Time, Space, and Space-Time Hybrid Clustering POD with Application to the Burgers’ Equation

Sami Sahyoun and Seddik M. Djouadi

Abstract—In this paper, we investigate new methods that make the Proper Orthogonal Decomposition (POD) more accurate in reducing the order of large scale nonlinear systems. The general framework is to apply POD locally to clusters instead of applying it to the global system. Each cluster contains relatively close in distance behavior within itself, and considerably far with respect to other clusters. We introduce three different clustering schemes in time, space and space-time. For time clustering, time snapshots of the solution are grouped into clusters where the solution exhibits significantly different features and a local basis is pre-computed and assigned to each cluster. Space clustering is done in a similar fashion for the space vectors of the solution instead of snapshots, and finally space-time clustering is applied through a hybrid clustering scheme that combines space and time behavior together. We apply our method to reduce a nonlinear convective PDE system governed by the Burgers’ equation for fluid flows over 1D and 2D domains and show a significant improvement over conventional POD.

I. INTRODUCTION

For nonlinear systems, Proper Orthogonal Decomposition (POD) is a model reduction technique that proved efficient performance when used to reduce models that approximate nonlinear infinite dimensional systems by high order finite dimensional systems, especially those who describe the dynamics of fluid flows [9].

Model reduction in general is an active research area in the last few decades because it is computationally difficult to design control laws for systems described by partial differential equations [19], [25], [8]. A very large number of states are needed to accurately capture the dynamics of such systems and this makes them unsuitable for control design [2], [5]. Conventionally the order of the system must be reduced before control law design can be done [1], [7].

In POD the snapshot method is usually used to create an ensemble of solutions with particular open loop control input data. The set is used to construct a set of POD basis modes [1]. It is well known that the modes maximize energy in mean square sense [9]. In our earlier paper [6], we showed that POD is optimal in a wider sense, which is of a distance minimization in spaces of Hilbert-Schmidt integral operators. These arguments are used to carry out model reduction, and determine its fidelity. From a given POD basis set, only the numbers of modes needed to capture a specified percentage of the total set energy are kept. This argument is problematic in the feedback control setting. Energy of POD modes generated from snapshots incorporating open-loop actuation might not correlate to the energy of the system under feedback control [1], [10].

The tools of POD have been used for some time as a model order reduction technique to achieve faster simulations of complex high dimensional systems. POD models of only a few dozen states have been shown to accurately capture the system dynamics of the full order system model of thousands of states [15], [16]. Model reduction using POD is often conducted to extract a relevant set of basis vectors. Sirovich suggested the method of snapshots as a way of determining these basis vectors without explicitly calculating the kernel necessary for POD [2].

The problem with POD (which is the motivation to write this paper) is that it fails to capture the nonlinear degrees of freedom in nonlinear systems, since it assumes that data belongs to a linear space and therefore relies on the Euclidean distance as the metric to minimize [3]. However, snapshots generated by nonlinear partial differential equations (PDEs) belong to manifolds for which the geodesics do not correspond in general to the Euclidean distance. A geodesic is a curve that is locally the shortest path between points. The global nonlinear manifold geodesic is difficult to be quantified in general but we show in this paper that it can be approximated efficiently by local linear Euclidean distances [17]. In [22], authors showed the poor performance of using global POD. They introduced a nonlinear model reduction approach via nonlinear projection framework. In [23], authors used the time domain partitioning approach and they presented a method for treating model reduction of evolution problems. This was realized by an adaptive partitioning of the time domain into several intervals and creating specialized reduced bases with limited size on each of the intervals.

In this paper, we investigate new methods that make POD more accurate. We will apply POD locally to clusters instead of applying it to the global system. Each cluster contains relatively close in distance behavior within itself, and considerably far with respect to other clusters. We introduce three different clustering schemes in time, space and space-time. For time clustering, time snapshots of the solution are grouped into clusters where the solution exhibits significantly different features and a local basis is pre-computed and assigned to each cluster. Space clustering is performed similarly for the space vectors of the solution instead of snapshots, and finally space-time clustering is applied through a hybrid clustering scheme that combines space and time behavior together. We apply our method to...
reduce a nonlinear convective PDE system governed by the Burgers’ equation for fluid flows over 1D and 2D domains.

