using $\gamma = 0.99999, 0.9999999, 0.9999999999$.

4.3 Numerical test III

In this numerical test, we consider a parametrized sine-Gordon equation (3.1) in the form

$$u_{tt} = c^2 u_{xx} - p\sin(u), \qquad (x,t) \in [0,L] \times [0,T]. \tag{4.8}$$

m'	$\Delta t'$	Fixed $k = k^*$	$\gamma = 0.99999$	$\lvert\mathbf{U}^{full} - \mathbf{U}^{poddeim}\rvert_m$ defined in (4.6)	$\mathbf{E}_{poddeim}$ defined in (4.5)
200	0.05	$k = 85$	$l = 21$	3.8947×10^{-4}	1.8872×10^{-2}
250	0.04		$l = 22$	3.3339×10^{-4}	1.3124×10^{-2}
300	0.033		$l = 23$	6.3457×10^{-5}	5.5081×10^{-3}
400	0.025		$l = 24$	2.6867×10^{-5}	2.6684×10^{-3}
m'	$\Delta t'$	Fixed $k = k^*$	$\gamma = 0.9999999$	$\lvert\mathbf{U}^{full} - \mathbf{U}^{poddeim}\rvert_m$ defined in (4.6)	$\mathbf{E}_{poddeim}$ defined in (4.5)
200	0.05	$k = 85$	$l = 31$	9.0562×10^{-6}	7.0288×10^{-2}
250	0.04		$l = 34$	3.5661×10^{-6}	3.3561×10^{-4}
300	0.033		$l = 35$	1.5482×10^{-6}	1.5579×10^{-4}
400	0.025		$l = 39$	5.9167×10^{-7}	4.8216×10^{-5}
m'	$\Delta t'$	Fixed $k = k^*$	$\gamma = 0.9999999999$	$\lvert\mathbf{U}^{full} - \mathbf{U}^{poddeim}\rvert_m$ defined in (4.6)	$\mathbf{E}_{poddeim}$ defined in (4.5)
200	0.05	$k = 85$	$l = 46$	1.5001×10^{-7}	1.2439×10^{-5}
250	0.04		$l = 50$	6.0529×10^{-8}	5.8603×10^{-6}
300	0.033		$l = 53$	7.3806×10^{-8}	8.5310×10^{-6}
400	0.025		$l = 59$	1.4729×10^{-9}	1.5620×10^{-6}
m'	$\Delta t'$	Fixed $k = k^*$	Fixed $l = l^*$	$\lvert\mathbf{U}^{full} - \mathbf{U}^{poddeim}\rvert_m$ defined in (4.6)	$\mathbf{E}_{poddeim}$ defined in (4.5)
200	0.05	$k = 85$	$l = 95$	1.8505×10^{-9}	1.0831×10^{-7}
250	0.04			3.3079×10^{-10}	1.9978×10^{-8}
300	0.033			1.8871×10^{-10}	2.4256×10^{-8}
400	0.025			1.1968×10^{-10}	1.3826×10^{-8}

Table 6: [Numerical test II] Errors of the approximate solutions from the POD-DEIM reduced-order systems for fixed POD reduced dimension $k = k^* = 85$ using different DEIM reduced dimensions and fixed $l = l^* = 95$ compared with the full-order system.

m'	$\Delta t'$	k	l	CPU time usage	$\lvert\mathbf{U}^{full} - \mathbf{U}^{poddeim}\rvert_m$ defined in (4.6)	$\mathbf{E}_{poddeim}$ defined in (4.5)
200	0.05	5	5	0.0277420770	5.2688×10^{-2}	6.2648
		10	10	0.0512592750	1.9741×10^{-2}	2.1134
		15	15	0.0649844400	3.2032×10^{-3}	2.6594×10^{-1}
		20	20	0.0679326750	4.1554×10^{-4}	4.0110×10^{-2}
250	0.04	5	5	0.0272968550	4.9916×10^{-2}	6.6039
		10	10	0.0594821610	1.4759×10^{-2}	1.5825
		15	15	0.0684177780	3.1629×10^{-3}	3.0072×10^{-1}
		20	20	0.0681456620	4.7816×10^{-4}	5.2400×10^{-2}
300	0.033	5	5	0.0280498630	3.6468×10^{-2}	4.1214
		10	10	0.0595894640	1.1449×10^{-2}	1.1365
		15	15	0.0624165550	3.4840×10^{-3}	3.7836×10^{-1}
		20	20	0.0614848950	6.9379×10^{-4}	7.1401×10^{-2}
400	0.025	5	5	0.0280621210	3.4785×10^{-2}	3.6695
		10	10	0.0546708950	4.1025×10^{-3}	4.3670×10^{-1}
		15	15	0.0589778150	1.8105×10^{-3}	$2.0304 \times 10 - 1$
		20	20	0.0643772950	5.3837×10^{-4}	6.1373×10^{-2}

Table 7: [Numerical test II] Errors of the approximate solutions from the POD-DEIM reduced-order systems constructed from each number of snapshots $m' = 200, 250, 300, 400$ compared with the full-order system and the corresponding simulation time.

