Applications in Engineering Science 8 (2021) 100074

Contents lists available at ScienceDirect

Applications in Engineering Science

journal homepage: www.elsevier.com/locate/apples

Model-reduction techniques for PDE models with Turing type electrochemical phase formation dynamics

Benedetto Bozzini [a], Angela Monti [b], Ivonne Sgura [b,*]

[a] Dipartimento di Energia, Politecnico di Milano, via Lambruschini, 4, 20156 Milano, Italy
[b] Dipartimento di Matematica e Fisica "E. De Giorgi", Università del Salento, Via per Arnesano, 73100 Lecce, Italy

ARTICLE INFO

Keywords:
Model Order Reduction (MOR)
POD-Galerkin
Dynamic Mode Decomposition
Turing patterns
Reaction–diffusion PDE systems
Battery modelling

ABSTRACT

Next-generation battery research will heavily rely on physico-chemical models, combined with deep learning methods and high-throughput and quantitative analysis of experimental datasets, encoding spectral information in space and time. These tasks will require highly efficient computational approaches, to yield rapidly accurate approximations of the models. This paper explores the capabilities of a representative range of model reduction techniques to face this problem in the case of a well-assessed electrochemical phase-formation model. We consider the Proper Orthogonal Decomposition (POD) with a Galerkin projection and the Dynamic Mode Decomposition (DMD) techniques to deal first of all with a semi-linear heat equation 2D in space as a test problem. As an application, we show that it is possible to save computational time by applying POD-Galerkin for different choices of the parameters without recalculating the snapshot matrix. Finally, we consider two reaction–diffusion (RD) PDE systems with Turing-type dynamics: the well-known Schnackenberg model and the DIB model for electrochemical phase formation. We show that their reduced models obtained by POD and DMD with suitable low-dimensional projections are able to approximate carefully both the Turing patterns at the steady state and the reactivity dynamics in the transient regime. Finally, for the DIB model we show that POD-Galerkin applied for different choices of parameters, by calculating once the snapshot matrices, is able to find reduced Turing patterns of different morphology.

1. Introduction

The development of next-generation batteries, that will allow to overcome safety, cost, sustainability and durability drawbacks of present-generation lithium-ion based technologies, is impaired by shape instability problems developing during discharge–charge cycling for some active materials, especially metallic anodes Liang et al. (2020) and Li et al. (2020). Such uncontrollable shape changes are related to poorly understood phase-formation processes accompanying the electric energy storage and release processes going on in batteries, and lead to loss of capacity and internal short circuits, in turn limiting the efficiency and lifetime of the devices. In order to gain a more insightful physico-chemical understanding of the processes and to control them in real-life devices, mathematical modelling of electrochemical phase formation processes is starting to play a key role: an updated account of the available approaches can be found in Bozzini et al. (2018). Among PDE approaches to electrode shape-change processes, some of the authors have developed a reaction–diffusion PDE model, customarily referred to as the DIB model (see e.g. Lacitignola et al. (2015) and Sgura et al. (2019)), characterized by the coupling of electrode morphology and surface chemistry. Specifically, this model, addressing the practically relevant mesoscopic scale, accounts for the interaction between material shape and material chemistry through adatom and adsorbate surface diffusion and source terms including the physics describing the growth process and the operating conditions (electrolyte chemistry and charge/discharge rate). This model has been proven by both theoretical analysis and comparison with a wide corpus of experimental data. It represents the essential features of unstable material growth in electrochemical systems, in particular by means of the so-called Turing patterns (see Lacitignola et al. (2015, 2017), Bozzini et al. (2019) and reference therein). One of the key results of the analysis of the DIB model Lacitignola et al. (2017), Sgura et al. (2019) and Sgura and Bozzini (2017) is the correlation of the model parameters with the occurrence and type of growth instabilities. In particular, in Sgura et al. (2019) a segmentation of the parameter-space in morphological classes has been proposed after solving the model for several choices of the parameters involved. This exhaustive parameter-space mapping task is extremely computationally expensive because each output requires a very fine spatial discretization to capture the characteristic features

* Corresponding author.
E-mail addresses: benedetto.bozzini@polimi.it (B. Bozzini), angela.monti@unisalento.it (A. Monti), ivonne.sgura@unisalento.it (I. Sgura).

https://doi.org/10.1016/j.apples.2021.100074
Received 27 October 2021; Accepted 1 November 2021
Available online 21 November 2021

of the pattern, together with a long integration in time, often with small time steps, because each pattern is a stationary solution of the PDE system. Despite its computational cost, this approach proved to be extremely successful in support of parameter identification purposes from experimental data maps Sgura and Bozzini (2017) and Sgura et al. (2019). High-throughput and automated versions of parameter-space exploration is strongly needed in view of systematic comparison between experimental and computed space–time dependent datasets, aimed at developing quantitative physical understanding as well as data reduction/compression tools. For this reason, new techniques would be needed to reduce complex computationally demanding operations. As a first step towards this goal, in D'Autilia et al. (2020), the authors proposed a matrix-oriented approach to solve the semi-discretized PDE system in time by a sequence of Sylvester matrix equations, that are much smaller in terms of dimension than the usual vector approach.

In the present paper, by following the approaches in Alla and Kutz (2017), Benner and Breiten (2015), Benner et al. (2015), Brunton and Kutz (2019), Kramer and Willcox (2019) and Peherstorfer and Willcox (2015), we deal with the semi-discretized PDEs by Model Order Reduction (MOR) techniques for the purpose of reducing costs and complexity, but such that the essential features of the solution are preserved, in terms of accuracy for the Turing pattern at the steady state and the time dynamics (also denominated "reactivity", see D'Autilia et al. (2020)). In particular, here in Section 2 we focus on the Proper Orthogonal Decomposition (POD) with a Galerkin projection Benner et al. (2015), Benner and Breiten (2015), Kramer and Willcox (2019), Peherstorfer and Willcox (2015), Sirovich (1987) and Volkwein (2013) and on the Dynamic Mode Decomposition (DMD) Alla and Kutz (2017) and Brunton and Kutz (2019). To present practical numerical aspects of these methods and to compare their performance in terms of reduction of computational cost and preserved solution accuracy, first of all in Section 3, we apply these techniques to a semi-linear heat equation 2D in space, for which the exact solution is available. For this simple model, we present numerical simulations to study the effect of the low-dimensional projections in terms of errors with respect to the full model and then we provide an application of POD-Galerkin approach when several choices of the diffusion and reaction parameters are considered, without recalculating the snapshot matrix. In Section 4, we consider MOR for reaction–diffusion PDE systems. First of all, we consider the Schnackenberg model, that is a prototype PDE system with Turing pattern formation studied in literature (see e.g. Madzvamuse (2006)), and the DIB model for electrodeposition the high-throughput implementation of which is the ultimate goal of the research presented in this paper. For both these models we show that POD-Galerkin and DMD with a suitable low-dimensional projection are able to approximate carefully the Turing pattern at the steady state and the reactivity dynamics in the transient regime. In particular, for both models we show that DMD exhibits an oscillating behaviour of the spatial mean that is damped for increasing values of the projection space. Finally, for the DIB model we show that POD-Galerkin applied for different choices of parameters, by calculating once the snapshot matrices, is able to find reduced Turing patterns of different morphology, as predicted by the classification in Sgura et al. (2019). In addition to the numerical advantages, the formation of a suitable base from a snapshot matrix from which a rich scenario of morphologies can be constructed, also contributes to the mechanistic clarification of the physico-chemical complexity underlying electrochemical pattern formation. All computations presented in the paper have been performed in Matlab (ver. 2019a) on a computer Dell, i7 IntelCore processor, 2.8 GHz, 16Gb RAM.

2. Model reduction techniques

We consider the following dynamical system

$$\begin{cases} \dot{\mathbf{u}}(t) = A\mathbf{u}(t) + \mathbf{f}(\mathbf{u}(t)), \quad t \in (t_0, T] \\ \mathbf{u}(t_0) = \mathbf{u}_0 \end{cases} \tag{1}$$

with initial conditions $\mathbf{u}_0 \in \mathbb{R}^n$, $A \in \mathbb{R}^{n\times n}$ and $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$. The ODE system (1) typically arises from the spatial semi-discretization of a time dependent partial differential equation (PDE), where n is the number of mesh points used for the spatial approximation by classical approaches (finite differences, finite elements, etc.). In the context of Model Order Reduction (MOR) techniques, this system is also referred to as *full model*. After the application of a numerical method in time to (1), with time step h_t on the meshgrid $t_k = t_0 + kh_t$, $k = 0, \dots, n_t$, as in Alla and Kutz (2017) and Benner et al. (2015), we construct the so-called *snapshot matrix* as

$$S = \begin{bmatrix} | & | & \dots & | \\ \mathbf{u}_0 & \mathbf{u}_1 & \dots & \mathbf{u}_{n_t} \\ | & | & \dots & | \end{bmatrix} \in \mathbb{R}^{n\times(n_t+1)} \tag{2}$$

where $\mathbf{u}_k \approx \mathbf{u}(t_k)$ is the numerical solution at each time t_k.

