SPARSE NONNEGATIVE DYNAMIC MODE DECOMPOSITION

Naoya Takeishi[⋆] *Yoshinobu Kawahara*[†§] *Takehisa Yairi*[⋆]

[⋆]Department of Aeronautics and Astronautics,
The University of Tokyo
{takeishi,yairi}@ailab.t.u-tokyo.ac.jp

[†]The Institute of Scientific and Industrial Research, Osaka University
[§]RIKEN Center for Advanced Intelligence Project
ykawahara@sanken.osaka-u.ac.jp

ABSTRACT

Dynamic mode decomposition (DMD) is a method to extract coherent modes from nonlinear dynamical systems. In this paper, we propose an extension of DMD, *sparse nonnegative DMD*, which generates a nonlinear and sparse modal representation of dynamics. In particular, this makes DMD more suitable for video processing. We reformulate DMD as a block-multiconvex optimization problem to impose constraints and regularizations directly on the structures of the estimated dynamic modes. We introduce the results of experiments with synthetic data and a surveillance video dataset and show that sparse nonnegative DMD can extract part-based dynamic modes from video streams.

Index Terms— Dynamic mode decomposition, dynamical systems, sparse modeling, nonnegative decomposition

1. INTRODUCTION

Dynamic mode decomposition (DMD) [1, 2, 3] is a method to extract coherent modes from numerical data generated by nonlinear dynamical systems. Recently, DMD has been utilized in various areas including fluid mechanics [4], neuroscience [5], epidemiology [6], robotic control [7] and analyses of power systems [8]. DMD is also a useful tool for video processing because it can extract a set of modes from a video stream according to their dynamical characteristics, i.e., temporal decay rates and frequencies. In fact, Kutz *et al.* [9] utilized DMD for background/foreground separation of video streams by extracting static low-frequency dynamic modes as the background. In addition, Erichson *et al.* [10] proposed the use of DMD with compressed sensing for the fast background separation of video streams.

For video processing, DMD becomes more attractive if it can compute dynamic modes that take *nonnegative* values because such modes can be easily inspected and understood due to the inherent nonnegativity of video data. Moreover, obtaining *sparse* part-based dynamic modes is important for a meaningful representation of video streams. While there have been proposed many algorithmic variants of DMD including low-rank approximation [11, 12, 13, 14, 15] and noise correction [16], none of them can explicitly impose nonnegativity and sparsity on the structures of the estimated dynamic modes. In this paper, we propose to reformulate DMD as a block-multiconvex optimization problem so as to impose the nonnegativity constraint and the sparsity regularization on dynamic modes. Using the proposed sparse nonnegative version of DMD, we can decompose a video stream into part-based modes as shown in Section 4. This is analogous to the well-known results of the part-based image decomposition by nonnegative matrix factorization (NMF) [17], but an important difference of DMD from NMF is that DMD extracts modes related to the *dynamics* behind data.

The remainder of this paper is organized as follows. In Section 2, we briefly review the theory of DMD. In Section 3, the details of the proposed method are described. In Section 4, the experimental results are shown. This paper is ended with conclusions in Section 5.

2. BACKGROUND

DMD can be regarded as a data-driven algorithm for spectral decomposition of Koopman operator [18], with which the analysis of nonlinear systems is lifted to a linear regime. We briefly review this theory in the following, while we recommend to consult literature such as [19] for more details.

Consider a (possibly, nonlinear) dynamical system

$$\boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t), \quad \boldsymbol{x} \in \mathcal{M},$$

where $\boldsymbol{f} : \mathcal{M} \to \mathcal{M}$, $\mathcal{M}$ is the phase space, and t is the time index. The Koopman operator $\mathcal{K}$ on this system is defined as

$$\mathcal{K}g(\boldsymbol{x}) = g(\boldsymbol{f}(\boldsymbol{x})),$$

where $g : \mathcal{M} \to \mathbb{C} \in \mathcal{G}$ denotes *observables* in a function space $\mathcal{G}$. Note that $\mathcal{K}$ is linear but infinite-dimensional.