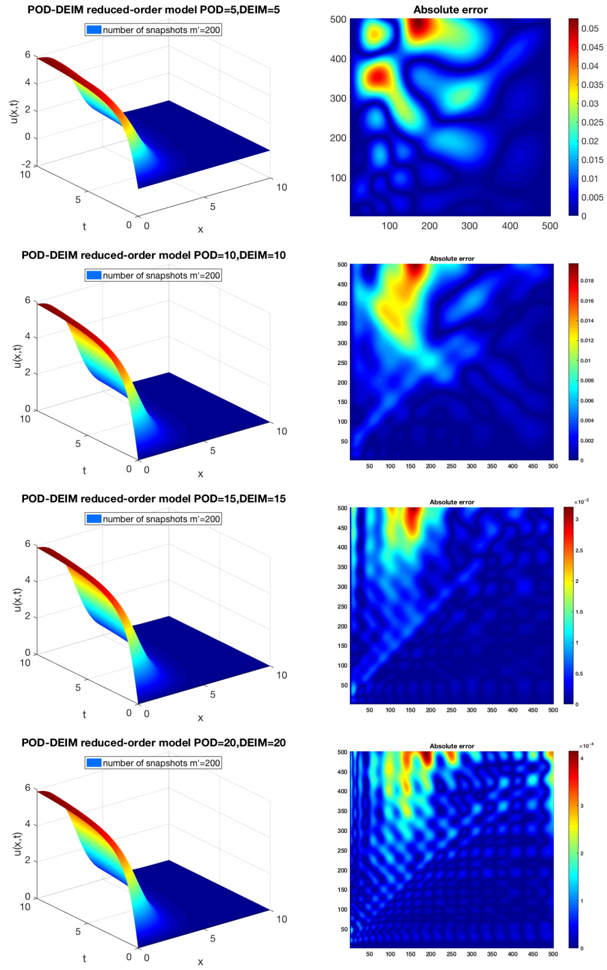

Figure 12: [Numerical test II] The POD-DEIM approximation in the case of $m' = 200$ using POD and DEIM reduced dimensions $k = l = 5, 10, 15, 20$ and the corresponding absolute error for every component compared with the full-order solution.

It is obvious that if $c^2 = 1$ and $p = 0$, the above equation (4.8) becomes a simple hyperbolic partial differential equation. When $c^2 = 1$ and $p = 1$, it is just the same as the sine-Gordon equation (3.1). Rosales [38] in 2001 derived the sine-Gordon equation for torsion coupled pendulums using the continuum modelling techniques and then obtained the equation (4.8) for the continuum limit, where $c = \sqrt{\frac{K}{\rho L^2}}$ is a wave propagation speed, $p = \omega^2 = \sqrt{\frac{K}{L}}$ is the pendulum angular frequency, K is a constant depending on axle material, L is the distance of attached mass of center of mass and ρ is the mass density along the rod.

In this numerical test, we fix $c^2 = 1$ and vary the value of p. Here, we construct a reduced-order model using the POD basis sets from solution snapshots and nonlinear snapshots of the full-order model with $p = 1$, which is the full-order model given in (3.2). Then, these POD basis sets will be used to construct the POD-DEIM reduced models with different parameter p. In this section, we use the following initial conditions and the boundary conditions

$$
\begin{aligned}
u(x,0) &= 1 - e^{-t}, \quad u_t(x,0) \;=\; 0, & (4.9)\\
u(0,t) &= 0, \quad u(L,t) \;=\; 0, & (4.10)
\end{aligned}
$$

which are defined on spatial domain $[0, L]$, where $L = 10$ with dimension of spatial grid points $n = 499$ and time interval $[0, T]$, where $T = 4$ with number of time steps $m = 200$, that is, $\Delta x = \frac{L}{n+1} = \frac{10}{500} = 0.02$ and $\Delta t = \frac{T}{m} = \frac{4}{200} = 0.02$. The full-order solution (3.2) is computed and used as snapshots for constructing the POD basis as shown below.

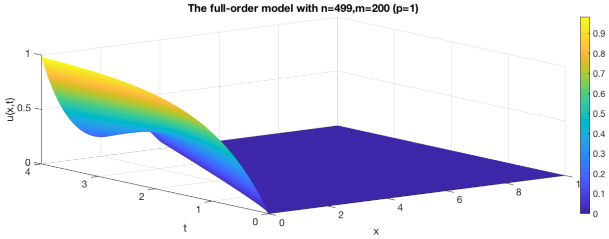

Figure 13: [Numerical test III] The full-order solution of the full-order system (3.2).