In some applications, long time integration and a large finely discretized domain are needed to capture the essential features of the solution, such that (1) is a high dimensional ODE system and also the snapshot matrix can be a huge matrix. For these reasons, following the approach in Alla and Kutz (2017), we aim to obtain low-dimensional models by applying MOR techniques in order to reduce costs and complexity, while preserving the feature of the original PDE. In this paper, we consider the Proper Orthogonal Decomposition with a Galerkin projection (POD- Galerkin) Sirovich (1987) and Volkwein (2013) and the Dynamic Mode Decomposition (DMD) Alla and Kutz (2017) and Brunton and Kutz (2019). The first one is an *intrusive* technique, that constructs the reduced model starting directly from the full model (1). In other words POD-Galerkin can be applied when we know the original PDE model that generates the values in S. On the contrary, DMD is a data-driven algorithm that ignores a possible underlying dynamics beyond S, which values could be given, for example, as experimental data. Since the Singular Value Decomposition (SVD) of a matrix is a crucial computation for the application of both the POD-Galerkin projection and the DMD technique, we start by defining its formulation and its main properties.

2.1. Singular value decomposition

The SVD is one of the most important matrix factorization with several theoretical properties of linear algebra (see Brunton and Kutz (2019)). In many fields of applications, the SVD is used to extract, from high dimensional data, dominant patterns that are typically low rank and to indicate where the information is concentrated. In our context it provides a unique matrix decomposition for the snapshot matrix (2) given by

$$S = V\Sigma W^T, \tag{3}$$

where Σ is a $n \times (n_t + 1)$ diagonal matrix which diagonal entries $\sigma_i \geq 0$, $i = 1, \dots, \min(n, n_t)$ are the *singular values* of S, that are sorted in decreasing order and in several cases rapidly go to zero. V, W are orthogonal matrices of dimensions n and $n_t + 1$, respectively. The SVD (3) allows to select a set of optimal basis modes by using the leading columns of V (see Volkwein (2013)). The main idea of a MOR technique to reduce the number n of ODEs in (1) is to choose a rank-r approximation of S by means of the corresponding truncated SVD given by

$$S_r = V_r\Sigma_r W_r^T. \tag{4}$$

(4) is obtained by keeping the leading r singular values, that is $\Sigma_r = diag(\sigma_1, \sigma_2, \dots, \sigma_r)$ and by taking the first r columns of V and W in (3) such that:

$$V_r = \begin{bmatrix} | & | & \dots & | \\ \mathbf{v}_1 & \mathbf{v}_2 & \dots & \mathbf{v}_r \\ | & | & \dots & | \end{bmatrix} \in \mathbb{R}^{n\times r}, \quad W_r = \begin{bmatrix} | & | & \dots & | \\ \mathbf{w}_1 & \mathbf{w}_2 & \dots & \mathbf{w}_r \\ | & | & \dots & | \end{bmatrix} \in \mathbb{R}^{(n_t+1)\times r} \tag{5}$$

The computation of the truncated SVD (4)–(5) is a key tool to construct a reduced space onto which the system (1) will be projected, as explained in details in the next subsections.

2.2. POD-Galerkin projection

The POD-Galerkin projection (see Alla and Kutz (2017) and Volkwein (2013)) is an intrusive technique that constructs the reduced model starting directly from the n-dimensional full model (1) and where the fundamental assumption is the explicit knowledge of the term $f(\mathbf{u})$ in the full model.

Let be $r \ll n$, given an orthogonal basis $V_r = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_r] \in \mathbb{R}^{n\times r}$ as in (4) computed from the snapshot matrix S of the full model (1), the POD Galerkin projection approximates the state vector $\mathbf{u}(t) \in \mathbb{R}^n$ as follows:

$$\mathbf{u}(t) \approx \tilde{\mathbf{u}}(t) := V_r \tilde{\mathbf{u}}(t) \ \in \mathbb{R}^n \tag{6}$$

where $\tilde{\mathbf{u}}(t) \in \mathbb{R}^r$ will be said *reduced state*. With this assumption (6) and by projecting the full model onto V_r we obtain the reduced dynamical system

$$\begin{cases} \dot{\tilde{\mathbf{u}}}(t) = A_r \tilde{\mathbf{u}}(t) + \mathbf{f}_r(\tilde{\mathbf{u}}(t)), \quad t \in (t_0, T], \\ \tilde{\mathbf{u}}(t_0) = \tilde{\mathbf{u}}_0, \end{cases} \tag{7}$$

where $A_r = V_r^T A V_r \in \mathbb{R}^{r\times r}$

$$\mathbf{f}_r(\tilde{\mathbf{u}}(t)) = V_r^T \mathbf{f}(V_r \tilde{\mathbf{u}}(t)) \in \mathbb{R}^r \tag{8}$$

and the initial condition $\tilde{\mathbf{u}}_0 = V_r^T \mathbf{u}_0$. The reduced system has dimension r and then if $r \ll n$, by solving (7)–(8) we achieve a significant dimensionality reduction.

2.3. Dynamic mode decomposition

The Dynamic Mode Decomposition introduced in Alla and Kutz (2017) and Brunton and Kutz (2019) is applied directly to a snapshot matrix, that can be formed by numerical simulations or can be provided by sets of experimental data. Hence, DMD ignores a possible underlying continuous or discrete dynamical system and tries to identify the underlying process by fitting the dynamics hidden in the snapshots in a linear least squares sense. DMD is able to return a modal decomposition of the dynamics, but it is different from POD Volkwein (2013) which computes orthogonal modes as described before. Following Brunton and Kutz (2019), we resume the DMD algorithm as follows. Given a snapshot matrix S as in (2), the first step constructs the two matrices

$$S_1 = \begin{bmatrix} | & | & \dots & | \\ \mathbf{u}_0 & \mathbf{u}_1 & \dots & \mathbf{u}_{n_t-1} \\ | & | & \dots & | \end{bmatrix} \in \mathbb{R}^{n\times n_t}, \quad S_2 = \begin{bmatrix} | & | & \dots & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \dots & \mathbf{u}_{n_t} \\ | & | & \dots & | \end{bmatrix} \in \mathbb{R}^{n\times n_t}. \tag{9}$$

Then DMD tries to find the best fitting matrix operator A such that

$$S_2 = AS_1, \quad \Longleftrightarrow \quad \mathbf{u}_{k+1} = A\mathbf{u}_k, \ k = 0, \dots, n_t - 1 \tag{10}$$

The solution in the least squares sense is obtained by computing $S_1^\dagger$ the Moore–Penrose pseudo-inverse of S_1 and it is given by $A = S_2 S_1^\dagger$. Due to the large dimensions of $S_i, i = 1, 2$, from a computational point of view it is not convenient to calculate A directly. In fact, the DMD algorithm computes a projection of A onto a reduced r-dimensional subspace, getting a matrix $\tilde{A} \approx A$, such that the time dynamics can be reconstructed by $\tilde{\mathbf{u}}_{k+1} = \tilde{A}\tilde{\mathbf{u}}_k$, starting by $\tilde{\mathbf{u}}_0$ a projection of the first snapshot $\mathbf{u}_0$, $\tilde{\mathbf{u}}_k$ is the reduced state variable in this case. The main idea is to get the spectral decomposition of $\tilde{A}$ and by its eigenvalues and eigenvectors recover an approximation $\bar{\mathbf{u}}_k$ of the full state variable $\mathbf{u}_k$. More implementation details about DMD are reported in **Algorithm 1**. As for the POD-Galerkin projection, a crucial point for DMD is to choose an appropriate value of r.

Algorithm 1 DMD.

Step 1. Given S_1 in Eq. (9), select a rank $r \geq 1$ and compute the truncated SVD of S_1 by $S_{1r} = V_r \Sigma_r W_r^T$.

Step 2. To project the matrix A onto the leading vectors of V_r, calculate the $r \times r$ matrix

$$\tilde{A} = V_r^T S_2 W_r \Sigma_r^{-1} \in \mathbb{R}^{r\times r}$$

and define the linear reduced model for the reduced state $\tilde{\mathbf{u}}_k$ by

$$\tilde{\mathbf{u}}_{k+1} = \tilde{A}\tilde{\mathbf{u}}_k.$$

Step 3. Compute the spectral decomposition of $\tilde{A}$

$$\tilde{A}W = W\Lambda$$

where columns of W are eigenvectors and $\Lambda = diag(\lambda_1, \dots, \lambda_r)$ contains the eigenvalues.

Step 4. Reconstruct the high-dimensional DMD modes of A by

$$\Phi = S_2 W_r \Sigma_r^{-1} W \in \mathbb{R}^{n\times r}$$

and recover an approximation $\bar{\mathbf{u}}_k$ of the full state variable $\mathbf{u}_k$ by

$$\mathbf{u}_k \approx \bar{\mathbf{u}}_k = \sum_{j=1}^{r} \phi_j \lambda_j^k b_j, \ k = 0, \dots, n_t, \quad \text{such that } \Phi\mathbf{b} = \mathbf{u}_0. \tag{11}$$

$\mathbf{u}_0$ is the first snapshot, ϕ_j are the columns of Φ and are eigenvectors of the high- dimensional matrix A, λ_j are the corresponding eigenvalues, while b_j are obtained by solving the above overdetermined system in least square sense.