Here, suppose that there exists a subspace of $\mathcal{G}$ that is invariant to $\mathcal{K}$, and let us denote such subspace by $G \subset \mathcal{G}$. Further assume that G is finite-dimensional and a set of observables $\{g_1, \dots, g_n\}$ spans G, and let $\boldsymbol{g} = \begin{bmatrix} g_1 & \dots & g_n \end{bmatrix}^\mathsf{T} : \mathcal{M} \to \mathbb{C}^n$. Then, the one-step evolution of $\boldsymbol{g}$ is expressed as $K\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{g}(\boldsymbol{f}(\boldsymbol{x}))$, where K is the restriction of $\mathcal{K}$ to G and is a finite-dimensional linear operator.

Let us denote the eigenfunction of K by $\boldsymbol{\varphi} : \mathcal{M} \to \mathbb{C}^n$ and the eigenvalue by $\lambda \in \mathbb{C}$, i.e., $K\boldsymbol{\varphi}(\boldsymbol{x}) = \lambda\boldsymbol{\varphi}(\boldsymbol{x})$. If all

Algorithm 1: DMD [2, 20]

Data: $\boldsymbol{Y}_0$ and $\boldsymbol{Y}_1$ defined in Eq. (2)
Result: dynamic modes $\boldsymbol{w}$ and eigenvalues λ
$\boldsymbol{U}_r, \boldsymbol{S}_r, \boldsymbol{V}_r \leftarrow$ compact SVD of $\boldsymbol{Y}_0$;
$\tilde{\boldsymbol{A}} \leftarrow \boldsymbol{U}_r^{\mathsf{H}} \boldsymbol{Y}_1 \boldsymbol{V}_r \boldsymbol{S}_r^{-1}$;
$\tilde{\boldsymbol{w}}, \lambda \leftarrow$ eigenvector and eigenvalue of $\tilde{\boldsymbol{A}}$;
$\boldsymbol{w} \leftarrow \lambda^{-1} \boldsymbol{Y}_1 \boldsymbol{V}_r \boldsymbol{S}_r^{-1} \tilde{\boldsymbol{w}}$;
return $\boldsymbol{w}, \lambda$;

the eigenvalues are distinct, any values of $\boldsymbol{g}$ are expressed as

$$\boldsymbol{g}(\boldsymbol{x}) = \sum_{i=1}^{n} \boldsymbol{\varphi}_i(\boldsymbol{x}) \xi_i,$$

with some coefficients $\{\xi_i\}$. From the definitions, we have

$$\boldsymbol{g}(\boldsymbol{x}_t) = \sum_{i=1}^{n} \lambda_i^t \boldsymbol{w}_i, \quad \boldsymbol{w}_i = \boldsymbol{\varphi}_i(\boldsymbol{x}_0) \xi_i, \tag{1}$$

where $\boldsymbol{g}$ is decomposed into a sum of modes $\{\boldsymbol{w}_1, \dots, \boldsymbol{w}_n\}$, and the modulus and the argument of λ_i express the decay rate and the frequency of $\boldsymbol{w}_i$, respectively. In contrast to the classical modal decomposition of linear time invariant systems, the decomposition of the Koopman operator can be applied to nonlinear systems and it often focuses on the spatial distribution of the eigenfunctions.

DMD is a method to compute such decomposition from numerical data. Suppose we have data matrices:

$$\begin{aligned} \boldsymbol{Y}_0 &= \begin{bmatrix} \boldsymbol{g}(\boldsymbol{x}_0) & \dots & \boldsymbol{g}(\boldsymbol{x}_{m-1}) \end{bmatrix} \in \mathbb{C}^{n \times m} \quad \text{and} \\ \boldsymbol{Y}_1 &= \begin{bmatrix} \boldsymbol{g}(\boldsymbol{x}_1) & \dots & \boldsymbol{g}(\boldsymbol{x}_m) \end{bmatrix} \in \mathbb{C}^{n \times m}. \end{aligned} \tag{2}$$

