May 20th, 2019 @ SIAM DS 2019

Data-Driven Analysis of Koopman Spectrum with Reproducing Kernels

(Joint work with I. Ishikawa, M. Ikeda, Y. Hashimoto, K. Fujii and N. Takeishi)

- Institute of Mathematics for Industry, Kyushu University
- Center for Advanced Intelligence Project (AIP), RIKEN

Yoshinobu Kawahara

Data-driven Extraction of Dynamics

- Analytic Approach (conventional)

(real) Dynamic Process

Hypothesis Generation

Analysis / Prediction

Mathematical model

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x})$$

(or, $\boldsymbol{x}_{t+1} = \boldsymbol{F}(\boldsymbol{x}_t)$)

- **Data-driven Approach**

(real) Dynamic Process

Observation

Y

Data

***Machine learning** is a key mathematical framework!*

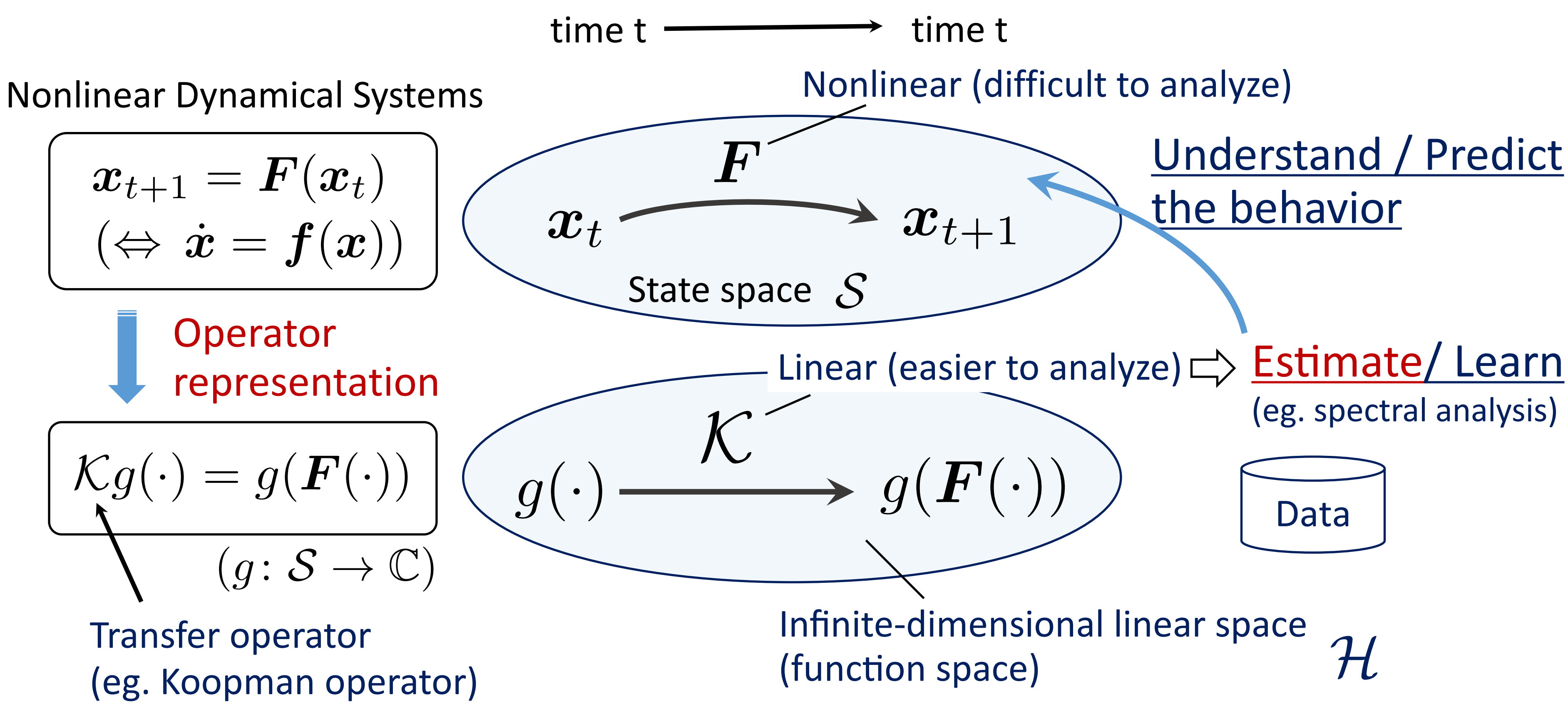
p (**f** | **Y**)