3. Scalar PDE: semi-linear heat equation

In this section, to show the performance of the reduction techniques for time dependent PDEs including diffusion and reaction scales, we consider as test problem a semi-linear heat equation with exact solution. In order to create the snapshot matrix, we approximate the PDE solution by the matrix-oriented approach described in D'Autilia et al. (2020) and briefly reported below, that allows a fast computation with respect to the classical vector approach. We recall that in general the PDEs that we want to solve depend on a set of parameters. For this first example, we want to study the performance of the MOR techniques: (i) for different values of the truncation parameter r and (ii) by solving directly the reduced systems for new parameter values without changing the snapshot matrix, i.e. by constructing once the snapshot matrix. We will demonstrate this last strategy by changing the values of the diffusion and reaction parameters in the following scalar PDE.

We consider the heat equation with a linear reaction term and zero Neumann boundary conditions given by

$$\begin{cases} u_t = d\Delta u + \alpha u, \ \ (x, y) \in \Omega \subset \mathbb{R}^2, \ t \in [0, T_f] \\ (\mathbf{n}\nabla u)_{|\partial\Omega} = 0, \ \ u(x, y, 0) = u_0(x, y) \end{cases} \tag{12}$$

$d > 0$ diffusion coefficient, α reaction coefficient, $\Omega = [0, L] \times [0, L]$ a square domain and the initial condition $u_0(x, y) = cos(\pi x)cos(2\pi y)$, such that the exact solution is given by $u^*(x, y, t) = e^{(\alpha - 5\pi^2 d)t} u_0(x, y)$.

As described in details in D'Autilia et al. (2020), solving (12) by the Method of Lines based on a finite difference spatial discretization of Ω with stepsizes $h_x = \frac{L}{n_x+1}, h_y = \frac{L}{n_y+1}$ and n_x and n_y interior meshpoints, we have the ODE system (full model):

$$\begin{cases} \dot{\mathbf{u}}(t) = d\tilde{\Delta}\mathbf{u}(t) + \alpha\mathbf{u}(t) \\ \mathbf{u}(t_0) = \mathbf{u}_0 \end{cases} \tag{13}$$

where $\tilde{\Delta} = \frac{1}{h_x^2}(I \otimes T_1) + \frac{1}{h_y^2}(T_2 \otimes I) \in \mathbb{R}^{n\times n}, n = n_x n_y$, is the discrete Laplace operator that is a very large block matrix defined in terms of the Kronecker product $\otimes$. $T_1 \in \mathbb{R}^{n_x\times n_x}$ and $T_2 \in \mathbb{R}^{n_y\times n_y}$ account for the discretization of second order derivatives along x and y, respectively, and include a contribution due to the approximation of zero Neumann BCs.

3.1. Snapshot matrix: an efficient computation

In order to reduce the computational cost due to the construction of the snapshot matrix, we approximate the PDE solution by applying a matrix-oriented approach (see D'Autilia et al. (2020)) that rewrite (12) as the following differential matrix equation such that $\mathrm{vec}(Z) = \mathbf{u}$:

$$\dot{Z} = T_1 Z + Z T_2 + F(Z), \quad Z(0) = Z_0,$$

where T_1 and T_2 are described before, $F(Z) = \alpha Z$ is the reaction term evaluated componentwise and $Z_0(x_i, y_j) = u_0(x_i, y_j)$ for $(x_i, y_j), i = 1, \dots, n_x, j = 1, \dots, n_y$ on the spatial meshgrid. Then the IMEX Euler method in matrix form is given by

$$Z_{k+1} - Z_k = h_t(T_1 Z_{k+1} + Z_{k+1} T_2) + h_t F(Z_k), \quad k = 0, \dots, n_t - 1 \tag{14}$$

that is equivalent to the following sequence of Sylvester matrix equations

$$M_1 Z_{k+1} + Z_{k+1} M_2 = C_k, \tag{15}$$

where $C_k = Z_k + h_t F(Z_k)$, while the coefficient matrices $M_1 = I - h_t T_1$ and $M_2 = -h_t T_2$ do not change during time evolution. The solution of (15) at each time t_k is a matrix Z_k which entries approximate the solution $u(x_i, y_j, t_k)$ in each point (x_i, y_j). Here, we calculate this solution by the *rEuler* method described in D'Autilia et al. (2020) solving (15) in the spectral space in very efficient and fast way. Hence, we create the snapshot matrix S in (2) with columns given by $\mathbf{u_k} = vec(Z_k)$. In other words $\mathbf{u_k}$ is the vectorization of Z_k, where vec stacks the columns of Z_k beneath each other. Once we have created the snapshot matrix S, we can proceed with the application of the POD-Galerkin and DMD techniques.

To apply the MOR techniques in exam, firstly we need to compute a truncated SVD of the snapshot matrix S as in (4) in order to obtain the orthogonal matrix V_r, for a given fixed $r \ll n$.

The application of a **POD-Galerkin projection** to the full system (12) yields the reduced system

$$\begin{cases} \dot{\tilde{\mathbf{u}}}(t) = d A_r \tilde{\mathbf{u}}(t) + \alpha \tilde{\mathbf{u}}(t) \\ \tilde{\mathbf{u}}(t_0) = \tilde{\mathbf{u}}_0 = V_r^T \mathbf{u}_0 \end{cases} \tag{16}$$

where $A_r = V_r^T \tilde{\Delta} V_r \in \mathbb{R}^{r\times r}$. If $\tilde{\mathbf{u}}_k$ is the numerical solution of this reduced r-dimensional ODE system, then we will recover an approximation of the solution of the full model (13) by $\bar{\mathbf{u}}_k = V_r \tilde{\mathbf{u}}_k$, for $k = 0, \dots, n_t$. It is worth noting that solving directly the full model (13) by an usual vector approach (see also D'Autilia et al. (2020)) will require at each time step the solution of a large sparse linear system of dimension $n = n_x \cdot n_y$ and this is very expensive for $n_x = n_y \geq 50$ ($n \geq 2500$) mesh points. For this reason, it is more convenient to solve the reduced system (16) that is not sparse, but it has dimension $r \ll n$, as we will show in the numerical simulations. In fact, solving (16) by the IMEX-Euler method in vector form we have the sequence of (small) linear systems

$$\tilde{A}\tilde{\mathbf{u}}_{k+1} = \tilde{B}\tilde{\mathbf{u}}_k \quad k = 0, \dots, n_t - 1 \tag{17}$$

where I_r denotes the identity matrix of order r, $\tilde{A} = I_r - d h_t A_r$ and $\tilde{B} = (1 + h_t\alpha) I_r$ have dimension $r \ll n$. Finally, we approximate the numerical solution of the full model (13) with a projection, that is $\mathbf{u}_k \approx \bar{\mathbf{u}}_k = V_r \tilde{\mathbf{u}}_k$, for each k.

Concerning the application of **DMD** we strictly follow the **Algorithm 1** to approximate the solution $\mathbf{u}(t)$. As for POD-Galerkin, DMD computes the reduced solution by solving a system of dimension $r \ll n$ as described in (11) in the least squares sense by using the QR solver. (In practice, in the next simulations this will be obtained by using the well known command "mldivide" (backslash) in Matlab.)

Both MOR techniques considered yield an approximation $\bar{\mathbf{u}}_k(r)$, $k = 0, \dots, n_t$, depending on the chosen truncation value r. Thus, in dependence of r, we can evaluate the error of each technique with respect to the numerical solution of the full model. To this aim we consider the errors in the Frobenius and maximum norms at the final time given by:

$$\begin{cases} e_F(r) = \|\mathbf{u}_{n_t} - \bar{\mathbf{u}}_{n_t}\|_F \\ e_{max}(r) = \|\mathbf{u}_{n_t} - \bar{\mathbf{u}}_{n_t}\|_\infty, \end{cases} \tag{18}$$

where $\mathbf{u}_{n_t} = vec(Z_{n_t})$ is calculated by solving (15) and $\bar{\mathbf{u}}_{n_t}$ by solving (17), respectively.