For this analysis, we use six specific examples of parameter p, which are $p = 0.1, 0.5, 0.7,\ 1.3, 1.5, 1.9$. The full-order solutions of (4.1) corresponding to these parameter values $p = 0.1, 0.7, 1.3, 1.9$ are shown in Figure 14 and Figure 15. The resulting solution of the parametrized full-order system will be used to compare with its corresponding reduced-order system later.

The SVD of snapshots which are obtained from (3.2) is computed. Figure 16 shows that the first 75 left singular vectors can be represented the dominant characteristic of the whole set of snapshots. The values $k = 20, 50, 60, 70$ are then chosen as the POD reduced dimension. The resulting solution from each parametrized reduced-order system is compared with its corresponding full-order solution. Table 8 and Table 9 illustrate the maximum absolute error and the error in Frobenius norm.

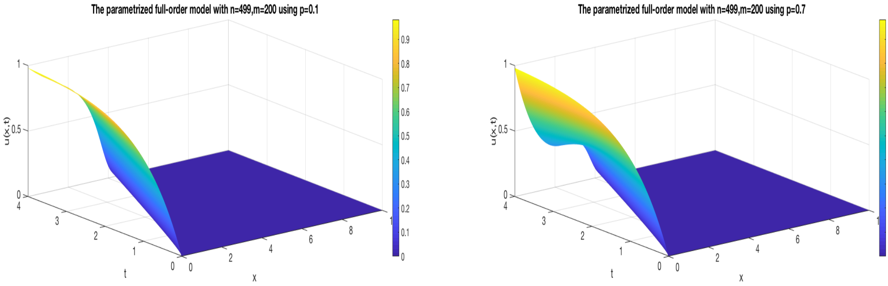

Figure 14: [Numerical test III] The full-order solution of the parametrized full-order system or each parameter $p = 0.1, 0.7$

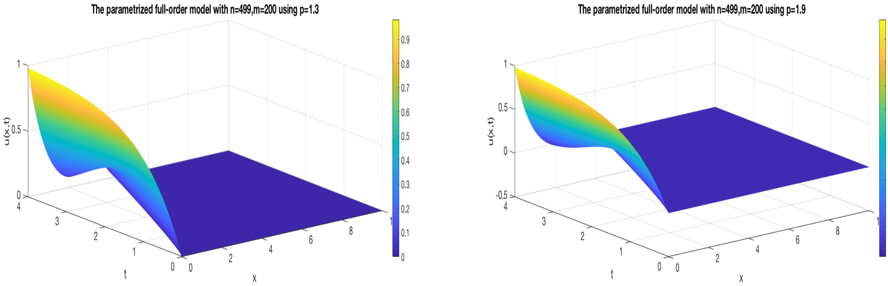

Figure 15: [Numerical test III] The full-order solution of the parametrized full-order system for each parameter $p = 1.3, 1.9$.

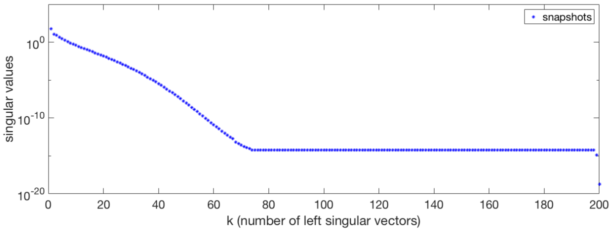

Figure 16: [Numerical test III] Singular values of 200 snapshots.

The parametrized reduced-order system can provide accurate approximation when we use enough number of left singular vectors as the POD basis.

p	k	$\lvert\mathbf{U}^{full} - \mathbf{U}^{pod}\rvert_m$ defined (4.6)	E_{pod} defined in (4.4)
0.1	20	1.6772×10^{-3}	2.1544×10^{-2}
	50	9.9885×10^{-10}	1.5503×10^{-8}
	60	1.2537×10^{-10}	1.6502×10^{-9}
	70	4.3969×10^{-11}	5.7988×10^{-10}
0.5	20	1.6772×10^{-3}	2.0996×10^{-2}
	50	4.7358×10^{-10}	9.7834×10^{-9}
	60	6.1561×10^{-11}	8.1805×10^{-10}
	70	2.0103×10^{-11}	2.8529×10^{-10}
0.7	20	1.6772×10^{-3}	2.0828×10^{-2}
	50	3.0355×10^{-10}	8.1335×10^{-9}
	60	3.3438×10^{-11}	4.6578×10^{-10}
	70	1.0919×10^{-11}	1.6428×10^{-10}

p	Fixed $k = k^*$	$\lvert\mathbf{U}^{full} - \mathbf{U}^{pod}\rvert_m$ defined (4.6)	E_{pod} defined in (4.4)
0.1	$k = 75$	3.9278×10^{-11}	4.9578×10^{-10}
0.5		1.9679×10^{-11}	2.4886×10^{-10}
0.7		1.0817×10^{-11}	1.4552×10^{-10}

Table 8: [Numerical test III] Errors of the approximate solutions from the parametrized POD reduced-order systems using different POD reduced dimensions $k = 20, 50, 60, 70$ and fixed $k = k^* = 75$ compared with the parametrized full-order system for each parameter $p = 0.1, 0.5, 0.7$.