Inverse Problem (Learning)

- Understanding of data generation mechanism
- Use for prediction / simulation

Operator-theoretic Analysis

- In place of directly analyzing nonlinear dynamics **f**, we analyze a ***linear*** operator $\mathcal{K}$, such as the Koopman operator (Koopman 31) , that corresponds to the time evolution in the dynamics:



Reproducing Kernel and RKHS

- Reproducing kernel $k: \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a mathematical tool for analyzing data via the reproducing property:
  - Symmetricity: $k(\boldsymbol{x}, \boldsymbol{y}) = k(\boldsymbol{y}, \boldsymbol{x})$ for any pair $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$
  - Positive definiteness: $\sum_{i,j=1}^{n} c_i c_j k(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0$

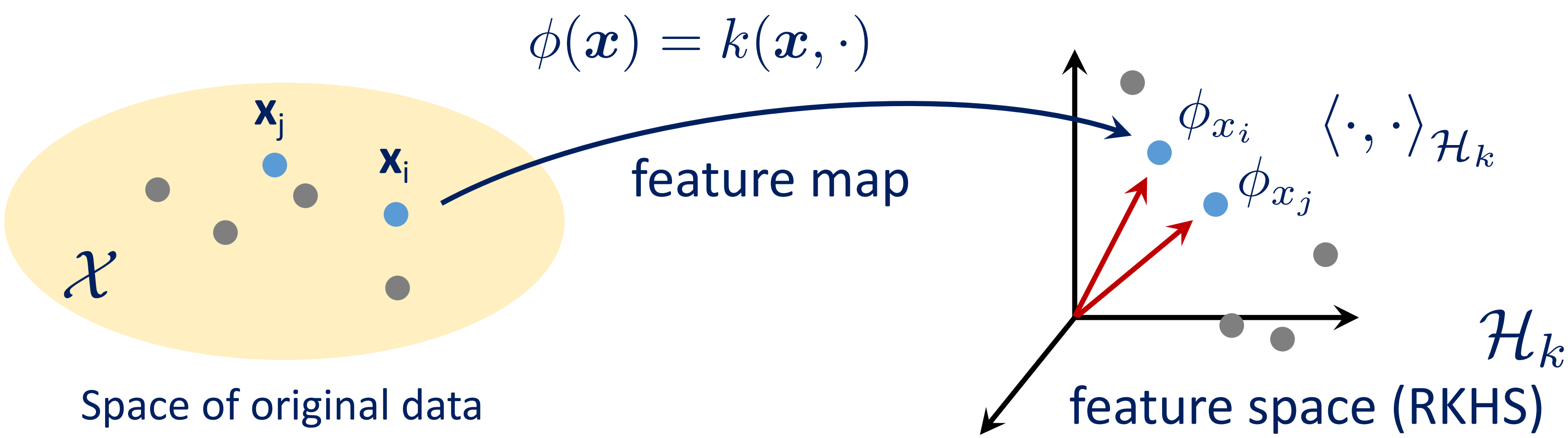
    for $n \in \mathbb{N}$, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n \in \mathcal{X}$, $c_1, \ldots, c_n \in \mathbb{R}$

ex.) RBF Gaussian kernel $k(\boldsymbol{x}, \boldsymbol{y}) = \exp\left(-c\|\boldsymbol{x} - \boldsymbol{y}\|^2\right)$