3.2. Numerical results

We consider (12) with the diffusion and reaction parameters given by $d = 0.2$ and $\alpha = -1$, respectively. We discretize the domain $\Omega = [0, 1] \times [0, 1]$ with $n_x = n_y = 400$ mesh points, such that the full model has dimension $n = 160000$. We consider the timestep $h_t = 10^{-4}$ and the final time $T = 0.3$, such that $n_t = 3000$. In Figs. 1(a)–(b) we show the full model solution by the IMEX Euler method (in matrix form) Z_{n_t} at final time T_f (left plot) and the corresponding absolute error with respect to the exact known solution $u^*(x, y, T_f)$ (middle plot). We construct the snapshot matrix $S \in \mathbb{R}^{160000\times 3001}$ which columns are $\mathbf{u}_k = vec(Z_k)$, $k = 0, \dots, n_t$ and we plot in Fig. 1(c) its first 50 singular values that decay when r increases. We want to find a value of r that gives a good approximation of the solution along the time dynamics (with respect to the snapshot matrix). Therefore, for $r = 1, \dots, 20$, we evaluate the errors (18) of the reduced solutions by POD and DMD in the Frobenius and maximum norms with respect to the full model numerical solution. In Fig. 2, we report the behaviour of $e_F(r)$ (left plot) and $e_{max}(r)$ (right plot). For $r = 5$, in Fig. 3 we show the absolute error of the reduced solutions with respect to the exact PDE solution $u^*(x, y, T_f) = e^{(\alpha - 5\pi^2 d)T_f} u_0(x, y)$ at the final time obtained with the two MOR methods. Very similar good errors are obtained also for this small truncation value and are given by $e_F = 4.7666 \times 10^{-8}$, $e_{max} = 3.0218 \times 10^{-10}$ (POD Galerkin), $e_F = 4.1110 \times 10^{-8}$, $e_{max} = 2.2889 \times 10^{-10}$ (DMD).

Fig. 2 shows that the two MOR techniques have very similar performance: the errors decrease for increasing values of r and very small errors are obtained already for $r = 20$. In particular, POD-Galerkin and DMD reach almost the same accuracy for $r = 20$: $e_F \approx 10^{-11}$ and $e_{max} \approx 10^{-13}$.

Finally, we want to investigate the performance of the MOR techniques for new values of parameters d and α without changing the snapshot matrix and solving only the new corresponding reduced systems. In fact, this application is important in the perspective of solving the same model with several choices of its parameters. In this case, we apply only the POD-Galerkin projection because for this technique it is possible to solve directly the reduced problems for the new parameter values, instead for DMD, it would be necessary to recalculate the snapshot matrix. For $r = 5$, we compare the results in terms of errors at the final time T_f and execution time: in Table 1 we report CPU times (in seconds) and errors (Frobenius and maximum norms) with respect to the full model solution. It is worth noting that the CPU time for solving the full problem (13) for each parameter choice is about 509.57 s. Hence, the numerical results reported in Table 1 show that by applying the POD-Galerkin technique it is possible not only to approximate accurately the full model solution, but also to save computational time such that, in the worst case, we have the ratio (timefull)/(timePOD) $\approx$ 3400. In Fig. 4, we show the reduced solutions obtained for the new choice of parameters.

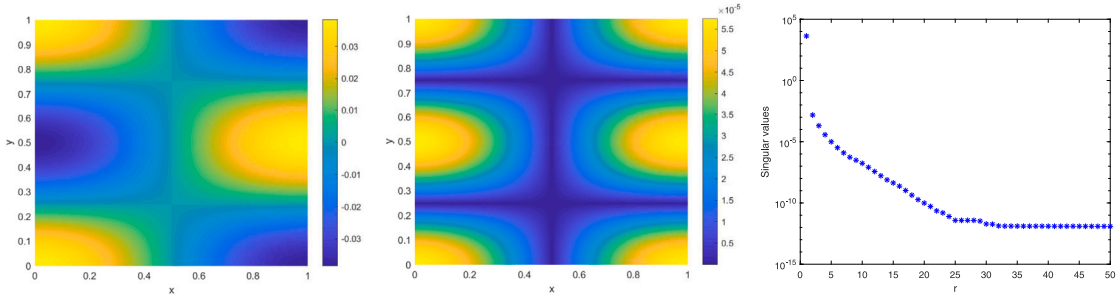


Fig. 1. Semilinear heat Eq. (12) with $d = 0.2$ and $\alpha = -1$. (a) Full model solution by the IMEX Euler method at final time $T_f = 0.3$; (b) absolute error of the full model solution wrt the exact solution at final time; (c) First fifty singular values of the snapshot matrix $S \in \mathbb{R}^{160000\times3001}$.

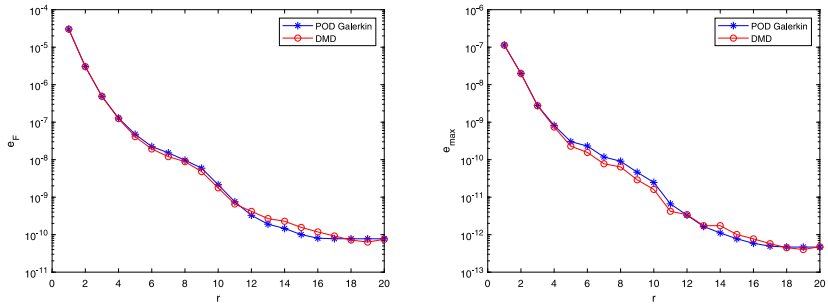


Fig. 2. Semilinear heat equation: comparisons of the errors e_F (left plot) and e_{max} (right plot) in (18) obtained by the POD-Galerkin and the DMD techniques. In all cases the errors decrease for increasing values of r and very small errors are obtained also for $r = 20$.

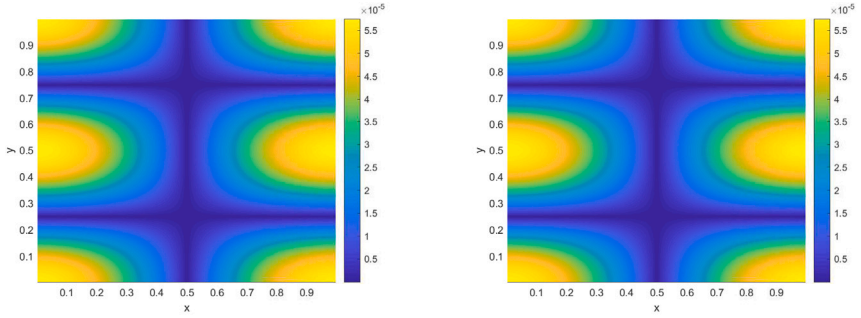


Fig. 3. Semilinear heat equation: for the truncation value $r = 5$, we show the absolute error of the reduced solutions by POD-Galerkin (left plot) and DMD (right plot) wrt the exact PDE solution at the final time $T_f = 0.3$. The corresponding errors with respect to the full model numerical solution are $e_F = 4.7666\times10^{-8}$, $e_{max} = 3.0218\times10^{-10}$ (POD Galerkin), $e_F = 4.1110\times10^{-8}$, $e_{max} = 2.2889\times10^{-10}$ (DMD). Very similar errors are obtained.

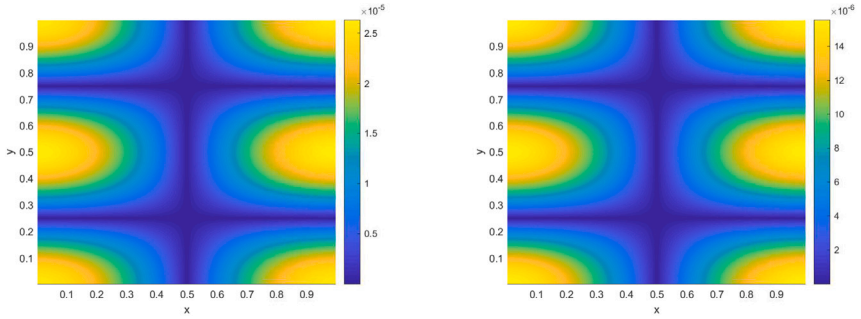


Fig. 4. Semilinear heat equation: POD-Galerkin with $r = 5$. We report the absolute error maps of the POD reduced solutions wrt the exact PDE solution for the indicated diffusion and reaction parameters. The corresponding maximum and Frobenius error norms (18) with respect to the full model are reported in Table 1.

Table 1
Semilinear heat equation: POD-Galerkin reduced solutions for $r = 5$ for indicated parameters. We report the errors at final time as defined in (18) and the CPU execution times (seconds). It is worth noting that the CPU time for solving the full problem (13) for each parameter choice is about 510 s and then a significant computational saving is obtained.

Method	Parameters	CPU time (s)	e_F	e_{max}
POD	$\alpha = -1,\ d = 0.2$	0.01034	4.7666×10^{-8}	3.0218×10^{-10}
$r = 5$	$\alpha = 0.04,\ d = 0.03$	0.028080	1.1984×10^{-7}	8.2488×10^{-10}
	$\alpha = -10,\ d = 0.09$	0.146731	2.9174×10^{-9}	$1{,}5953 \times 10^{-11}$

4. Reaction–diffusion PDE systems

We consider a reaction–diffusion PDE system with nonlinear reaction terms and zero Neumann boundary conditions, given by

$$\begin{cases} u_t = d_1 \Delta u + f(u,v), \quad (x,y) \in \Omega \subset \mathbb{R}^2,\ t \in [0,T] \\ v_t = d_2 \Delta v + g(u,v) \\ (\mathbf{n}\nabla u)_{|\partial\Omega} = (\mathbf{n}\nabla v)_{|\partial\Omega} = 0 \\ u(x,y,0) = u_0(x,y)\ ,\ v(x,y,0) = v_0(x,y). \end{cases} \tag{19}$$

where d_1, d_2 are the diffusion coefficients and $\Omega = [0, L_x] \times [0, L_y]$. The model kinetics f and g depend on a variety of parameters, therefore different kinds of solutions can be studied, such as travelling waves or oscillating dynamics Bozzini et al. (2019), Lacitignola et al. (2017, 2015), Sgura and Bozzini (2017) and Sgura et al. (2019).