The well-known algorithm of DMD [2, 20] is shown in Algorithm 1. If the component of $\boldsymbol{g}$ spans G, the dynamic modes computed by Algorithm 1 converge to $\boldsymbol{w}$ in Eq. (1) in the large sample limit. Note that the important premise of DMD is that we have $\boldsymbol{g}$ whose component spans an (approximately) invariant subspace. While there are some extension of DMD to address this issue (for example, the use of nonlinear basis functions [21], reproducing kernels [22], and delay coordinates [23, 24]), we simply assume that the data are generated with an observable that spans an approximately invariant subspace, as in the several previous studies on DMD.

3. PROPOSED METHOD

3.1. DMD as a block-multiconvex problem

We reformulate DMD as a block-multiconvex optimization problem, which enables us to impose constraints and regularizations directly on the structures of the estimated dynamic modes. In the sequel, we use the polar-coordinate expression for dynamic modes and eigenvalues, i.e.,

$$\begin{aligned} [\boldsymbol{w}_i]_d &= q_{i,d} \exp(j\theta_{i,d}), \quad q_{i,d}, \theta_{i,d} \in \mathbb{R}, \\ \lambda_i &= r_i \exp(j\phi_i), \quad r_i, \phi_i \in \mathbb{R}, \end{aligned} \tag{3}$$

Algorithm 2: DMD as a block-multiconvex problem

Data: $\boldsymbol{Y}$ defined in Eq. (5) and the number of modes n'
Result: dynamic modes q, θ and eigenvalues r, ϕ
1 $q_0, \theta_0, r_0, \phi_0 \leftarrow$ initialization by Algorithm 1;
2 **for** $k = 1, 2, \dots$ **do**
3 $\quad q_k \leftarrow \arg\min_q h(q, \theta_{k-1}, r_{k-1}, \phi_{k-1}; \boldsymbol{Y})$;
4 $\quad \theta_k \leftarrow \arg\min_\theta h(q_k, \theta, r_{k-1}, \phi_{k-1}; \boldsymbol{Y})$;
5 $\quad r_k \leftarrow \arg\min_r h(q_k, \theta_k, r, \phi_{k-1}; \boldsymbol{Y})$;
6 $\quad \phi_k \leftarrow \arg\min_\phi h(q_k, \theta_k, r_k, \phi; \boldsymbol{Y})$;
7 $\quad$ **if** *converge* **then return** $q_k, \theta_k, r_k, \phi_k$;
8 **end**

where j is the imaginary unit, and $[\boldsymbol{w}_i]_d$ denotes the d-th element of $\boldsymbol{w}_i$. In a matrix form,[1] we define

$$\begin{aligned} \boldsymbol{W} = \boldsymbol{Q} \odot \boldsymbol{\Theta}, \quad [\boldsymbol{Q}]_{i,d} = q_{i,d}, \quad [\boldsymbol{\Theta}]_{i,d} = \exp(j\theta_{i,d}), \\ \boldsymbol{V} = \boldsymbol{R} \odot \boldsymbol{\Phi}, \quad [\boldsymbol{R}]_{i,t} = r_i^{t-1}, \quad [\boldsymbol{\Phi}]_{i,t} = \exp(j(t-1)\phi_i), \\ \text{for} \quad i = 1, \dots, n', \quad d = 1, \dots, n, \quad t = 1, \dots, m+1, \end{aligned}$$

where $\boldsymbol{W} = [\boldsymbol{w}_1, \dots, \boldsymbol{w}_{n'}]$, $\boldsymbol{V}$ is a Vandermonde matrix with $\{\lambda_1, \dots, \lambda_{n'}\}$, and n' is a hyperparameter that denotes the effective number of dynamic modes. Following the decomposition in Eq. (1), the objective function to be minimized is

$$h(q, \theta, r, \phi; \boldsymbol{Y}) = \frac{1}{2} \left\| \boldsymbol{Y} - (\boldsymbol{Q} \odot \boldsymbol{\Theta})(\boldsymbol{R} \odot \boldsymbol{\Phi}) \right\|_{\mathrm{F}}^2, \tag{4}$$

where $\boldsymbol{Y}$ is the full-data matrix defined as

$$Y = \begin{bmatrix} \boldsymbol{g}(\boldsymbol{x}_0) & \dots & \boldsymbol{g}(\boldsymbol{x}_m) \end{bmatrix} \in \mathbb{C}^{n \times (m+1)}. \tag{5}$$