Table 8, Table 9, and Figure 17 show that the approximated solution becomes less accurate if a parameter p is farther away from 1. This may result from the fact that the POD basis is generated from the full-order system with $p = 1$.

To further reduce complexity on the nonlinear term, the POD approach with fixed reduced dimension $k = k^* = 75$ is used with the DEIM procedure.

p	k	$\lvert\mathbf{U}^{full} - \mathbf{U}^{pod}\rvert_m$ defined (4.6)	E_{pod} defined in (4.4)
1.3	20	1.6772×10^{-3}	2.0584×10^{-2}
	50	3.0346×10^{-10}	8.1984×10^{-9}
	60	2.6227×10^{-11}	4.0344×10^{-10}
	70	1.2059×10^{-11}	1.4292×10^{-10}
1.5	20	1.6772×10^{-3}	2.0570×10^{-2}
	50	4.1014×10^{-10}	9.4245×10^{-9}
	60	4.1155×10^{-11}	6.4479×10^{-10}
	70	2.0227×10^{-11}	2.2767×10^{-10}
1.9	20	1.6772×10^{-3}	2.0561×10^{-2}
	50	6.8939×10^{-10}	1.2542×10^{-8}
	60	7.9749×10^{-11}	1.0841×10^{-9}
	70	3.6191×10^{-11}	3.8701×10^{-10}

p	Fixed $k = k^*$	$\lvert\mathbf{U}^{full} - \mathbf{U}^{pod}\rvert_m$ defined (4.6)	E_{pod} defined in (4.4)
1.3	$k = 75$	8.5359×10^{-12}	1.2388×10^{-10}
1.5		1.3180×10^{-11}	1.9464×10^{-10}
1.9		2.6785×10^{-11}	3.2308×10^{-10}

Table 9: [Numerical test III] Errors of the approximate solutions from the parametrized POD reduced-order systems using different POD reduced dimensions $k = 20, 50, 60, 70$ and fixed $k = k^* = 75$ compared with the parametrized full-order system for each parameter $p = 1.3, 1.5, 1.9$.

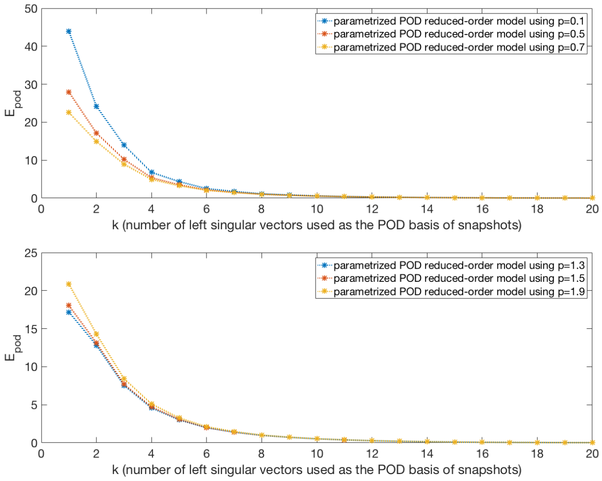

Figure 17: [Numerical test III] The number of left singular vectors used as the POD basis of snapshots is plotted against $\mathbf{E}_{pod}$ defined in (4.4) for each parameter $p = 0.1, 0.5, 0.7$ (top) and $p = 1.3, 1.5, 1.9$ (bottom) for all $k = 1, 2, \ldots, 20$.

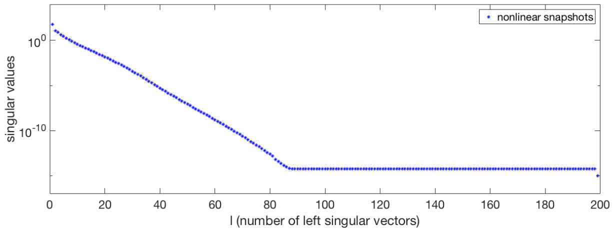

Figure 18: [Numerical test III] Singular values of 200 nonlinear snapshots.

For nonlinear snapshots, the decreasing of singular values reached a plateau for all $l \geq 90$, so that $l = 10, 20, 50, 70, 80$ are chosen as the DEIM reduced dimension in the numerical test shown next. In Table 10, we use small dimensions for POD and DEIM. Notice that, for each dimensions k and l, the errors are in the same order for all cases of p. The results show that the maximum absolute error and the error in Frobenius norm become in the order $\mathcal{O}(10^{-3})$ and $\mathcal{O}(10^{-2})$, respectively when using both POD and DEIM reduced dimension equal to 20. Similarly, Table 11 demonstrates the accuracy of the POD-DEIM reduced system with a fixed dimension of POD $k = k^* = 75$ (from the plot of singular values in Figure 15) with different DEIM dimensions $l = 20, 50, 70, 80$. As expected, the errors given in Table 11 is smaller than the ones given in Table 10, since the dimensions for both POD and DEIM are larger.