In this paper, we are interested in the approximation of the so-called *Turing patterns* that are spatially non-homogeneous stationary solutions of (19). We start by considering the well-known Schnackenberg model as prototype of pattern formation obtained by the well-known Turing instability. The Turing instability, also known as *diffusion-driven instability*, arises when a homogeneous equilibrium that is stable in absence of diffusion becomes unstable in presence of diffusion and at the steady state attains characteristic patterns like spots, labyrinths, among others (see e.g. D'Autilia et al. (2020)).

The computational challenges in the numerical approximation of Turing patterns are related to: (i) long time integration to identify the final pattern as asymptotic solution of the PDE system and (ii) a large and finely discretized spatial domain to recognize the pattern structure. For these reasons, here we apply a MOR technique to show if it is able to reduce the computational costs, also in view of a mapping of the parameter space able to identify all the possible Turing patterns present in the RD model in exam.

In the second part of this section, we will consider the DIB model for electrodeposition that is a more realistic application for electrochemical phase formation modelling and, as shown in Lacitignola et al. (2015), presents Turing pattern formation able to describe the experimental shape of electrodeposits Lacitignola et al. (2017), Sgura and Bozzini (2017) and Sgura et al. (2019). Moreover, we want to emphasize that in these RD-PDE systems it is important to identify the pattern but also to reproduce the whole time dynamics. In fact, as discussed also in D'Autilia et al. (2020), in presence of Turing instability the time dynamics exhibits essentially two regimes, the so-called *reactivity* for small times in the transient regime and the asymptotic steady state for long times. For this reason, here we shall also study when a reduction technique is able to preserve these features. Hence, in this section, both for the Schnackenberg and the DIB models, first we will apply POD-Galerkin and DMD to identify the optimal $r-$ dimensional reduction spaces able to reproduce with sufficient accuracy the final pattern and, secondly, we will show the behaviour of the spatial mean $\langle v(t) \rangle$ of the solution, chosen as in D'Autilia et al. (2020) as an indicator for highlighting the time dynamics.

4.1. MOR techniques

As usual, the starting point both of POD and DMD is to build the snapshot matrices by using a numerical approximation of (19). For this goal, we apply the IMEX Euler scheme in matrix-oriented form, such that two sequences of Sylvester equations like (15) are solved as in D'Autilia et al. (2020) by the *rEuler* method. Hence, if n_x and n_y are the meshsizes used for the spatial semi-discretization of $\Omega = [0, L_x] \times [0, L_y]$, with total meshpoints $n = n_x n_y$, and n_t the number of time steps, we construct the snapshot matrices by considering the numerical approximations both for u and v by

$$S_u = \begin{bmatrix} | & | & \dots & | \\ \mathbf{u}_0 & \mathbf{u}_1 & \dots & \mathbf{u}_{n_t} \\ | & | & \dots & | \end{bmatrix}, \quad S_v = \begin{bmatrix} | & | & \dots & | \\ \mathbf{v}_0 & \mathbf{v}_1 & \dots & \mathbf{v}_{n_t} \\ | & | & \dots & | \end{bmatrix} \in \mathbb{R}^{n \times (n_t+1)} \tag{20}$$

where $\mathbf{u}_k = vec(Z_k^u)$, $\mathbf{v}_k = vec(Z_k^v)$ and Z_k^u, Z_k^u are the matrix solutions by the *rEuler* method at each time step. As second step, we need to compute a truncated SVD of the snapshot matrices (20), i.e. $S_u = V_r \Sigma_{r,u} W_r^T$ and $S_v = Q_r \Sigma_{r,v} P_r^T$, in order to obtain the orthogonal bases V_r and Q_r, respectively. We can generalize the semi-discretized ODE system (7) such that now POD-Galerkin yields the following reduced coupled ODE system for $\tilde{\mathbf{u}}(t), \tilde{\mathbf{v}}(t)$:

$$\begin{cases} \dot{\tilde{\mathbf{u}}}(t) = d_1 A_r \tilde{\mathbf{u}}(t) + \mathbf{f}_r(\tilde{\mathbf{u}}(t), \tilde{\mathbf{v}}(t)) \\ \dot{\tilde{\mathbf{v}}}(t) = d_2 B_r \tilde{\mathbf{v}}(t) + \mathbf{g}_r(\tilde{\mathbf{u}}(t), \tilde{\mathbf{v}}(t)) \\ \tilde{\mathbf{u}}(t_0) = \tilde{\mathbf{u}}_0 = V_r^T \mathbf{u}_0 \\ \tilde{\mathbf{v}}(t_0) = \tilde{\mathbf{v}}_0 = Q_r^T \mathbf{v}_0 \end{cases} \tag{21}$$

where $A_r = V_r^T \tilde{\Delta} V_r, \quad B_r = Q_r^T \tilde{\Delta} Q_r \in \mathbb{R}^{r \times r}$

$$\mathbf{f}_r(\tilde{\mathbf{u}}(t), \tilde{\mathbf{v}}(t)) = V_r^T \mathbf{f}(V_r \tilde{\mathbf{u}}(t), Q_r \tilde{\mathbf{v}}(t)), \quad \mathbf{g}_r(\tilde{\mathbf{u}}(t), \tilde{\mathbf{v}}(t)) = Q_r^T \mathbf{g}(V_r \tilde{\mathbf{u}}(t), Q_r \tilde{\mathbf{v}}(t)) \tag{22}$$

and $\tilde{\Delta}$ the Laplace discrete operator in Kronecker form defined in the previous section. The POD-Galerkin projection then approximates the solutions $\mathbf{u}, \mathbf{v}$ of the ODE *full model* associated to (19) after space semi-discretization, as follows

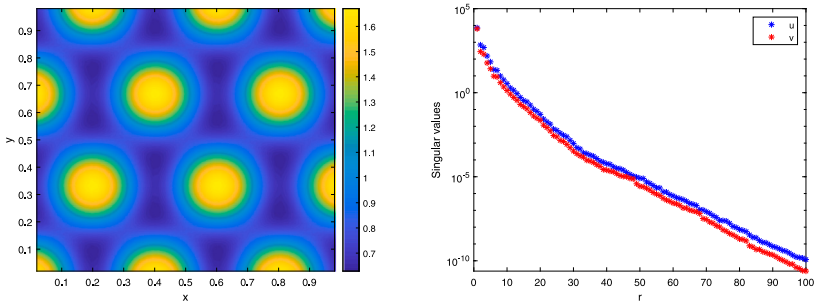
$$\begin{cases} \mathbf{u}(t) \approx \tilde{\mathbf{u}}(t) := V_r \tilde{\mathbf{u}}(t) \\ \mathbf{v}(t) \approx \tilde{\mathbf{v}}(t) := Q_r \tilde{\mathbf{v}}(t). \end{cases} \tag{23}$$

In the following computations, to solve the reduced system (21), we apply the IMEX-Euler method in vector form, given by

$$\begin{cases} (I_r - d_1 h_t A_r)\tilde{\mathbf{u}}_{k+1} = \tilde{\mathbf{u}}_k + h_t \mathbf{f}_r(\tilde{\mathbf{u}}_k, \tilde{\mathbf{v}}_k) \quad k = 0, \dots, n_t - 1 \\ (I_r - d_2 h_t B_r)\tilde{\mathbf{v}}_{k+1} = \tilde{\mathbf{v}}_k + h_t \mathbf{g}_r(\tilde{\mathbf{u}}_k, \tilde{\mathbf{v}}_k) \end{cases} \tag{24}$$

where I_r is the r-dimensional identity matrix. We remark that at each time step, we have to solve for $\tilde{\mathbf{u}}_{k+1}$ and $\tilde{\mathbf{v}}_{k+1}$ two (full) linear systems of dimension $r \ll n$ where $n = n_x n_y$, instead of large sparse n-dimensional linear systems required by the corresponding application of IMEX to the *full model*. Therefore, we will solve easily the reduced $r \times r$ linear systems in (24) by the vector approach, by using the direct solver ("backslash") in Matlab.

Concerning the application of DMD, we apply the Algorithm 1 using only one snapshot matrix S_u or S_v, depending on which component of the solution we want to approximate, i.e. $\mathbf{u}(t)$ or $\mathbf{v}(t)$ respectively. In this case, S_1 and S_2 in (9) are built only from S_u (or S_v) and consequently also the corresponding truncated SVD in Step 1 of the Algorithm 1. As for POD-Galerkin, DMD approximates the solutions by solving a system of dimension $r \ll n$ given in (10).