Since h is a block-multiconvex function [25] with regard to q, θ, r, and ϕ, its local minimum is easily obtained by a block coordinate descent. The pseudocode is shown in Algorithm 2. The update of each block (at Lines 3–6) can be performed with any solver, and we used the L-BFGS method [26] with gradients of h as follows:[2]

$$\begin{aligned} \frac{\partial h}{\partial q_{i,d}} &= -\operatorname{Re}\left[\boldsymbol{D} \boldsymbol{V}^{\mathsf{H}} \left[\bar{\boldsymbol{\Theta}} \right]_{i,d} \right]_{i,d}, \\ \frac{\partial h}{\partial \theta_{i,d}} &= -\operatorname{Im}\left[\boldsymbol{D} \boldsymbol{V}^{\mathsf{H}} \left[\bar{\boldsymbol{\Theta}} \right]_{i,d} \right]_{i,d} [\boldsymbol{Q}]_{i,d}, \\ \frac{\partial h}{\partial r_i} &= -\operatorname{Re}\left[\operatorname{tr}\left(\boldsymbol{D}^{\mathsf{T}} \bar{\boldsymbol{W}} \left(\boldsymbol{O}_i \odot \bar{\boldsymbol{\Phi}} \right) \right) \right], \\ \frac{\partial h}{\partial \phi_i} &= -\operatorname{Im}\left[\operatorname{tr}\left(\boldsymbol{D}^{\mathsf{T}} \bar{\boldsymbol{W}} \left(\boldsymbol{R} \odot \boldsymbol{M}_i \odot \bar{\boldsymbol{\Phi}} \right) \right) \right], \end{aligned}$$

where $\boldsymbol{D} = \boldsymbol{Y} - \boldsymbol{W}\boldsymbol{V}$. In addition, the i-th row of $\boldsymbol{O}_i \in \mathbb{R}^{n' \times (m+1)}$ is $\begin{bmatrix} 0 & 1 & 2r_i & \dots & m r_i^{m-1} \end{bmatrix}$ and the other elements are zero. The i-th row of $\boldsymbol{M}_i \in \mathbb{R}^{n' \times (m+1)}$ is $\begin{bmatrix} 0 & j & 2j & \dots & mj \end{bmatrix}$ and the other elements are zero.

[1] $\odot$ denotes element-wise multiplication.
[2] $\bar{\cdot}$ denotes complex conjugate and $\cdot^{\mathsf{H}}$ denotes conjugate transpose.

Algorithm 3: Sparse nonnegative DMD

Data: $\boldsymbol{Y}$ defined in Eq. (5) and the number of modes n'
Result: dynamic modes q and eigenvalues r, ϕ
1 $q_0, \theta_0, r_0, \phi_0 \leftarrow$ initialization by Algorithm 1;
2 $\theta_0 \leftarrow 0$;
3 **for** $k = 1, 2, \ldots$ **do**
4 $\quad q_k \leftarrow \arg\min_q f(q, r_{k-1}, \phi_{k-1}; \boldsymbol{Y})$;
5 $\quad r_k \leftarrow \arg\min_r f(q_k, r, \phi_{k-1}; \boldsymbol{Y})$;
6 $\quad \phi_k \leftarrow \arg\min_\phi f(q_k, r_k, \phi; \boldsymbol{Y})$;
7 $\quad$ **if** *converge* **then return** q_k, r_k, ϕ_k ;
8 **end**