Reproducing Kernel and RKHS

- Reproducing kernel $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a mathematical tool for analyzing data via the reproducing property:
  - Feature map: $\phi\colon \mathcal{X} \to \mathcal{H}_k$ ($\phi(\boldsymbol{x}) = k(\boldsymbol{x}, \cdot)$)
  - An inner product in the feature space can be calculated as

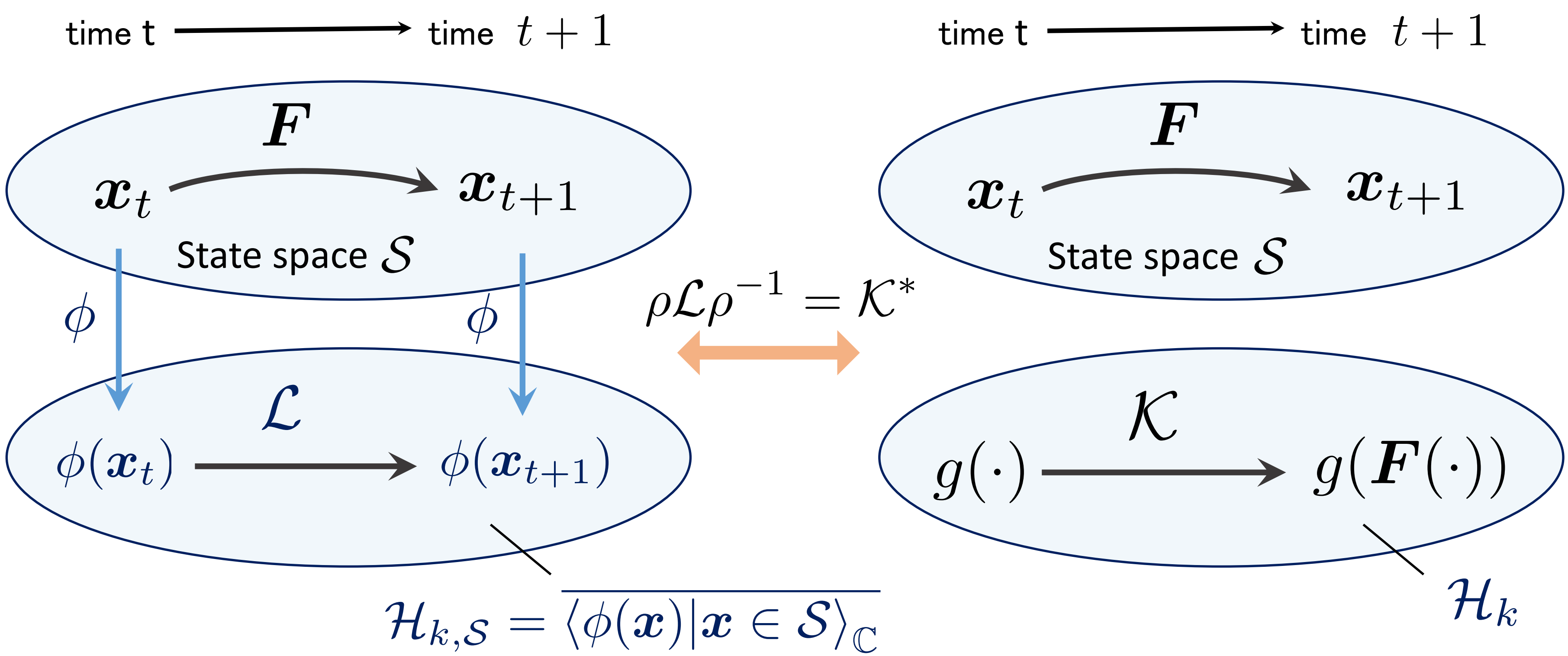
$$\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle_{\mathcal{H}_k} = k(\boldsymbol{x}, \boldsymbol{x}')$$