(a) (b)

Fig. 5. Schnakenberg model: (a) numerical solution $u(x, y)$ at final time $T_f = 2$; (b) a selection of the singular values of the snapshot matrices S_u and S_v for $1 \le r \le 100$.

4.2. Schnakenberg model

We first apply POD-Galerkin and DMD to the well known Schnackenberg model given by

$$\begin{cases} u_t = \Delta u + \gamma(a - u + u^2 v), \quad (x, y) \in \Omega = [0,1] \times [0,1],\ t \in [0, T_f] \\ v_t = d\Delta v + \gamma(b - u^2 v) \\ (\mathbf{n}\nabla u)_{|\partial\Omega} = (\mathbf{n}\nabla v)_{|\partial\Omega} = 0 \\ u(x, y, 0) = u_0(x, y)\ ,\ v(x, y, 0) = v_0(x, y). \end{cases} \tag{25}$$

Although this model is very simple, it represents an important example of patterns typically found in biological experiments (see e.g. Madzvamuse (2006)). The model parameters a, b, γ are positive constants and there is a unique stable homogeneous equilibrium (u_e, v_e), $u_e = a+b$ and $v_e = \frac{b}{(a+b)^2}$, that undergoes Turing instability in presence of diffusion.

4.2.1. Numerical results

In our numerical tests we consider the typical values of parameters $d = 10$, $\gamma = 1000$, $a = 0.1$ and $b = 0.9$ taken from the literature (see e.g. Madzvamuse (2006)) and the initial conditions $u_0(x, y) = u_e + 10^{-5}\text{rand(x,y)}$, $v_0(x, y) = v_e + 10^{-5}\text{rand(x,y)}$ small spatially random perturbations of (u_e, v_e). We discretize the domain Ω with $n_x = n_y = 50$ interior points, such that $n = n_x n_y = 2500$, we consider the time stepsize $h_t = 10^{-4}$ and $T_f = 2$, such that $n_t = 20000$. In Fig. 5 are shown the numerical solution at final time (left plot) and the singular value decay of the snapshot matrices $S_u, S_v \in \mathbb{R}^{2500\times 20001}$ (right plot) for $r = 1, \dots, 100$, $r \ll 2500$.

We choose three different values of r to find the solution with POD-Galerkin and DMD and then evaluate the errors defined in (18) comparing the full model and reduced solutions at final time for the variable $\mathbf{u}(t)$.

By choosing $r = 5$, we find:

$e_F = 0.0839$, $e_{max} = 0.0052$ (POD-Galerkin)

$e_F = 4.0522$, $e_{max} = 0.2220$ (DMD).

It is clear that POD-Galerkin gives a better solution, as shown by comparing Figs. 6(a) and 7(a).

By increasing the truncation value to $r = 25$ we obtain a better solution from DMD but not as accurate as POD-Galerkin (see Figs. 6(b) and 7(b)). In fact, for $r = 25$ we find the errors

$e_F = 1.4827 \times 10^{-5}$, $e_{max} = 1.2002 \times 10^{-6}$ (POD-Galerkin)

$e_F = 1.9787$, $e_{max} = 0.1356$ (DMD).

In order to further improve the approximation by DMD we choose $r = 70$, the pattern obtained is shown in Fig. 7(c). In this case we obtain the errors

$e_F = 3.8571 \times 10^{-10}$, $e_{max} = 2.4960 \times 10^{-11}$ (POD-Galerkin).

$e_F = 0.0440$, $e_{max} = 0.0021$ (DMD).

Therefore, for the Schnackenberg model we can deduce that POD-Galerkin performs better than DMD because requires a smaller truncation value r to catch the final pattern and exhibits smaller errors. To study the entire time dynamics and see if also the *reactivity zone* is well approximated in the transient regime, for the above values of r we study also the behaviour of the spatial mean obtained for the full and reduced solutions given by

$$\langle u(t) \rangle = \frac{1}{|\Omega|} \int_\Omega u(x, y, t) dx dy \approx mean(\mathbf{u}_k), \quad k = 0, \dots, n_t. \tag{26}$$

The results are reported in Fig. 8 for $r = 5, 25, 70$. By comparing the profiles for the different truncation values of r, we note that the DMD technique has an oscillating behaviour of $\langle u(t) \rangle$ which amplitude decreases with larger r, then this can explain why the largest value of $r = 70$ is needed to capture both the final pattern and the entire dynamics. On the other hand, it is evident that POD-Galerkin for $r = 5$ is already able to capture the asymptotic regime and the final pattern, but it requires $r = 25$ to follow the reactivity zone more accurately, in any case much smaller than $r = 70$ needed by DMD for similar behaviour.

4.3. DIB electrochemical model

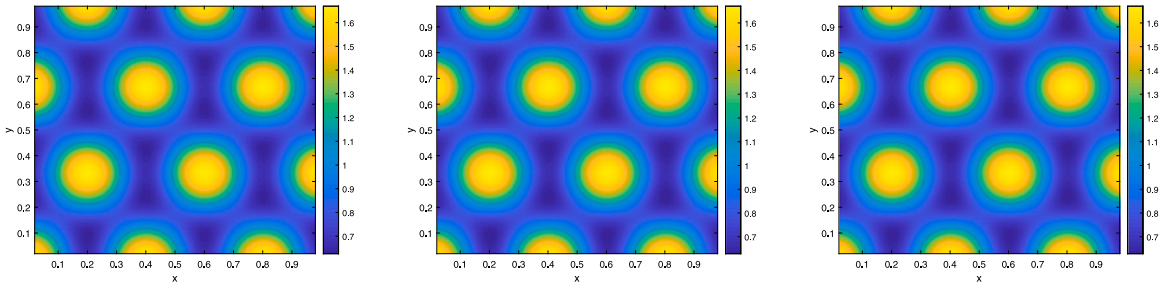
We focus here on the morphochemical model (briefly said DIB model as for example in Sgura et al. (2019)) that describes an electrodeposition process for metal growth, typical of recharge processes in batteries with metal electrodes. The nonlinear kinetics in (19) are given by

$$\begin{aligned} f(u, v) &= \rho\big(A_1(1 - v)u - A_2 u^3 - B(v - \alpha)\big), \\ g(u, v) &= \rho\big(C(1 + k_2 u)(1 - v)[1 - \gamma(1 - v)] - Dv(1 + k_3 u)(1 + \gamma v)\big). \end{aligned} \tag{27}$$

The unknown functions $u(x, y, t)$ and $v(x, y, t)$ represent the morphology of the metal deposit and the surface coverage with a functionally relevant adsorbate respectively, while the nonlinear source terms account for generation and loss of the relevant material. The physico-chemical meaning of the parameters is reported for example in Lacitignola et al. (2015). In particular, $(u_e, v_e) = (0, \alpha)$ is a homogeneous equilibrium for any choice of parameter values that is of interest for battery modelling and that undergoes Turing instability in presence of diffusion Lacitignola et al. (2015). Moreover, in Lacitignola et al. (2017) it has been proved that for a given parameter choice there exists an intrinsic pattern type that can only emerge if an *effective domain size* of integration is considered.

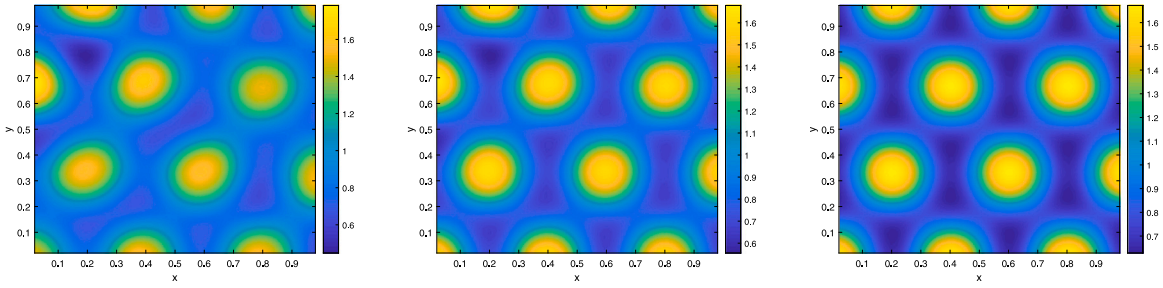
4.3.1. Numerical results

In our numerical tests we consider the parameter values $B = 66$, $C = 3$, $d = 20$, $\rho = 1$, $A_1 = 10$, $A_2 = 30$, $\alpha = 0.5$, $k_2 = 2.5$, $\gamma = 0.2$, $D = 2.4545$ and $k_3 = 1.5$ and initial conditions that are small spatially random perturbations of $(u_e, v_e) = (0, \alpha)$ as in D'Autilia et al. (2020). We discretize the domain $\Omega = [0, 20] \times [0, 20]$ with $n_x = n_y = 50$ interior points ($n = n_x n_y = 2500$), we consider time stepsize $h_t = 10^{-3}$ and final time $T_f = 100$, such that $n_t = 100000$. We are interested in finding by the MOR techniques an approximation of the Turing pattern solution at the final time and in tracking the entire time dynamics. In Fig. 9 are shown the numerical solution for v at final time T_f and the singular value decay of the snapshot matrices $S_u, S_v \in \mathbb{R}^{2500\times 100001}$ (right plot) until $r = 100$.