This paper is organized as follows. In section (II), the Burgers’ equation in both 1D and 2D domains are numerically solved using finite difference techniques to construct the full order model for comparison purposes. In section (III) we reduce the full order system using Time Snapshots Clustering (TSP) POD. In section (IV) we show the performance when Space Vectors Clustering (SVC) POD reduced order bases are constructed based on space clustering. In section (V) we present the Space-Time Hybrid (STH) clustering that combines space and time behavior together and finally section (VI) is the conclusion and future work.

II. THE BURGERS’ EQUATION

The Burgers’ equation is a nonlinear PDE with a quadratic type nonlinearity. POD assumes Euclidean distance minimization which is not the case in the nonlinear Burgers’ equation, and that is the reason why POD fails to reduce the order of the system efficiently. The 1-D Burgers’ equation is given by [21]:

$$\frac{du}{dt} + u \frac{du}{dx} = \frac{d^2u}{dx^2}$$ (1)

where $x \in [a, b]$ and $t \in [0, T]$ for some initial condition $u(x, 0) = u_0(x)$ and boundary conditions $u(a, t) = u_a(t)$ and $u(b, t) = u_b(t)$. Discretization of the space domain into $N$ points and using finite difference approximations for the space derivatives yields:

$$\frac{du_i}{dt} + u_i \frac{u_i - u_{i-1}}{\Delta x} = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}$$ (2)

where $\lim_{\Delta x \to 0} \Delta x = \frac{1}{N}$ and $i = 0 \cdots N$.

Writing (2) in the matrix form we have:

$$\dot{u}(t) = Au + N(u) + Bu_{a,b} := f(u, u_{a,b})$$ (3)

$$u(0) = u_0$$

where $Au$ is the linear term $\frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}$ and $N(u)$ is the nonlinear term $u_i \frac{u_i - u_{i-1}}{\Delta x}$, $Bu_{a,b}$ is the boundaries term which will be used in the boundary control process and $u_0$ is the initial value. Full order system is solved in Matlab with $N = 500$, $x \in [0, 100]$, $t \in [0, 50]$, Dirichlet boundary condition $u_0(t) = 2$ and Newman boundary condition $\frac{du_{x_0}(t)}{dx} = 0$. The solution at three different times is shown in Fig. (1).

For the 2D Burgers’ equation system, (1) becomes:

$$\frac{du}{dt} + u \left( \frac{du}{dx} + \frac{du}{dy} \right) = \frac{1}{Re} \left( \frac{d^2u}{dx^2} + \frac{d^2u}{dy^2} \right)$$ (4)

where $x$ and $y$ are the spatial variables, $t$ is the time variable and $Re$ is a constant that is analogues to the Reynolds number that appears in the Navier Stokes equations. The 2D Burgers’ equation PDE shares the same nonlinearity as the Navier stokes PDE. It has the same quadratic nonlinearity and can be used to model incompressible fluid flows. 2D Burgers’ equation full order system is solved in Matlab with $Re = 300$, $u_0 = 0.1$, $x = 100$, $y = 100$, and $t = 30$.

III. TIME SNAPSHOTS CLUSTERING (TSC) POD

The time snapshots are grouped into clusters where the solution exhibits significantly different features. A cluster is a group that contains states which are close in some defined distance. Time Clustering is shown in Fig. (3). Local bases are pre-computed and assigned to each cluster. The set of pre-computed time snapshots are partitioned into $T$ clusters using K-means clustering algorithm discussed in the next subsection.

A. Clustering Using K-Means Algorithm

K-means algorithm groups together nearby locations according to their relative clustering distances. The clustering distance is defined as follows:

$$d(E_i, E_j) = \sqrt{(E_i - E_j)^T (E_i - E_j)}$$ (5)

Fig. 1. 1D Full order solution at three different times

Fig. 2. 2D Full order solution at t=30

$N = 2000$ and $Re = 300$, on the space domain shown in Fig. (2) which shows the solution at $t = 30$ seconds. This domain models the velocity of a fluid that is initially at rest all over the domain with a constant Dirichlet parabolic velocity at the left boundary that is maximum in the middle and zero in the top and bottom. The fluid passes around an obstacle to show the velocity behavior that the 2D Burgers’ equation models.
where \( d \) is the 2-norm distance between two time snapshots \( E_i \) and \( E_j \). These vectors contain the solution at times \( i \) and \( j \) respectively for all space location times.