p	k	l	$\lvert\mathbf{U}^{full} - \mathbf{U}^{poddeim}\rvert_m$ defined in (4.6)	$\mathrm{E}_{poddeim}$ defined in (4.5)
0.1	10	10	3.1191×10^{-2}	5.8325×10^{-1}
	20	20	1.6772×10^{-3}	2.1547×10^{-2}
0.5	10	10	3.1184×10^{-2}	5.4698×10^{-1}
	20	20	1.6772×10^{-3}	2.1004×10^{-2}
0.7	10	10	3.1180×10^{-2}	5.3615×10^{-1}
	20	20	1.6772×10^{-3}	2.0837×10^{-2}
1.3	10	10	3.1170×10^{-2}	5.2474×10^{-1}
	20	20	1.6772×10^{-3}	2.0594×10^{-2}
1.5	10	10	3.1166×10^{-2}	5.2609×10^{-1}
	20	20	1.6772×10^{-3}	2.0570×10^{-2}
1.9	10	10	3.1159×10^{-2}	5.3405×10^{-1}
	20	20	1.6772×10^{-3}	2.0575×10^{-2}

Table 10: [Numerical test III] Errors of the approximate solutions from the parametrized POD-DEIM reduced-order systems using both POD and DEIM reduced dimensions $k = l = 10, 20$ compared with the parametrized full-order system for each parameter $p = 0.1, 0.5, 0.7, 1.3, 1.5, 1.9$.

p	Fixed $k = k^*$	l	$\lvert \mathbf{U}^{full} - \mathbf{U}^{poddeim} \rvert_m$	$\mathrm{E}_{poddeim}$
			defined in	defined in
			(4.6)	(4.5)
0.1	$k = 75$	20	3.7245×10^{-7}	2.1164×10^{-5}
		50	6.7328×10^{-10}	6.8051×10^{-9}
		70	1.3239×10^{-10}	1.5352×10^{-9}
		80	6.9482×10^{-11}	8.5743×10^{-10}
0.5	$k = 75$	20	1.8359×10^{-6}	1.0305×10^{-4}
		50	1.8565×10^{-9}	1.6664×10^{-8}
		70	2.8564×10^{-10}	3.4155×10^{-9}
		80	1.3916×10^{-10}	1.6287×10^{-9}
0.7	$k = 75$	20	2.5451×10^{-6}	1.4245×10^{-4}
		50	1.5334×10^{-9}	1.3196×10^{-8}
		70	2.1592×10^{-10}	2.6537×10^{-9}
		80	1.0879×10^{-10}	1.2600×10^{-9}
1.3	$k = 75$	20	4.5516×10^{-6}	2.5526×10^{-4}
		50	2.5708×10^{-9}	2.0578×10^{-8}
		70	3.7011×10^{-10}	3.9943×10^{-9}
		80	1.5772×10^{-10}	1.8632×10^{-9}
1.5	$k = 75$	20	5.1754×10^{-6}	2.9131×10^{-4}
		50	4.6997×10^{-9}	3.7305×10^{-8}
		70	7.1751×10^{-10}	7.2036×10^{-9}
		80	2.7675×10^{-10}	3.3289×10^{-9}
1.9	$k = 75$	20	6.3879×10^{-6}	3.6158×10^{-4}
		50	9.4859×10^{-9}	7.5372×10^{-8}
		70	1.6018×10^{-10}	1.4524×10^{-9}
		80	5.3135×10^{-10}	6.5688×10^{-9}

Table 11: [Numerical test III] Errors of the approximate solutions from the parametrized POD-DEIM reduced-order systems for fixed $k = k^* = 75$ using different DEIM reduced dimensions $l = 20, 50, 70, 80$ compared with the parametrized full-order system for each parameter $p = 0.1, 0.5, 0.7, 1.3, 1.5, 1.9$.

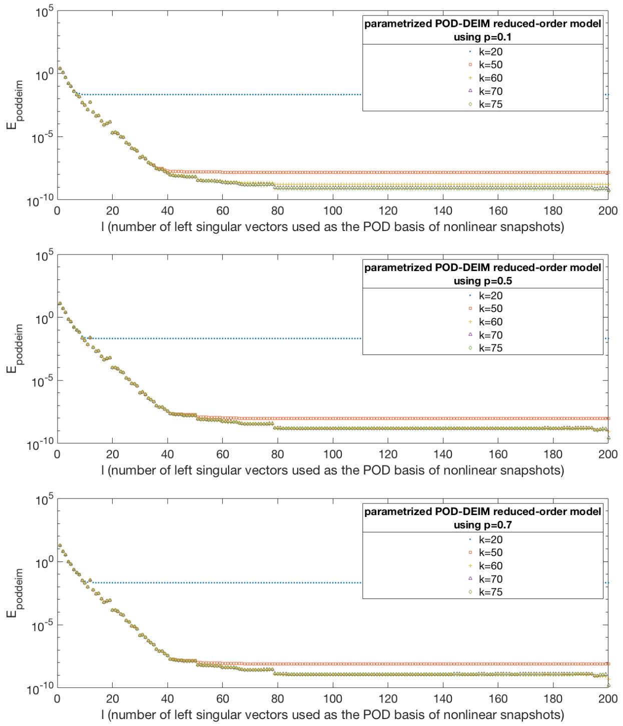

Figure 19: [Numerical test III] The number of left singular vectors used as the POD basis of nonlinear snapshots is plotted against $\mathbf{E}_{poddeim}$ defined in (4.5) for each parameter $p = 0.1, 0.5, 0.7$.