3.2. Sparse nonnegative DMD

Using the reformulation, we can impose constraints and/or regularizations directly on variables q, θ, r, and ϕ. Our purpose is formulating *sparse nonnegative DMD* (SN-DMD) by imposing the nonnegativity constraint and the L1 regularization[3] on the dynamic modes, i.e., q is constrained to be nonnegative, θ is fixed to be zero, and a regularization term $\gamma|q|$ is introduced into the objective function. Formally, the new objective function f is given as

$$\begin{aligned} f(q, r, \phi; \boldsymbol{Y}) &= h(q, 0, r, \phi; \boldsymbol{Y}) + g(q), \\ g(q) &= \gamma \sum_{i=1}^{n'} \sum_{d=1}^{n} |q_{i,d}| + I_{q \geq 0}(q), \end{aligned} \tag{6}$$

where h is defined in Eq. (4), γ is a regularization parameter, and $I_{q \geq 0}(q)$ is an indicator function whose value is 0 when $q \geq 0$ and $+\infty$ otherwise. The pseudocode for SN-DMD using the block coordinate descent is shown in Algorithm 3. To solve the update step at Line 4, we utilize the proximal Newton-type method with an L-BFGS Hessian approximation [27]. The updates of the other quantities (at Lines 5–6) are computed in the same manner as in Algorithm 2.

4. EXPERIMENTS

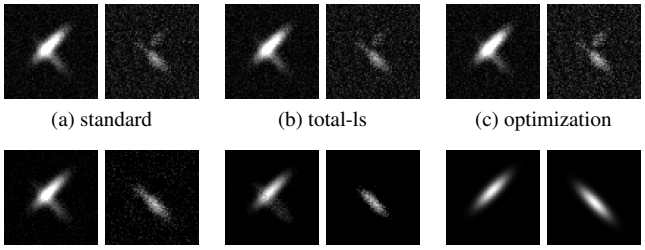
4.1. Mode extraction from noisy data

To investigate the performance of SN-DMD, we conducted an experiment with synthetic data as follows. First, we generated a sequence of noisy images $\{\boldsymbol{y}_t \in \mathbb{R}^{64 \times 64}\}$ by the following equations for $t = 0, \ldots, 15$:

$$\boldsymbol{z}_t = 0.99^t \boldsymbol{w}_1 + 0.9^t \boldsymbol{w}_2, \quad \boldsymbol{y}_t = \boldsymbol{z}_t + \boldsymbol{e}_t,$$

where $\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathbb{R}^{64 \times 64}$ were basis images shown in Figure 1 (f) ($\boldsymbol{w}_1$ on the left and $\boldsymbol{w}_2$ on the right), and $\{\boldsymbol{e}_t\}$ was a noise sequence whose element was generated independently by a zero-mean Gaussian with variance 10^{-2}. Obviously the dynamic modes of the noise-free sequence $\{\boldsymbol{z}_t\}$ are $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ with eigenvalues 0.99 and 0.9, respectively. However, it is not trivial how accurate we can estimate these dynamic modes and eigenvalues from noisy sequence $\{\boldsymbol{y}_t\}$.

[3]It is known that the nonnegativity itself encourages the sparsity in the setting of NMF [17], but in our case, we need the explicit L1 regularization because of the presence of the imaginary part.



(a) standard (b) total-ls (c) optimization
(d) nonneg (e) sparse-nonneg (f) ground truth

Fig. 1: (a–e) Estimated and (f) true dynamic modes (best viewed on a display). The left of each panel corresponds to $\boldsymbol{w}_1$ and the right corresponds to $\boldsymbol{w}_2$.

Table 1: Estimated and true eigenvalues.