Perron-Frobenius Operator in RKHS

- Use the Perron-Frobenius Operator in RKHS $\mathcal{H}_k$ endowed with a reproducing kernel:

$$g \in \mathcal{H}_k \quad \Longrightarrow \quad \mathcal{L}\phi(\boldsymbol{x}) = \phi \circ \boldsymbol{F}(\boldsymbol{x}) \quad (\forall \boldsymbol{x} \in \mathcal{S})$$



($\rho: \mathcal{H}_{k,\mathcal{S}} \to H$: linear isomorphism)

(Kawahara, *NIPS*'16), (Ishikawa et al., *NeurIPS*'18)

Properties and Extensibility

- Applicable to wide range of dynamical systems without preparing observables (just need to choose (*suitable*) kernel functions).
- A DMD procedure (for the naïve case) is reduced to the equivalent one of Extended DMD (Williams+ 16) (for SVD-based implementation) (Kawahara, 16).

  => but, PF operators from kernels are not necessarily bounded.

- Deliver useful *extensibility* such as
  - Random systems with kernel-mean embeddings (Hashimoto et al., *under review*)
  - Structured observables (eg. Graph sequence) (Fujii & Kawahara, *Neural Networks* (in press))
  - Metric with PF operators in RKHSs (Ishikawa et al., *NeurIPS'18*)
  - and others …

Random Systems (1/3)

- Consider a nonlinear system with random noise:

$$X_{t+1} = h(X_t) + \xi_t$$

  – X and ξ are random vars. from measurable space to state space $\mathcal{S}$, and $h: \mathcal{S} \to \mathcal{S}$ is a map. ξ is assumed to be independent of X.

- Transform random variable X into probabilistic measure $X_* P$ (pushforward measure of P w.r.t. X defined by $X_P(A) = P(X^{-1}(A))$)

  metric

- Perron-Frobenius operator K is defined via kernel-mean embeddings:

$$\Phi : \mathcal{D}(\mathcal{S}) \to \mathcal{H}_k \text{ defined by } \mu \mapsto \int_{x \in \mathcal{S}} \phi(x) d\mu(x)$$