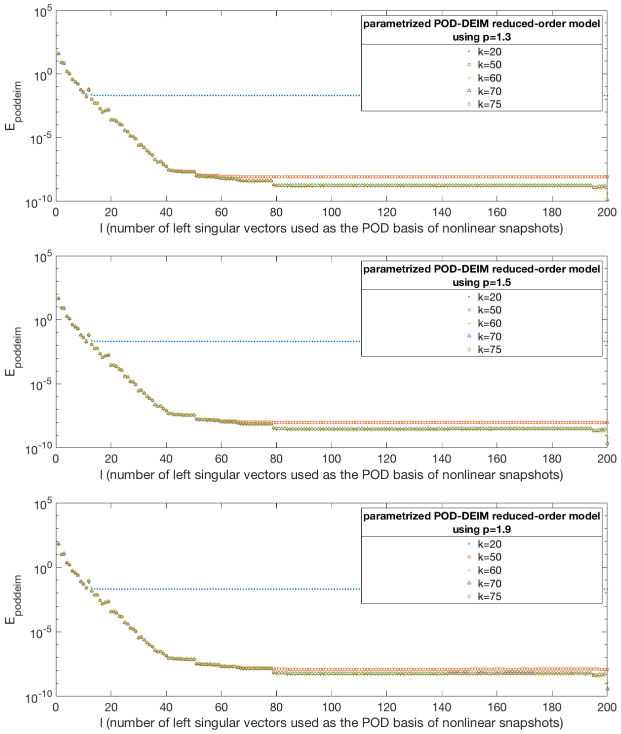

Figure 20: [Numerical test III] The number of left singular vectors used as the POD basis of nonlinear snapshots is plotted against $\mathbf{E}_{poddeim}$ defined in (4.5) for each parameter $p = 1.3, 1.5, 1.9$.

Notice that, the parametrized POD-DEIM reduced-order model give almost the same order of accuracy for all parameter values p as demonstrated in Table 11, Figure 19 and Figure 20. The numerical results in this section demonstrate that the POD-DEIM approach can be used for accurately approximating the solutions of many parametrized dynamic systems with various parameter values by using only one low-dimensional basis for each POD and DEIM. That is, we can construct a POD-DEIM reduced system for a given parameter value without actually solving the original full-order system using this same parameter value. Therefore, this can significantly save the simulation time and memory storage when approximating the solutions of dynamical systems with varying parameter.

5 CONCLUSION

This work applies model order reduction techniques known as proper orthogonal decomposition (POD) and discrete empirical interpolation method (DEIM) on a nonlinear dynamical system arising from the discretized sine-Gordon equation. POD is used to extract the dominant features of the trajectories or snapshots into a low-dimensional representation called POD basis, which is optimal in the sense that a certain approximation error involving snapshots is minimized. The POD basis is used together with the Galerkin projection on the discretized system of the sine-Gordon equation to obtain a reduced-order system with much smaller dimension. DEIM is then employed for constructing a selected interpolation spatial indices to provide a nearly optimal subspace approximation to avoid computing the nonlinear term with the full-order complexity.

The numerical experiments in this work investigate efficiency of the reduced-order model in three test cases. First, we studied efficiency of the POD-DEIM procedure on reproducing the numerical solutions of the sine-Gordon equation. It can transform the original model into a reduced-order model with much smaller dimension with less computational time and less memory storage while providing accurate approximation when compared with the full-order system. Next, we investigate the accuracy of the reduced-order model constructed from different amount of snapshots. These snapshots are the numerical solutions solved from the full-order system with different time step sizes, which are larger than the step size used in the reduced system. That is, we can approximate the solution at the time steps, which are not available or previously computed from the full-order system. In this numerical test, for fixed small dimensions of POD and DEIM, there is no significant difference in accuracy of the approximation when different amount of snapshots are used. When the ratio of the retained energy defined in term of singular values is used to select the POD and DEIM dimensions, the approximation becomes more accurate as this ratio approaching 1 for every case of the amount of snapshots. Finally, we studied an ability of POD-DEIM approach for approximating the parametrized full-order system. This numerical experiment illustrates that the POD-DEIM approach can approximate the solutions of parametrized dynamical systems with various parameter values without previously solving for the solutions of the original full-order systems with those parameter values. The POD-DEIM approach is therefore efficient for accurately predicting the numerical solutions for the sine-Gordon equation. Similar applicability of the POD-DEIM concept can be readily extended to various classes of nonlinear dynamical systems.