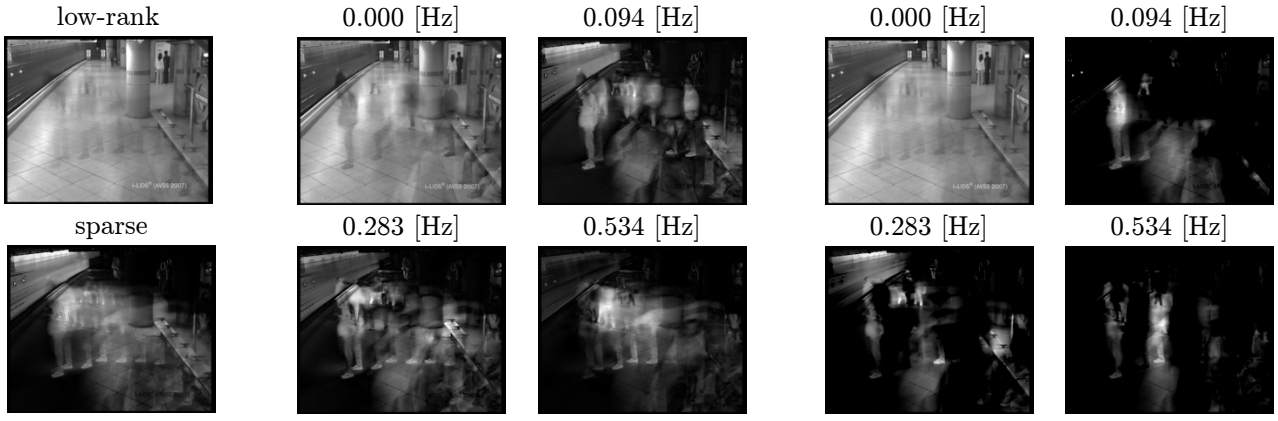
	λ_1	λ_2
standard	0.967	0.805
total-ls	0.969	0.825
optimization	0.967	0.805
nonneg	0.977	0.770
sparse-nonneg	**0.996**	**0.891**
ground truth	0.990	0.900

We input the noisy sequence to standard DMD (Algorithm 1, referred to as *standard*), total-least-squares DMD ([16], *total-ls*), DMD as the block-multiconvex problem (Algorithm 2, *optimization*), and the proposed method (Algorithm 3) with $n' = 2$. The proposed method is applied with two settings: one with only the nonnegativity (*nonneg*) and the other with both the sparsity and the nonnegativity (*sparse-nonneg*). We set $\gamma = 1$ without any intensive search. The estimation results are shown in Figure 1 and Table 1. The proposed method, *sparse-nonneg*, gives the best estimation among the methods listed above, in the sense that the estimated dynamic modes (Figure 1 (e)) are not contaminated with much noise and the estimated eigenvalues are the most accurate. Comparing the results of *nonneg* and those of *sparse-nonneg*, we can suppose that the sparsity plays an important role for estimation accuracy.

4.2. Extraction of part-based dynamic modes

Applying DMD to a video stream, we can expect that it will be decomposed into multiple modes with different temporal frequencies, wherein the "zero-frequency" mode corresponds to the background and the other modes to the foreground. The background/foreground separation has been studied intensively; one of the popular solutions is the low-rank/sparse decomposition such as robust PCA (RPCA) [29]. Moreover, Kutz et al. [9] utilized DMD for this task. Here, note that

Fig. 2: Summary of the first 150 frames (6 secs) of AVSS AB Hard dataset [28] (displayed every 15 frames, i.e., 0.6 secs).



(a) RPCA (b) standard DMD (c) SN-DMD

Fig. 3: Modal decomposition of AVSS AB Hard dataset video [28]. The upper panel of (a) and the upper-left panel of (b,c) correspond to the static part (i.e., background) of the video. The other panels correspond to the moving part (i.e., foreground).

the advantage of DMD is that it can distinguish not only the background and foreground, but also the components *within the foreground* according to their temporal frequencies. We address the task of foreground decomposition by DMD.

We used AVSS AB Hard dataset [28], which was a surveillance video of a platform with moving people and trains. As a preprocessing, we trimmed the first 251 frames of the original sequence that were showing a title credit, resized each frame into 192×240 pixels and extracted the first 150 frames as a dataset. The example frames are shown in Figure 2. To this dataset, we applied RPCA by fast principal component pursuit [30], standard DMD (Algorithm 1), and SN-DMD (Algorithm 3). The DMDs are computed with the specified number of modes $n' = 7$, and the regularization parameter was set $\gamma = 1$ without any intensive search.