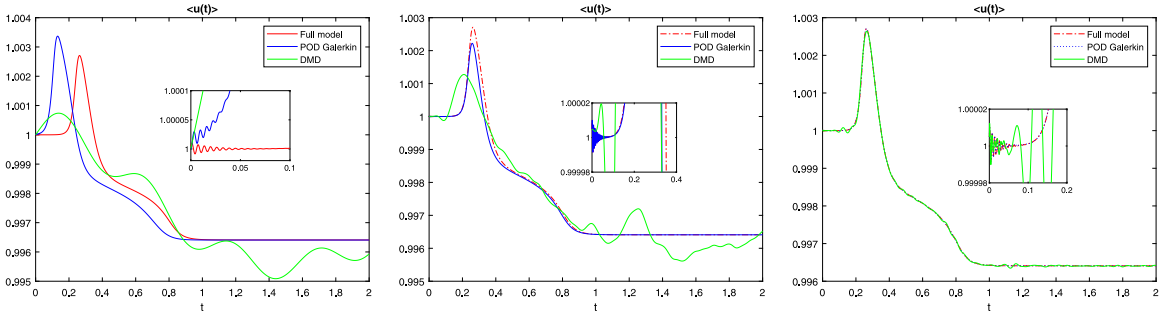
(a) $r = 5$ (b) $r = 25$ (c) $r = 70$

Fig. 6. Schnackenberg model: solutions at final time with POD-Galerkin.



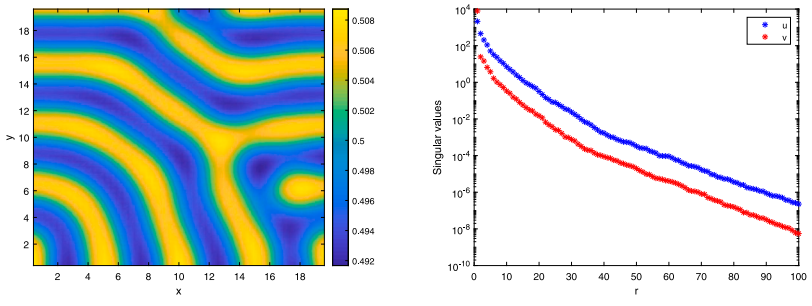
(a) $r = 5$ (b) $r = 25$ (c) $r = 70$

Fig. 7. Schnackenberg model: solutions at final time with DMD.



(a) $r = 5$ (b) $r = 25$ (c) $r = 70$

Fig. 8. Schnackenberg model: time dynamics of the spatial mean $\langle u(t) \rangle$ for the reduced solutions by POD-Galerkin and DMD compared with the full model. The final patterns corresponding to the same values of r are reported in Figs. 6 and 7.



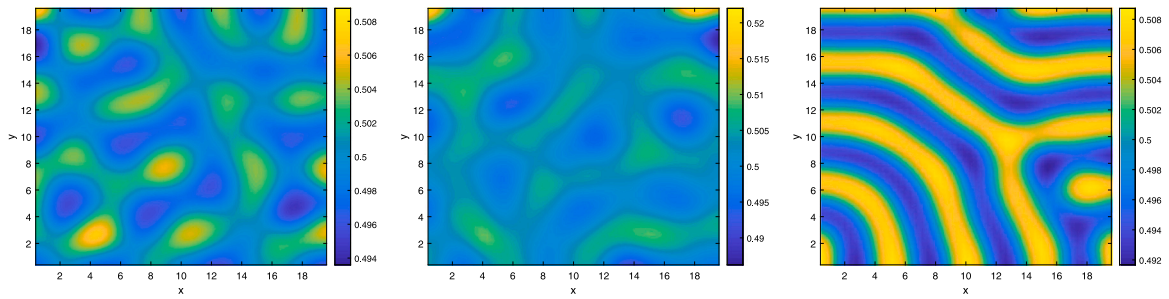
(a) (b)

Fig. 9. DIB model: (a) numerical solution v at final time $T_f = 100$ by the IMEX Euler method (*rEuler*); (b) singular values of the snapshot matrices $S_u, S_v \in \mathbb{R}^{2500\times100001}$ for $1 \le r \le 100$.

First of all, we choose the values of $r = 5, 20$ to find the reduced solutions with POD-Galerkin and DMD. The errors in the Frobenius and maximum norms (as in (18)) comparing full model and reduced solutions at the final time for the unknown variable v are given by:

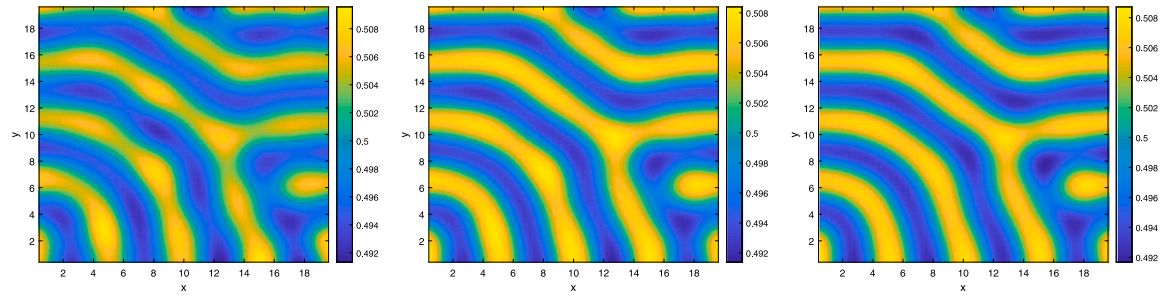
$r = 5$: $e_F = 0.2957$, $e_{max} = 0.0120$ (POD-Galerkin) and $e_F = 0.0568$, $e_{max} = 0.0028$ (DMD).

$r = 20$: $e_F = 0.2976$, $e_{max} = 0.0159$ (POD-Galerkin) and $e_F = 0.0186$, $e_{max} = 0.0016$ (DMD).



(a) $r = 5$ (b) $r = 20$ (c) $r = 100$

Fig. 10. DIB model: solutions for v at final time with POD-Galerkin for increasing values of r. The corresponding errors e_F, e_{max} with respect to the full model solution v are reported in the main text.



(a) $r = 5$ (b) $r = 20$ (c) $r = 100$

Fig. 11. DIB model: solutions v at final time with DMD for increasing values of r. The corresponding errors e_F, e_{max} with respect to the full model solution v are reported in the main text.

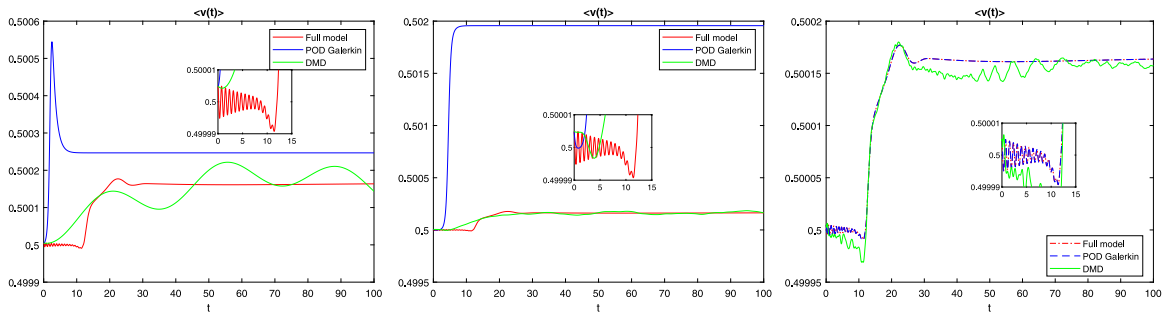
By comparing the final patterns in Figs. 10(a)–(b) and 11(a)–(b) and the above errors, it is clear that DMD gives a better solution for smaller values of r. This result becomes more clear looking at the entire dynamics of the spatial mean $\langle v(t)\rangle = \frac{1}{|\Omega|}\int_\Omega v(x,y,t)dxdy \approx mean(\mathbf{v}_k),\ k = 0,\dots,n_t$ in Figs. 12(a)–(b). For $r = 5, 20$, it is easy to see that the POD-Galerkin solution tends to another steady state, while DMD exhibits an oscillating behaviour around the dynamics of the full model, like in the Schnackenberg case. On the other hand, in Figs. 10(c) and 11(c) we show that by choosing a sufficiently large value $r = 100 \ll 2500$ POD-Galerkin gives a better approximation even more accurate than DMD for the same r, because as shown in Fig. 12(c) it is able to capture both the transient and the asymptotic regimes. In fact, in this case $e_F = 4.9120\times10^{-9}$, $e_{max} = 4.2719\times10^{-10}$ (POD-Galerkin) and $e_F = 0.0029$, $e_{max} = 3.7061\times10^{-4}$ (DMD).