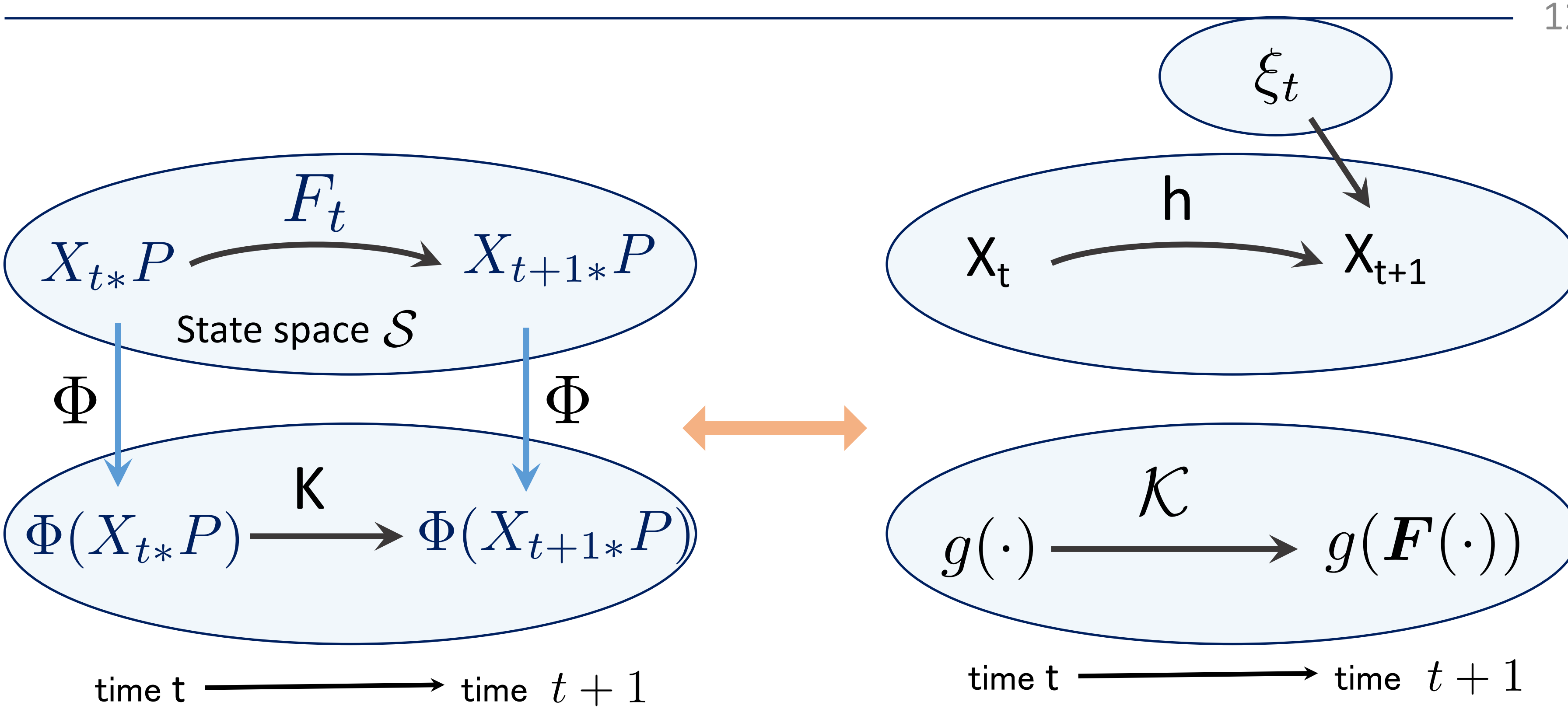
probability measure

by $$K\Phi(\mu) = \Phi(F_{t*}(\mu \otimes P))$$

($F_t : \mathcal{S} \times \Omega \to \mathcal{S}$ is defined by $(x, \omega) \mapsto h(x) + \xi(\omega)$)

(Hashimoto et al., *under review*)

Random Systems (2/3)



*) (Klus+ 17) considers PF operators in RKHS via kernel-mean embeddings in another way, and (Crnjarić -Žic et al., 2017) considers the Koopman operator for random system $x_{t+1} = \varphi(t, \omega, x_t)$, where $\varphi\colon \mathbb{Z}_{\geq 0} \times \Omega \times \mathcal{S} \to \mathcal{S}$.

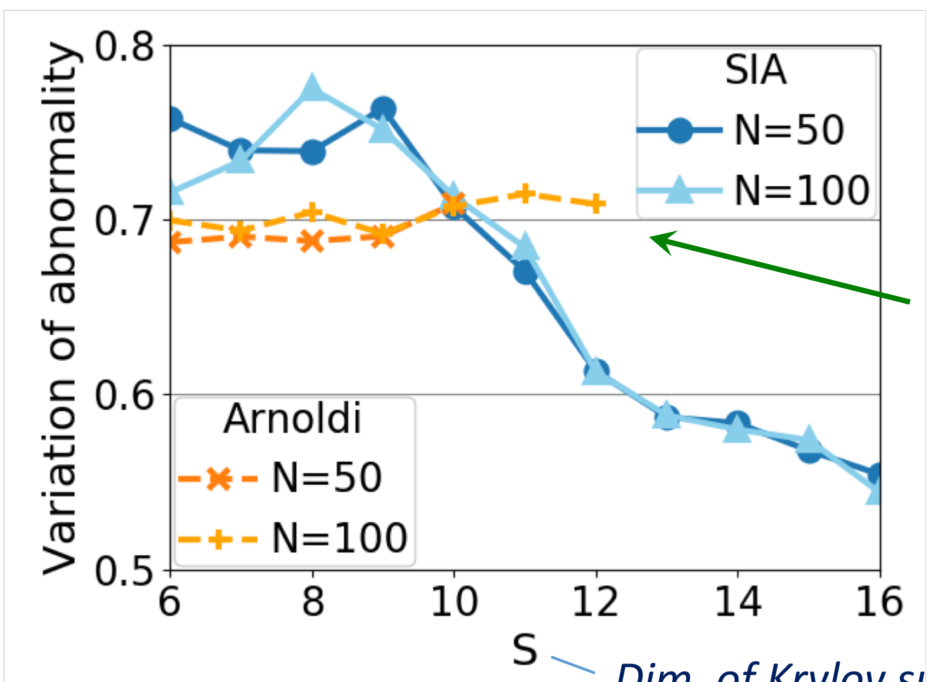
=> *The relations of our case with the above works can be explicitly described.*

