Selecting dominant modes and reducing data dimensionality on GENE data with DMD

Dami Jung [a], Jae-Min Kwon[b], Eisung Yoon[a*]
[a]*Ulsan National Institute of Science and Technology, 50, UNIST-gil, Ulsan, 44919*
[b] *Korea Institute of Fusion Energy, 169-148 Gwahak-ro, Yuseong-gu, Daejeon, 34133*
[*]Corresponding author: esyoon@unist.ac.kr

1. Introduction

One of the challenges in fusion plasma simulation is to process high dimension of the data. As the dimensionality of the data increases the calculation requires extensive computing power, which leads to making real-time simulation using Virtual Reality [1] more difficult. Exploiting low dimensionality in complex systems has already been demonstrated to be an effective means in either computationally or theoretically reducing a given system to a more tractable form [2].

Among various methods for the dimensionality reduction techniques, we apply Proper Orthogonal Decomposition (POD) and Dynamic Mode Decomposition (DMD) to simulation data generated from a gyrokinetic plasma turbulence simulation code, GENE [3][4] to identify the dominant modes of the system from high-dimensional data. In order to check the accuracy of the model, the reconstructed matrix from the resulting model was compared to the original data to see if it contains a similar structure over time. The resulting model has a lower dimensionality than the original data and thus can be used for prediction, control, and analysis with high efficiency in terms of computational speed and resources.

2. Methods and Results

This section covers the data generation with GENE and methods used to reduce the dimensionality of the data; DMD and POD. In this work, the PyDMD library was used, including multi-resolution DMD (mrDMD).

2.1 Data generation

We have used a GENE sample input that conducts a nonlinear electrostatic collisionless local simulation with adiabatic electrons against Cyclone case [5] – Detailed physics parameters such as density, temperature, and their gradients can be referred to Ref. 5. In this simulation, we used a grid (nx0, nky0, nz0, nv0, nw0) = (96, 16, 16, 32, 8), where nx0, nky0, nz0, nv0, and nw0 are the number of radial grids, Fourier modes in the magnetic field-line index direction, magnetic field-following grids, parallel velocity grids, and magnetic moment grids, respectively. Time step ($\Delta t = 3.05 \times 10^{-2}$ transit time) was automatically decided by the code based on the embedded gyrokinetic eigenvalue solver. Data of our interest, electrostatic potential, was recorded at each grid point related to configuration space every 100 time steps for 1186 snapshots, and the finally obtained data has the size of (96, 16, 16) complex values for each time snapshot, which results in (96, 32, 16) 3D potential of a snapshot for 1186 snapshots in configuration space after inverse Fourier transform over radial (x) and field-line index (y) directions. From the transformed data, we use z=0 plane data only for testing purpose in this work.

2.2 Proper Orthogonal Decomposition (POD)

POD is a technique used for extracting the highest energy modes characterizing the fluid flow in fluid mechanics. The POD provides orthogonal basis for complex geometries based on empirical measurements [6]. POD shares the same algorithm with PCA (Principal Component Analysis), which computes the eigenvectors and eigenvalues of the covariance matrix of the data and selects a subset of the eigenvectors to represent the data in a lower-dimensional space. From data matrix $X \in \mathbb{C}^{m\times n}$ where m is the number of grid points in each snapshot and n is the number of snapshots, after subtracting mean of X from data matrix, compute the SVD (Singular Value Decomposition) and reconstruct the matrix with reduced rank numbers of eigenvectors sorted by size. A low-dimensional data matrix can be reconstructed by combining the mean of the data matrix and the previously reconstructed matrix.

The limitation of POD is that, in constructing the covariance matrix, it assumes that the mean and variance are sufficient statistics for capturing the underlying dynamics [7]. Therefore, POD may not capture some important dynamic features in complex system.

2.3 Dynamic Mode Decomposition (DMD)

The DMD method provides a decomposition of experimental data into a set of dynamic modes that are derived from snapshots of the data in time [2]. From data matrix $X \in \mathbb{C}^{m\times n}$ where n is the number of time step, let X' be the matrix one time step after X. DMD algorithm seeks the leading spectral decomposition of the best fit linear operator A in $X' \approx AX$ [8]. This linear operator A can be found as $A = \underset{A}{\operatorname{argmin}}\|X' - X\|_F = X'X^+$, using Frobenius norm and $X^+ \in \mathbb{R}^{m\times(n-1)}$ is a pseudo-inverse matrix.

DMD is data-driven method, which can extract dominant modes without requiring knowledge of the governing equation of the system. While POD finds a set of orthonormal basis of data matrix through SVD and reconstructs the matrix, DMD finds an eigenvalues and eigenvectors of best-fit linear operator A expressing the

evolution over time of a system. Since the motion of the plasma will vary through time, we expected that DMD would be more suitable for extracting the dominant modes for GENE data and reducing the dimensionality than POD.