Acknowledgement: The authors gratefully acknowledge the financial support provided by Thammasat University Research Fund under the TU Research Scholar, Contract No. 2/19/2561.

References

[1] Maxime Barrault, Yvon Maday, Ngoc Cuong Nguyen, and Anthony T. Patera. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667 – 672, 2004.

[2] B. Batiha, M.S.M. Noorani, and I. Hashim. Numerical solution of sine-gordon equation by variational iteration method. *Physics Letters A*, 370(5):437 – 440, 2007.

[3] Matteo Bonotto, Angelo Cenedese, and Paolo Bettini. Krylov subspace methods for model order reduction in computational electromagnetics**this work is partially supported by the cpda144817 grant of the university of padova. *IFAC-PapersOnLine*, 50(1):6355 – 6360, 2017.

[4] A.G. Bratsos. A numerical method for the one-dimensional sine-gordon equation. *Numerical Methods for Partial Differential Equations*, 24(3):833–844, 2008.

[5] Jens Bremer, Pawan Goyal, Lihong Feng, Peter Benner, and Kai Sundmacher. Pod-deim for efficient reduction of a dynamic 2d catalytic reactor model. *Computers and Chemical Engineering*, 106:777 – 784, 2017.

[6] Yanhua Cao, Jiang Zhu, Zhendong Luo, and I.M. Navon. Reduced-order modelling of the upper tropical pacific ocean model using proper orthogonal decomposition. *Computers and Mathematics with Applications*, 52(8):1373 – 1386, 2006.

[7] P Caudrey, J Eilbeck, and John Gibbon. The sine-gordon equation as a model classical field theory. *Il Nuovo Cimento B*, 25:497–512, 02 1975.

[8] Gary Han Chang, Clemens M. Schirmer, and Yahya Modarres-Sadeghi. A reduced-order model for wall shear stress in abdominal aortic aneurysms by proper orthogonal decomposition. *Journal of Biomechanics*, 54:33 – 43, 2017.

[9] Saifon Chaturantabut and Danny C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

[10] Saifon Chaturantabut and Danny C. Sorensen. A state space error estimate for pod-deim nonlinear model reduction. *SIAM Journal on Numerical Analysis*, 50(1):46–63, 2012.

[11] Mehdi Dehghan and Mostafa Abbaszadeh. A combination of proper orthogonal decomposition–discrete empirical interpolation method (pod–deim) and meshless local rbf-dq approach for prevention of groundwater contamination. *Computers and Mathematics with Applications*, 75(4):1390 – 1412, 2018.

[12] Zhe Ding, Li Li, Jianyi Kong, and Li Qin. A modal projection-based reduction method for transient dynamic responses of viscoelastic systems with multiple damping models. *Computers and Structures*, 194:60 – 73, 2018.

[13] Karl Pearson F.R.S. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[14] F. Ghavamian, P. Tiso, and A. Simone. Pod–deim model order reduction for strain-softening viscoplasticity. *Computer Methods in Applied Mechanics and Engineering*, 317:458 – 479, 2017.

[15] C. Guarcello, K. Fedorov, D. Valenti, B. Spagnolo, and A. Ustinov. Sine-Gordon breathers generation in driven long Josephson junctions. *ArXiv e-prints*, January 2015.

[16] Xian hang Sun and Ming hai Xu. Optimal control of water flooding reservoir using proper orthogonal decomposition. *Journal of Computational and Applied Mathematics*, 320:120 – 137, 2017.

[17] H. Hotelling. Analysis of a complex of statistical variables into principal components. *J. Educ. Psych.*, 24, 1933.

[18] V. G. Ivancevic and T. T. Ivancevic. Sine-gordon solitons, kinks and breathers as physical models of nonlinear excitations in living cellular structures. *ArXiv e-prints*, 2013.

[19] Zi-Wu Jiang and Ren-Hong Wang. Numerical solution of one-dimensional sine–gordon equation using high accuracy multiquadric quasi-interpolation. *Applied Mathematics and Computation*, 218(15):7711 – 7716, 2012.

[20] Bülent Karasözen, Canan Akkoyunlu, and Murat Uzunca. Model order reduction for nonlinear schrödinger equation. *Applied Mathematics and Computation*, 258:509 – 519, 2015.

[21] Tanya Kostova-Vassilevska and Geoffrey M. Oxberry. Model reduction of dynamical systems by proper orthogonal decomposition: Error bounds and comparison of methods using snapshots from the solution and the time derivatives. *Journal of Computational and Applied Mathematics*, 330:553 – 573, 2018.

[22] E. Liberge and A. Hamdouni. Reduced order modelling method via proper orthogonal decomposition (pod) for flow around an oscillating cylinder. *Journal of Fluids and Structures*, 26(2):292 – 311, 2010.