(Hashimoto et al., *under review*)

Random Systems (3/3)

- PF operator K is not necessarily bounded. => *eg., if* k *is the Gaussian kernel, and* h *is nonlinear (even when* $\xi_t \not\to 0$
- We develop Shift-invert Arnoldi (SIA) method for the estimation:
  – Use Krylov subspace of $(\gamma I - K)^{-1}$ ($\gamma \notin \sigma(K)$) instead of K.

Spectrum of K



Dim. of Krylov subspace

(Standard) Arnoldi method tends to fail if S is large (not so robust).

(Hashimoto et al., *under review*)

Properties and Extensibility

- Applicable to wide range of dynamical systems without preparing observables (just need to choose (*suitable*) kernel functions).
- A DMD procedure (for the naïve case) is reduced to the equivalent one of Extended DMD (Williams+ 16) (for SVD-based implementation) (Kawahara, 16).

=> but, PF operators from kernels are not necessarily bounded.

- Deliver useful *extensibility* such as
  - Random systems with kernel-mean embeddings (Hashimoto et al., *under review*)
  - Structured observables (eg. Graph sequence) (Fujii & Kawahara, *Neural Networks* (in press))
  - Metric with PF operators in RKHSs (Ishikawa et al., *NeurIPS'18*)
  - and others …

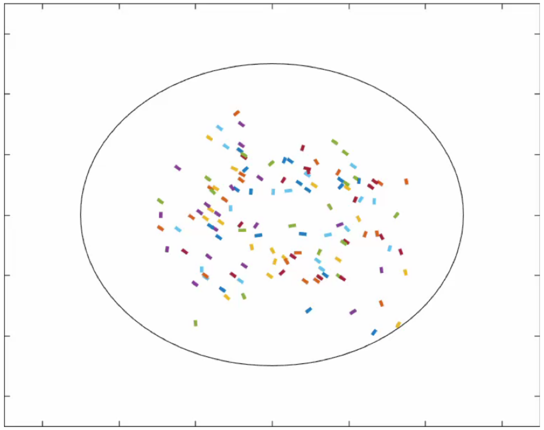
Extension to Structured Observations (1/4)

- DMD => Operator is defined in the space of observable g

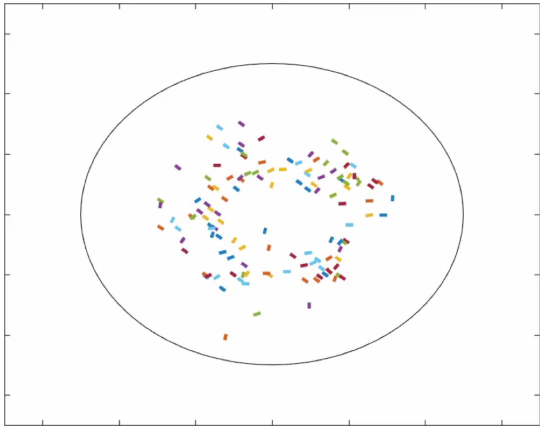
- *DMD for relations among observables when given structured data (such as sequences of graphs or distances)?*

ex) Extract the dynamics in collective motions such as fish school ?