2.3.1 Multi-resolution DMD

Multi-resolution DMD, one of the extensions of DMD, has much strength in complex system then DMD. The mrDMD recursively removes low-frequency content from a given collection of snapshots [9]. Thus, mrDMD is adaptable in the data with the dynamics of the system evolve at different rates. Since the underlying dynamics in GENE's data may not vary at a similar rate, mrDMD was also applied in this work.

2.4 Model construction

In this work, POD and DMD are used to reduce the dimensionality of original data. POD was applied by following the algorithm, and DMD model was constructed using a Python package PyDMD.

PyDMD is a Python package that uses Dynamic Mode Decomposition for a data-driven model simplification based on spatiotemporal coherent structures [10]. DMD model can be constructed by setting a DMD instance such as svd_rank, tlsq_rank, exact, opt where svd_rank is the rank for the truncation and tlsq_rank is the rank truncation computing Total Least Square. PyDMD has various functions and several modules such as mrDMD and SpDMD, and in this work, multi-resolution (mrDMD) module that compensated for the disadvantages of DMD was used.

2.5 Results

Because data generated from GENE is in Fourier space, the data was subject to inverse Fourier transform in the x-axis and y-axis directions respectively, and then applied to DMD, mrDMD and POD, each.

Fig. 1 is the integral of the data value to check the approximate accuracy between original data and reconstructed data from each model at same number of rank (svd_rank=100). As illustrated in Fig. 1, multi-resolution DMD (mrDMD) showed the best performance while DMD only follows the initial trend and POD remains constant.

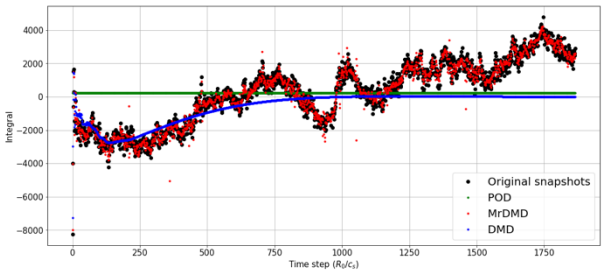

Fig. 1. Integral over time step of original data and reconstructed data from DMD, mrDMD and POD for rank=100.

To see the structure change of the electrostatic potential over time, we draw Fig. 2~4 using *coutourf* function in matplotlib python library.

Fig. 2 is the comparison between original data and reconstructed data in approximately beginning, middle, and end of total time. As shown below, mrDMD captures the most similar structures to the original data.

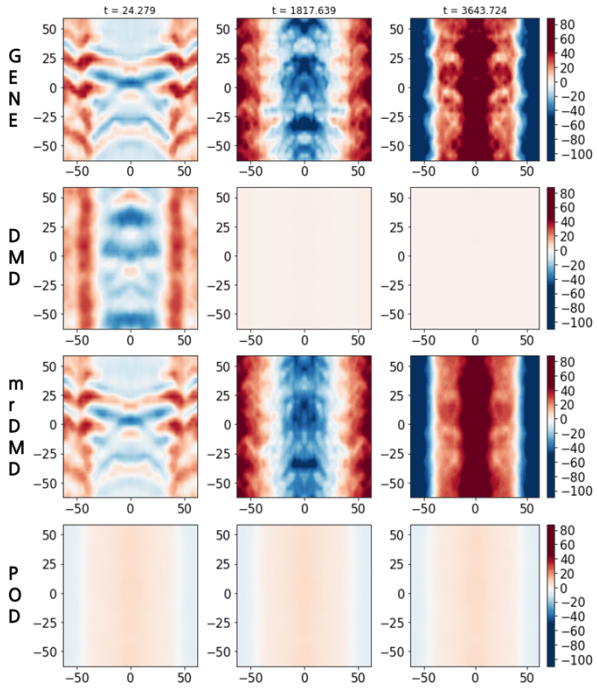

Fig. 2. Comparison of reconstructed data from DMD, mrDMD, POD with same rank(rank=100) and original data in approximate beginning, middle, and end portion of simulation time.

As DMD follows the trend of original data at the early part of simulation time In Fig. 1, DMD in the first column of Fig. 2, which is the early part of the simulation time, also shows a similar structure with the original data but still mrDMD captures the small structures.

In addition, we observe that as the number of ranks decreases, the resolution of the structures from the reconstructed data is also reduced. Fig. 3 compares the performance of mrDMD and POD with minimum ranks that capture fine-scale fluctuation structure from the original data. In the case of mrDMD, 20 rank out of 1866 was used with condition tlsq=100 and max_level=9, and a rank number of 1220 for POD was used among 3072 modes. In the case of mrDMD, if the rank number is further reduced from 10, the number of points out of the trend of the original data increases. In the case of POD, as the rank number decreases from 1220, the resolution

decreases significantly, and after 1207, it shows a constant trend like in Fig. 1.