[23] Shenglan Liu, Jun Wu, Lin Feng, Sen Luo, and Deqin Yan. Quasi-curvature local linear projection and extreme learning machine for nonlinear dimensionality reduction. *Neurocomputing*, 277:208 – 217, 2018.

[24] M. Loeve. *Probability theory / M. Loeve*. Springer-Verlag New York, 4th ed. edition, 1977.

[25] B. Maboudi Afkham and J. S. Hesthaven. Structure-Preserving Model-Reduction of Dissipative Hamiltonian Systems. *ArXiv e-prints*, 2017.

[26] Hessam Mirgolbabaei and Tarek Echekki. Nonlinear reduction of combustion composition space with kernel principal component analysis. *Combustion and Flame*, 161(1):118 – 126, 2014.

[27] M. Mordhorst, T. Strecker, D. Wirtz, T. Heidlauf, and O. Röhrle. Pod-deim reduction of computational emg models. *Journal of Computational Science*, 19:86 – 96, 2017.

[28] Ana I. Moreno, Artur A. Jarzabek, José M. Perales, and José M. Vega. Aerodynamic database reconstruction via gappy high order singular value decomposition. *Aerospace Science and Technology*, 52:115 – 128, 2016.

[29] K. Nakajima, T. Yamashita, and Y. Onodera. Mechanical analogue of active Josephson transmission line. *Journal of Applied Physics*, 45:3141–3145, July 1974.

[30] S Nam and Hwajoon Kim. The representation on solutions of the sine-gordon and klein-gordon equations by laplace transform. *Applied Mathematical Sciences*, pages 4433–4440, 01 2014.

[31] Benjamin Peherstorfer and Karen Willcox. Data-driven operator inference for nonintrusive projection-based model reduction. *Computer Methods in Applied Mechanics and Engineering*, 306:196 – 215, 2016.

[32] Austin A. Phoenix and Pablo A. Tarazaga. Dynamic model reduction using data-driven loewner-framework applied to thermally morphing structures. *Journal of Sound and Vibration*, 396:274 – 288, 2017.

[33] Wenting Shao and Xionghua Wu. The numerical solution of the nonlinear klein–gordon and sine–gordon equations using the chebyshev tau meshless method. *Computer Physics Communications*, 185(5):1399 – 1409, 2014.

[34] H.S. Shukla and Mohammad Tamsir. Numerical solution of nonlinear sine–gordon equation by using the modified cubic b-spline differential quadrature method. *Beni-Suef University Journal of Basic and Applied Sciences*, 2016.

[35] Qiu-Yan Song, Yao-Lin Jiang, and Zhi-Hua Xiao. Arnoldi-based model order reduction for linear systems with inhomogeneous initial conditions. *Journal of the Franklin Institute*, 354(18):8570 – 8585, 2017.

[36] Zachary P. Stanko, Scott E. Boyce, and William W.-G. Yeh. Nonlinear model reduction of unconfined groundwater flow using pod and deim. *Advances in Water Resources*, 97:130 – 143, 2016.

[37] R. Ştefănescu, A. Sandu, and I.M. Navon. Pod/deim reduced-order strategies for efficient four dimensional variational data assimilation. *Journal of Computational Physics*, 295:569 – 595, 2015.

[38] Abel Kurura Tinega and Oduor Okoya Michael Edmund. Finite difference solution of (1+1) sine-gordon equation: A mathematical model for the rigid pendula attached to a stretched wire. *The Standard International Journals (The SIJ)*, 3(4):55–61, 2015.

[39] Christian Tolks and Christoph Ament. Model order reduction of glucose-insulin homeostasis using empirical gramians and balanced truncation**this work was supported by iav automotive engineering gmbh, berlin, germany. *IFAC-PapersOnLine*, 50(1):14735 – 14740, 2017.

[40] Mengyu Wang, Debaditya Dutta, Kang Kim, and John C. Brigham. A computationally efficient approach for inverse material characterization combining gappy pod with direct inversion. *Computer Methods in Applied Mechanics and Engineering*, 286:373 – 393, 2015.

[41] Huanhuan Yang and Alessandro Veneziani. Efficient estimation of cardiac conductivities via pod-deim model order reduction. *Applied Numerical Mathematics*, 115:180 – 199, 2017.

[42] Zhihua Zhang and John C. Moore. Chapter 6 - empirical orthogonal functions. In Zhihua Zhang and John C. Moore, editors, *Mathematical and Physical Fundamentals of Climate Change*, pages 161 – 197. Elsevier, Boston, 2015.

[43] Li zhou Li, Jun Zhang, Jun li Zhao, and Zhu feng Yue. An enhanced 3d data transfer method for fluid–structure interface by isomap nonlinear space dimension reduction. *Advances in Engineering Software*, 83:19 – 30, 2015.

(Received 17 August 2018)
(Accepted 25 December 2018)