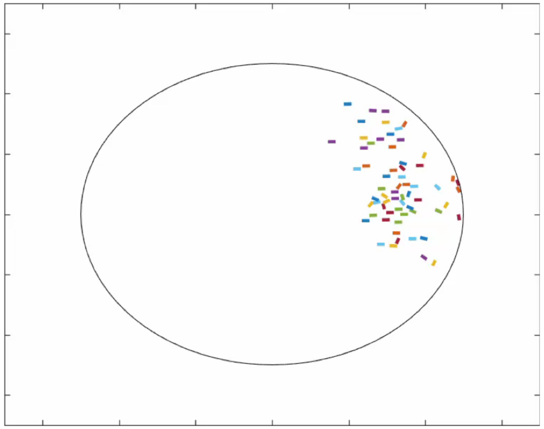
Swarm



Torus



Parallel



(Fujii et al., PLOS Computational Biology (2018))

Extension to Structured Observations (2/4)

Before: Dynamics on observables g_i

=> *Extend it to dynamics on relations among observables*

Use vector-valued kernels

Obs. func. $\boldsymbol{g} = [g_1, \ldots, g_p],\ g_i \in \mathcal{H}_k$ → $\boldsymbol{g} \in \mathcal{H}_{\boldsymbol{K}}$

RKHS

RKHS with $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$

Rep. prop.: $\phi(\boldsymbol{x}) = k(\boldsymbol{x}, \cdot)$

$f(\boldsymbol{x}) = \langle f, \phi(\boldsymbol{x}) \rangle$

RKHS with $\boldsymbol{K} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}^{m \times m}$

Rep. prop.: $\phi_{\boldsymbol{c}}(\boldsymbol{x}) = \boldsymbol{K}(\boldsymbol{x}, \cdot)\boldsymbol{c}$

$\boldsymbol{f}(\boldsymbol{x})^\top \boldsymbol{c} = \langle \boldsymbol{f}, \boldsymbol{K}(\boldsymbol{x}, \cdot)\boldsymbol{c} \rangle$

Ex.) Regard a graph sequence $\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_T$ as realizations of covs. in $\boldsymbol{g}(\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{x}), \boldsymbol{K}(\boldsymbol{x}, \boldsymbol{x}))$

→ $\boldsymbol{g} \in \mathcal{H}_{\boldsymbol{K}}$ (where $\mathcal{H}_{\boldsymbol{K}}$ is the RKHS endowed with covariance matrix **K**)

(Fujii & Kawahara, Neural Networks (2019))

Extension to Structured Observations (3/4)

- Determine a kernel func. **K** ($\phi_{\boldsymbol{c}}$: feature map)
- Finite length graph sequence (seq. of adj. matrices) **A**$_1$, **A**$_2$, …, **A**$_T$

(define $\Phi_0 := [\phi_{\boldsymbol{c}}(\boldsymbol{x}_0), \phi_{\boldsymbol{c}}(\boldsymbol{x}_1), \cdots, \phi_{\boldsymbol{c}}(\boldsymbol{x}_{T-1})]$, $\Phi_1 := [\phi_{\boldsymbol{c}}(\boldsymbol{x}_1), \phi_{\boldsymbol{c}}(\boldsymbol{x}_2), \cdots, \phi_{\boldsymbol{c}}(\boldsymbol{x}_T)]$)

1. Calculate an orthogonal basis in $\mathcal{H}_{\boldsymbol{K}}$: $\mathcal{U} := \Phi_0 \boldsymbol{M}$ $(\boldsymbol{M} \in \mathbb{R}^{T-1 \times p})$
2. Solve the LS prob. : $\boldsymbol{P} := \underset{\boldsymbol{P}' \in \mathbb{R}^{p \times p}}{\operatorname{argmin}} \frac{1}{T} \sum_{t=0}^{T-1} \left\| \mathcal{U}^* \phi_{\boldsymbol{c}}(\boldsymbol{x}_{t+1}) - \boldsymbol{P}'(\mathcal{U}^* \phi_{\boldsymbol{c}}(\boldsymbol{x}_t)) \right\|^2$