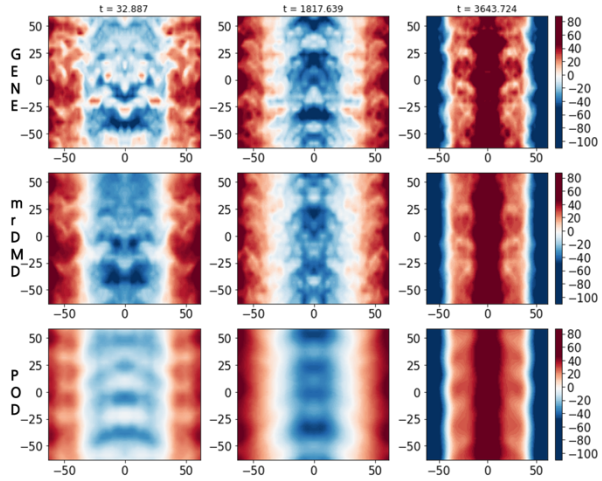

Fig. 3. Comparison of reconstructed data and original data in minimum ranks that showed good accuracy in mrDMD and POD (20 modes out of 1866 in mrDMD, 1220 modes out of 3072 in POD).

Fig. 4 is results of mrDMD and POD at rank 20 and 1220, respectively, and both follow a similar trend to the original data.

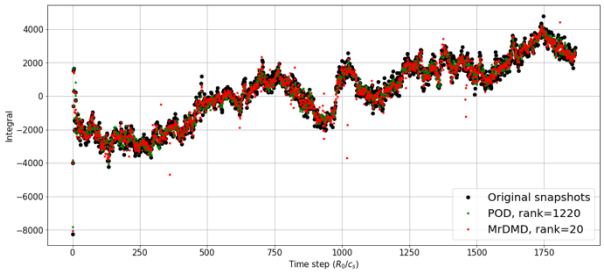

Fig. 4. Integral over time step of original data and reconstructed data from mrDMD of rank=20 and POD of rank=1220, respectively.

3. Conclusions

Through the comparison of three methods, DMD, POD and mrDMD, mrDMD showed the best performance in reducing the dimensionality of the plasma turbulence simulation data even with a smaller number of ranks while maintaining the structures of original data. We observed mrDMD can capture the dynamic features at spatiotemporally separate scales as designed and showed practical potential to reconstruct the structures of original data similarly. On the contrary, we found that DMD only follows the trend at the early portion of total time and converged to a constant value. Also, POD couldn't capture the dynamics of the data in our case.

Before closing this paper, we'd like to discuss potential future works. In analyzing high dimensional plasma data generated from large scale kinetic simulations, the techniques developed in this work will provide a useful tool to reduce and economize data processing. Based upon this, we plan to explore the dynamics governing the reduced plasma data. Traditionally, the reductions of plasma kinetic model have relied upon conventional fluid closures and some crude analogies connecting distant physical systems exhibiting similar behaviors. More systematic and quantitative analyses of plasma dynamics, which were enabled by the efficient data handling techniques and modern machine learning algorithms, will provide a new way to develop more accurate and robust reduced plasma models along with deeper insights into complex plasma physics.

Acknowledgment

This work was supported by the 2023 Research Fund(1.230028.01) of UNIST(Ulsan National Institute of Science & Technology). Also, this research was supported by R&D Program through the Korea Institute of Fusion Energy (KFE) funded by the Government funds, Republic of Korea (KFE-EN2341). Computing resources were provided on the KFE computer, KAIROS, funded by the Ministry of Science and ICT of the Republic of Korea (KFE-EN2341-9).

REFERENCES

[1] Jae-Min Kwon et al., Fusion Eng. Des. **184**, 113281 (2022)
[2] J. Nathan Kutz, *Data-Driven Modeling & Scientific Computation*, 1st ed. (OXF, 2013), 506.
[3] F. Jenko, W. Dorland, M. Kotschenreuther, and B. N. Rogers, Phys. Plasmas **7**, 1904 (2000)
[4] T. Gorler, X. Lapillonne, S. Brunner, T. Dannert, F. Jenko, F. Merz, and D. Told, J. Comput. Phys. **230**, 7053 (2011)
[5] A. M. Dimits, Phys. Plasmas **7**, 969 (2000)
[6] Steven L. Brunton, Bernd R. Noack, and Petros Koumoutsakos, Machine Learning for Fluid Mechanics, Annu. Rev. Fluid Mech, 2020.
[7] J. Nathan Kutz, *Data-Driven Modeling & Scientific Computation*, 1st ed. (OXF, 2013), 403.
[8] Steven L. Brunton, J. Nathan Kutz, *Data-Driven Science and Engineering*, 2nd ed, (Cambridge, 2022), 261.
[9] J. Nathan Kutz, Xing Fu, and Steven L. Brunton, Multiresolution Dynamic Mode Decomposition, Society for Industrial and Applied Mathematics, 2016.
[10] PyDMD contributors, "PyDMD's documentation", last modified on Mar 01, 2023, https://mathlab.github.io/PyDMD/