Suppose we want to group \( N \) time snapshots \( \{E_i\}_{i=1}^N \) into \( T \) clusters \( \{\chi_j\}_{j=1}^T \), we first randomly choose \( T \) time snapshots as centroids \( \{E_{c_j}\}_{j=1}^T \). Then the distance between each time snapshot and the centroid is calculated as:

\[
d(E_i, E_{c_j}) = \sqrt{(E_i - E_{c_j})^T(E_i - E_{c_j})}
\]  

(6)

Let \( c_i \) be the argument of the minimum distance between \( E_i \) and \( E_{c_j} \):

\[
c_i = \arg \min_{j=1,\ldots,T} d(E_i, E_{c_j})
\]  

(7)

Then the new centroids would be:

\[
E_{c_j} = \frac{\sum_{i=1}^N 1_{c_i=j}E_i}{\sum_{i=1}^N 1_{c_i=j}}
\]  

(8)

where \( j = 1, \cdots, T \) and \( 1_{c_i=j} = 1 \) if \( c_i = j \) and zero otherwise. Then the last step is to assign each time snapshot \( E_i \) to the cluster \( \chi_{c_i} \).

B. Construction of TSC POD Bases

In the previous section, time snapshots were grouped into clusters. Reduced order bases are now computed for each cluster as follows:

Let the number of time snapshots in cluster \( k \) be \( N^k \), the \( N^k \times N^k \) correlation matrix \( L^k \) is defined by [1]:

\[
L^k_{i,j} = \langle E^k_i, E^k_j \rangle
\]  

(9)

is constructed, where \( \langle \cdot, \cdot \rangle \) denotes the Euclidian inner product of time snapshots \( E^k \). With \( R^k \) denotes the number of TSC POD modes to be constructed for cluster \( k \), the first \( R^k \) eigenvalues of largest magnitude, \( \{\lambda_i\}_{i=1}^{R^k} \), of \( L^k \) are found. They are sorted in descending order, and their corresponding eigenvectors \( \{v^k_i\}_{i=1}^{R^k} \) are calculated. Each eigenvector is normalized so that

\[
\|v^k_i\|^2 = \frac{1}{\lambda_i}
\]  

(10)

The orthonormal TSC POD basis set \( \{\phi^k_i\}_{i=1}^{R^k} \) is constructed according to [18]:

\[
\phi^k_i = \sum_{j=1}^{N^k} v^k_{i,j} E^k_j
\]  

(11)

where \( v^k_{i,j} \) is the \( j^{th} \) component of \( v^k_i \). The 1-D Burgers’ equation solution time snapshots were grouped into 4 clusters as shown in figure and the first four modes of each cluster are shown in Fig. (4).

The 2-D Burgers’ equation solution time snapshots were grouped into 8 clusters. The first four modes of cluster 1 are shown in Fig.(5).

![Fig. 3. TSC 4 Clusters](image)

![Fig. 4. First 4 modes of TSC POD in 4 clusters](image)

![Fig. 5. 2D First four modes of TSC POD in cluster 1](image)
The set of pre-computed solution space domain is partitioned into \( T \) clusters using K-means clustering algorithm. Space vector clustering is shown in Fig. (6).

Let the number of space vectors in cluster \( k \) be \( N_k \), the \( N_k \times N_k \) correlation matrix \( L_k \) is defined by [1]:

\[
L_k \left[ i,j \right] = \langle W_k^i, W_k^j \rangle 
\]

(13)

is constructed, where \( \langle \cdot \rangle \) denotes the Euclidean inner product of space vectors \( W_k^i \). With \( R_k \) denotes the number of SVC POD modes to be constructed for cluster \( k \), the first \( R_k \) eigenvalues of largest magnitude, \( \{ \lambda_i \}_{i=1}^{R_k} \), of \( L_k \) are found. They are sorted in descending order, and their corresponding eigenvectors \( \{ v_k^i \}_{i=1}^{R_k} \) are calculated. Each eigenvector is normalized so that

\[
\| v_k^i \|_2 = \frac{1}{\lambda_i} 
\]

(14)

The orthonormal SVC POD basis set \( \{ \phi_k^i \}_{i=1}^{R_k} \) is constructed according to [18]:

\[
\phi_k^i = \sum_{j=1}^{N_k} v_k^i j W_k^j 
\]

(15)

where \( v_k^i j \) is the \( j \)th component of \( v_k^i \). The 1-D Burgers’ equation solution space vectors were grouped into 4 clusters and the first four modes of each cluster are shown in figures (7).

The 2-D Burgers’ equation solution space vectors were grouped into 8 clusters. The first four modes of cluster 1 are shown in Fig. (8).