The time-averages of the low-rank and sparse components extracted by RPCA are shown in Figure 3 (a), and the dynamic modes and temporal frequencies[4] extracted by DMDs are shown in Figures 3 (b,c). From the viewpoint of background/foreground separation, the background was successfully extracted as the low-rank component by RPCA and the zero-frequency dynamic modes by DMDs. In contrast, the other modes (the sparse component by RPCA and the nonzero-frequency dynamic modes) correspond to non-static parts of the video, i.e. the foreground. As can be seen, peaks (white regions) of the foreground modes correspond to the regions where some moving objects passed. In Figure 3 (c), the nonzero-frequency dynamic modes by SN-DMD are *part-based* in the sense that, for example, 0.094Hz-mode represents a standing person whereas 0.283Hz-mode represents the train and some other people. On the other hand, in Figure 3 (b), the nonzero-frequency dynamic modes by standard DMD have less distinctive spatial features; e.g., the region corresponding to the train is activated both in 0.094Hz-mode and in 0.283Hz-mode.

[4]Frequency f [Hz] is calculated by $f = \mathrm{Im}(\log \lambda)/(2\pi\Delta t)$ with (discrete-time) eigenvalue λ and time interval Δt [sec] between frames.

5. CONCLUSIONS

In this paper, we have reformulated DMD and proposed sparse nonnegative DMD (SN-DMD), which directly imposes the sparsity regularization and the nonnegativity constraint on the structures of the estimated dynamic modes. In particular, SN-DMD can decompose a video with multiple moving objects into a set of *part-based* dynamic modes, which is analogous to the well-known results of NMF [17].

Using the proposed formulation, it is possible to impose more complex regularizations and constraints. For example, structured regularizations like fused-lasso and group-lasso will be effective for highly-structured data. On the computational efficiency, while we adopted the simple block-coordinate descent, developing a more sophisticated solver is necessary for dealing with large-scale datasets.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Numbers JP15J09172, JP16H01548, JP26280086, and JP26289320.

References

[1] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, "Spectral analysis of nonlinear flows," *J. of Fluid Mechanics*, vol. 641, pp. 115–127, 2009.

[2] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *J. of Fluid Mechanics*, vol. 656, pp. 5–28, 2010.

[3] J. N. Kutz., S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems*, SIAM, 2016.

[4] P. J. Schmid, L. Li, M. P. Juniper, and O. Pust, "Applications of the dynamic mode decomposition," *Theoretical and Computational Fluid Dynamics*, vol. 25, no. 1, pp. 249–259, 2011.

[5] B. W. Brunton, L. A. Johnson, J. G. Ojemann, and J. N. Kutz, "Extracting spatial-temporal coherent patterns in large-scale neural recordings using dynamic mode decomposition," *J. of Neuroscience Methods*, vol. 258, pp. 1–15, 2016.

[6] J. L. Proctor and P. A. Eckhoff, "Discovering dynamic patterns from infectious disease data using dynamic mode decomposition," *Int. Health*, vol. 7, no. 2, pp. 139–145, 2015.

[7] E. Berger, M. Sastuba, D. Vogt, B. Jung, and H. B. Amor, "Estimation of perturbations in robotic behavior using dynamic mode decomposition," *Advanced Robotics*, vol. 29, no. 5, pp. 331–343, 2015.

[8] Y. Susuki and I. Mezić, "Nonlinear Koopman modes and power system stability assessment without models," *IEEE Transactions on Power Systems*, vol. 29, no. 2, pp. 899–907, 2014.

[9] J. N. Kutz, X. Fu, S. L. Brunton, and N. B. Erichson, "Multi-resolution decomposition for foreground/background separation and object tracking," in *Proc. of IEEE Int. Conf. on Computer Vision Workshop*, 2015, pp. 921–929.