Hence, for the DIB model we emphasize that for a labyrinth solution POD-Galerkin needs larger r to attain a good approximation, even if for the chosen $r = 100$ a highly accurate approximation (errors $\approx 10^{-9}$) is obtained with respect to DMD with the same projection size. A possible explanation is that, like for the Schnackenberg case, as shown in Fig. 12, the spatial mean of the DMD has an oscillatory behaviour for small values of r that tends to be slowly damped for larger value of r, such as $r = 100$. This oscillatory phenomenon is worth to be further investigated in a future research.

Finally, we apply the POD-Galerkin projection, and then solve the reduced model (24), for new choices of the parameters B, C, A_2 without changing the snapshot matrices obtained for the previous set $B = 66, C = 3, A_2 = 30$. We choose the case $r = 100$, such that the reduced pattern solution for this set is reported in Fig. 10(c). The POD-Galerkin solutions $\bar{\mathbf{v}}_{n_t}$ obtained at the final time of integration $T_f = 100$ for the new parameter sets (a) $B = 45, C = 2.9, A_2 = 30$ and (b) $B = 30, C = 3, A_2 = 1$ are shown in Fig. 13(a) and Fig. 13(b), respectively. The pattern in (a) belongs to the class of labyrinths and the pattern in (b) belongs to the class of *reversed spots (holes)*, as can be expected looking at the segmentation of the parameter space in Sgura et al. (2019) (Fig. 5).

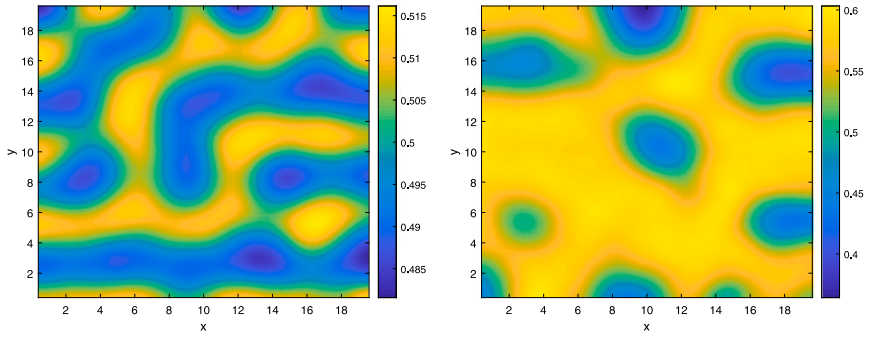
5. Conclusions

In this paper, we have considered a representative selection of model-reduction techniques, i.e. POD-Galerkin projection and DMD to approximate the numerical solutions of a selection of PDE models into a lower dimensional space, preserving the main features of the solutions. First of all, we have considered a semi-linear heat equation, 2D in space, with a known exact solution and we have shown how it is possible to save computational time by applying the POD-Galerkin projection for different choices of the parameters without recalculating the snapshot matrix. In fact, in the worst case, we have the ratio (timefull)/(timePOD) ≈ 3400. Finally, we have considered the reaction–diffusion Schnackenberg model and the DIB model for electrochemical phase formation that present Turing pattern formation. For these models we have shown that, as an alternative to solve the full model of dimension $n = 2500$, POD-Galerkin and DMD with a suitable low-dimensional projection ($r = 70$ for Schnakenberg and $r = 100$ for the DIB model), yield reduced models able to approximate carefully both Turing patterns at the steady state and the reactivity dynamics in the transient regime. In particular, DMD exhibits an oscillating behaviour during the time dynamics that tends to be damped for larger values of r (see Fig. 8 and Fig. 12) . Moreover, we have shown that, starting from a given parameter set and calculating once the snapshot matrices, the POD-Galerkin technique applied to the DIB model for new parameters is able to find reduced Turing patterns of different morphology, as predicted by the classification in Sgura et al. (2019) for the range of values considered. These encouraging results are a first step towards the application of more sophisticated model-reduction techniques to approximate in cheap and accurate way the Turing patterns of the DIB electrochemical PDE system of great interest for battery modelling. In fact, MOR techniques will be of particular interest in devising smart parameter identification techniques to compare numerical solutions with experimental data of batteries where it is required to solve the PDE model for different parameter choices. For example, in our group



(a) $r = 5$ (b) $r = 20$ (c) $r = 100$

Fig. 12. DIB model: time dynamics of the spatial mean $\langle v(t) \rangle$ for the reduced solutions compared with the full model for the values of $r = 5, 20, 100$. The corresponding final patterns are shown in Figs. 10 and 11.



(a) $B = 45, C = 2.9, A_2 = 30$ (b) $B = 30, C = 3, A_2 = 1$

Fig. 13. DIB model: POD-Galerkin $r = 100$ reduced solutions $\bar{\mathbf{v}}_{n_t}$ for indicated parameter choice. The snapshots have not been recalculated and correspond to those of the parameters $B = 66, C = 3, A_2 = 30$ and reduced final pattern in Fig. 10(c).

we are currently studying the case of constraints evaluations in classical approaches based on PDE constrained optimization problems or for training set construction in approaches based on machine learning techniques.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

IS is member of the InDAM-GNCS activity group and she acknowledges the PRIN 2017 research Project (No. 2017KL4EF3) “Mathematics of active materials: from mechanobiology to smart devices”.

References

Alla, A., Kutz, J.N., 2017. Nonlinear model order reduction via Dynamic Mode Decomposition. SIAM J. Sci. Comput. 39, 778–796.
Benner, P., Breiten, T., 2015. Two-sided projection methods for nonlinear model order reduction. SIAM J. Sci. Comput. 37 (2), B239–B260.
Benner, P., Gugercin, S., Willcox, K., 2015. A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Rev. 57 (4), 485–531.
Bozzini, B., Amati, M., Dobrovolska, T., Gregoratti, L., Krastev, I., Sgura, I., Taurino, A., Kiskinova, M., 2018. Depth-dependent scanning photoelectron microspectroscopy unravels the mechanism of dynamic pattern formation in alloy electrodeposition. J. Phys. Chem. 122 (28), 15996–16007.
Bozzini, B., Mele, C., D’Autilia, M.C., Sgura, I., 2019. Dynamics of zinc-air battery anodes: an electrochemical and optical study complemented by mathematical modelling. Metallurgia Italiana 111 (7–8), 33–40.
Brunton, S.L., Kutz, J.N., 2019. Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control. Cambridge University Press.
D’Autilia, M.C., Sgura, I., Simoncini, V., 2020. Matrix-oriented discretization methods for reaction–diffusion PDEs: comparisons and applications. Comput. Math. Appl. 79, 2067–2085.
Kramer, B., Willcox, K., 2019. Nonlinear model order reduction via lifting transformations and proper orthogonal decomposition. AIAA J. 57 (6), 2297–2307.
Lacitignola, D., Bozzini, B., Frittelli, M., Sgura, I., 2017. Turing pattern formation on the sphere for a morphochemical reaction–diffusion model for electrodeposition. Commun. Nonlinear Sci. Numer. Simul. 48, 484–508.
Lacitignola, D., Bozzini, B., Sgura, I., 2015. Spatio-temporal organization in a morphochemical electrodeposition model: Hopf and Turing instabilities and their interplay. European J. Appl. Math. 26 (2), 143–173.
Li, M., Lu, J., Ji, X., Li, Y., Shao, Y., Chen, Z., Zhong, C., Amine, K., 2020. Design strategies for nonaqueous multivalent-ion and monovalent-ion battery anodes. Nat. Rev. Mater. 1–19.
Liang, Y., Dong, H., Aurbach, D., Yao, Y., 2020. Current status and future directions of multivalent metal-ion batteries. Nat. Energy 1–11.
Madzvamuse, A., 2006. Time-stepping schemes for moving grid finite elements applied to reaction–diffusion systems on fixed and growing domains. J. Comput. Phys. 214, 239–263.
Peherstorfer, B., Willcox, K., 2015. Online adaptive model reduction for nonlinear systems via low-rank updates. SIAM J. Sci. Comput. 37 (4), A2123–A2150.
Sgura, I., Bozzini, B., 2017. XRF map identification problems based on a PDE electrodeposition model. J. Phys. D: Appl. Phys. 50 (15), 618–647.
Sgura, I., Lawless, A., Bozzini, B., 2019. Parameter estimation for a morphochemical reaction–diffusion model of electrochemical pattern formation. Inverse Probl. Sci. Eng. 27 (5), 618–647.
Sirovich, L., 1987. Turbulence and the dynamics of coherent structures. Parts I-II. Quart. Appl. Math. XVL, 561–590.
Volkwein, S., 2013. Model Reduction using Proper Orthogonal Decomposition. In: Lecture Notes, University of Konstanz.