Now we have computed the clusters with their local reduced order bases, the last step is the projection to the full solution. The constructed local reduced order bases are projected to their corresponding locations in the full solution as follows:

\[
\{ U_{SVC} \}_{k=1}^{T} \approx \{ \sum_{i=1}^{R_k} \alpha_k^i \phi_k^i \}_{k=1}^{T} 
\]

(16)

V. SPACE-TIME HYBRID (STH) POD

In this section, the whole solution space and time domains is clustered using K-means algorithm. Note that clusters now contain space-time points instead of time snapshots or space vectors in sections III and IV respectively. However, these points are then reshaped to form either time or space vectors in each cluster. STH clustering is shown in Fig. (9). Note that this is not uniform for all clusters, meaning that some clusters could contain time snapshot vectors while others contain space vectors, we call them STH vectors. The optimum choice will be investigated in future work. Reduced order bases are computed for each cluster as follows: Let the number of STH vectors in cluster \( k \) be \( N_k \), the \( N_k \times N_k \) correlation matrix \( L_k \) is defined by [1]:

\[
L_k \left[ i,j \right] = \langle H_k^i, H_k^j \rangle 
\]

(17)
is constructed, where $\langle \cdot, \cdot \rangle$ denotes the Euclidean inner product of space vectors $H^k$. With $R_k$ denotes the number of STH POD modes to be constructed for cluster $k$, the first $R_k$ eigenvalues of largest magnitude, $\{\lambda_i^k\}_{i=1}^{R_k}$, of $L^k$ are found. They are sorted in descending order, and their corresponding eigenvectors $\{v_i^k\}_{i=1}^{R_k}$ are calculated. Each eigenvector is normalized so that

$$
\|v_i^k\|^2 = \frac{1}{\lambda_i^k}
$$

(18)

The orthonormal STH POD basis set $\{\phi_i^k\}_{i=1}^{R_k}$ is constructed according to [18]:

$$
\phi_i^k = \sum_{j=1}^{N^k} v_{i,j}^k H_j^k
$$

(19)

where $v_{i,j}^k$ is the $j^{th}$ component of $v_i^k$.

It is important to record the original ordering of snapshots because we need this in the projection process. The constructed local reduced order bases are projected to their corresponding locations in the full solution as follows:

$$
\{U_{STH}^T\}_{k=1}^{R} \approx \left\{ \sum_{i=1}^{R_k} \alpha_i^k \phi_i^k \right\}_{k=1}^{R}
$$

(20)

Performance Comparison between Reduced order models using all methods presented in this paper compared with global POD for the 1D Burgers’ equation is shown in Fig. (10) with the full number of states $N = 500$ is reduced to $R = 15$ where the space domain is $x \in [0, 100]$ and the time domain is $t \in [0, 50]$. The Figure shows the comparison at $t = 30$.

<table>
<thead>
<tr>
<th>Method</th>
<th>Error norm at $t = 30$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global POD</td>
<td>0.3080</td>
</tr>
<tr>
<td>TSC POD</td>
<td>0.0352</td>
</tr>
<tr>
<td>SVC POD</td>
<td>0.0012</td>
</tr>
<tr>
<td>STH POD</td>
<td>0.0009</td>
</tr>
</tbody>
</table>

As we notice from the reduction error results, Time Clustering shows better performance than POD, but the significant improvement comes from SVC and STH. This is justified by the fact that the nonlinear system (3) is constructed by the space discretization of the nonlinear PDE (1). This makes clustering in space more efficient. Combining space and time clustering in STH is better because solving the nonlinear ODE system is done numerically using the ODE45 Matlab solver, which brings the time numerical approximation issue as well, and this makes STH the winner in this paper.

VI. CONCLUSION AND FUTURE WORK

In this paper, the solution space of nonlinear PDEs is grouped into clusters where the behavior has significantly different features. We first solved the nonlinear problem for
the full solution, then we introduced three different clustering schemes in time, space and space-time. For time clustering, time snapshots of the solution were grouped into clusters where the solution exhibits significantly different features and a local basis was pre-computed and assigned to each cluster. Space clustering was done in a similar way for the space vectors of the solution instead of snapshots, and finally space-time clustering was applied using a hybrid clustering scheme that combined space and time behavior together. We applied our method to reduce a nonlinear convective PDE system governed by the Burgers’ equation for fluid flows over 1D and 2D domains and showed a significant improvement over conventional POD.

REFERENCES