[10] N. B. Erichson, S. L. Brunton, and J. N. Kutz, "Compressed dynamic mode decomposition for background modeling," *J. of Real-Time Image Processing*, pp. 1–14, 2016.

[11] K. K. Chen, J. H. Tu, and C. W. Rowley, "Variants of dynamic mode decomposition: Boundary condition, Koopman, and Fourier analyses," *Nonlinear Science*, vol. 22, no. 6, pp. 887–915, 2012.

[12] A. Wynn, D. S. Pearson, B. Ganapathisubramani, and P. J. Goulart, "Optimal mode decomposition for unsteady flows," *J. of Fluid Mechanics*, vol. 733, pp. 473–503, 2013.

[13] M. R. Jovanović, P. J. Schmid, and J. W. Nichols, "Sparsity-promoting dynamic mode decomposition," *Physics of Fluids*, vol. 26, no. 2, pp. 024103, 2014.

[14] C. Dicle, H. Mansour, D. Tian, M. Benosman, and A. Vetro, "Robust low rank dynamic mode decomposition for compressed domain crowd and traffic flow analysis," in *Proc. of the IEEE Int. Conf. on Multimedia and Expo*, 2016, pp. 1–6.

[15] P. Héas and C. Herzet, "Optimal low-rank dynamic mode decomposition," *arXiv:1701.01064v2*, 2017.

[16] S. T. M. Dawson, M. S. Hemati, M. O. Williams, and C. W. Rowley, "Characterizing and correcting for the effect of sensor noise in the dynamic mode decomposition," *Experiments in Fluids*, vol. 57, no. 3, pp. 42, 2016.

[17] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[18] I. Mezić, "Spectral properties of dynamical systems, model reduction and decompositions," *Nonlinear Dynamics*, vol. 41, pp. 309–325, 2005.

[19] I. Mezić, "Analysis of fluid flows via spectral properties of the Koopman operator," *Annual Review of Fluid Mechanics*, vol. 45, pp. 357–378, 2013.

[20] J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz, "On dynamic mode decomposition: Theory and applications," *J. of Computational Dynamics*, vol. 1, no. 2, pp. 391–421, 2014.

[21] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data-driven approximation of the Koopman operator: Extending dynamic mode decomposition," *J. of Nonlinear Science*, vol. 25, no. 6, pp. 1307–1346, 2015.

[22] Y. Kawahara, "Dynamic mode decomposition with reproducing kernels for Koopman spectral analysis," in *Advances in Neural Information Processing Systems*, vol. 29, pp. 911–919. 2016.

[23] Y. Susuki and I. Mezić, "A prony approximation of Koopman mode decomposition," in *Proc. of the IEEE 54th Annual Conf. on Decision and Control*, 2015, pp. 7022–7027.

[24] H. Arbabi and I. Mezić, "Ergodic theory, dynamic mode decomposition and computation of spectral properties of the Koopman operator," *arXiv:1611.06664v2*, 2016.

[25] Y. Xu and W. Yin, "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion," *SIAM J. on Imaging Sciences*, vol. 6, pp. 1758–1789, 2013.

[26] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming*, vol. 45, pp. 503–528, 1989.

[27] J. D. Lee, Y. Sun, and M. A. Saunders, "Proximal Newton-type methods for minimizing composite functions," in *Advances in Neural Information Processing Systems*, vol. 25, pp. 827–835. 2012.

[28] IEEE Int. Conf. on Advanced Video and Signal based Surveillance, "i-Lids dataset for AVSS 2007," `http://www.eecs.qmul.ac.uk/~andrea/avss2007_d.html`, 2007, [accessed 24-May-2017].

[29] E. J. Candés, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?," *J. of the ACM*, vol. 58, pp. 11:1–11:37, 2011.

[30] P. Rodriguez and B. Wohlberg, "Fast principal component pursuit via alternating minimization," in *Proc. of the 20th IEEE Int. Conf. on Image Processing*, 2013, pp. 69–73.