Basically can be performed by applying **Tensor DMD** (Klus+ 16)

3. Calculate the eigen-value / -vectors of **P** : $\boldsymbol{P}\hat{\boldsymbol{v}}_j = \hat{\lambda}_j \hat{\boldsymbol{v}}_j$
4. Obtain the decomposition : $\mathcal{U}^* \phi_{\boldsymbol{c}}(\boldsymbol{x}) = \sum_{j=1}^{p} \hat{\lambda}_j^t (\hat{\boldsymbol{\varphi}}_j(\boldsymbol{x}_0)^\top \boldsymbol{c}) \hat{\boldsymbol{v}}_j$

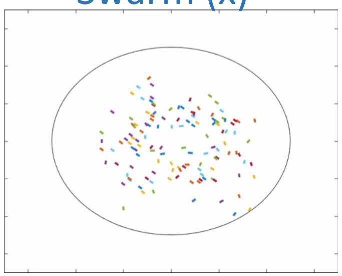
(where $\hat{\boldsymbol{\varphi}}_j(\cdot)^\top \boldsymbol{c} = \boldsymbol{\kappa}_j^*(\mathcal{U}^* \phi_{\boldsymbol{c}}(\cdot))$ ($\hat{\boldsymbol{\kappa}}_j$ is the left-eigenvector of **P**))

(Fujii & Kawahara, Neural Networks (2019))

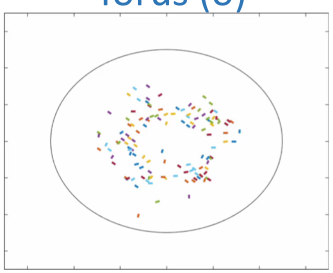
Extension to Structured Observations (4/4)

- Empirical example of the application to data from fish school simulation:
  - Data are ***the sequence of distance matrices*** among fishes.

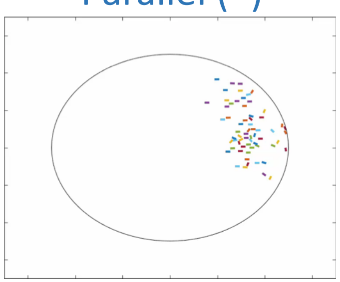
Swarm (x)



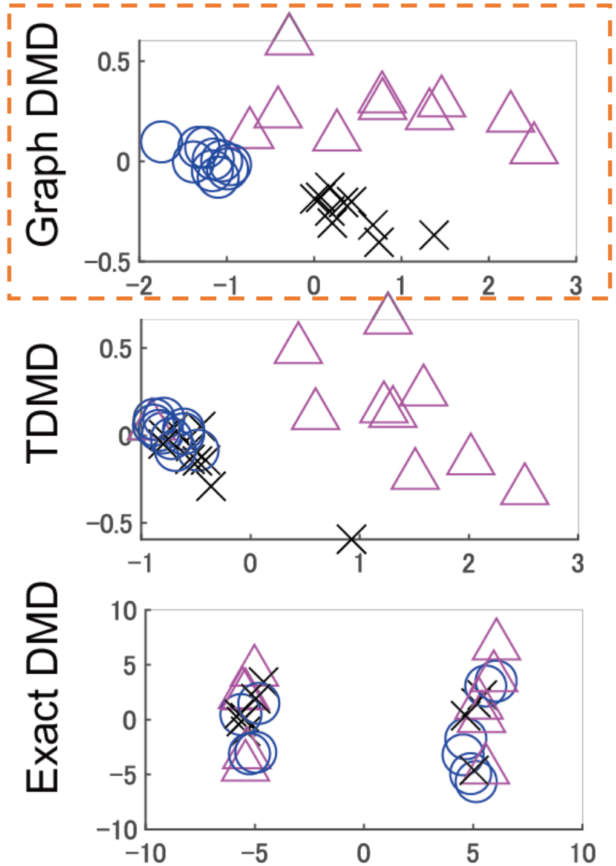
Torus (o)



Parallel (^)

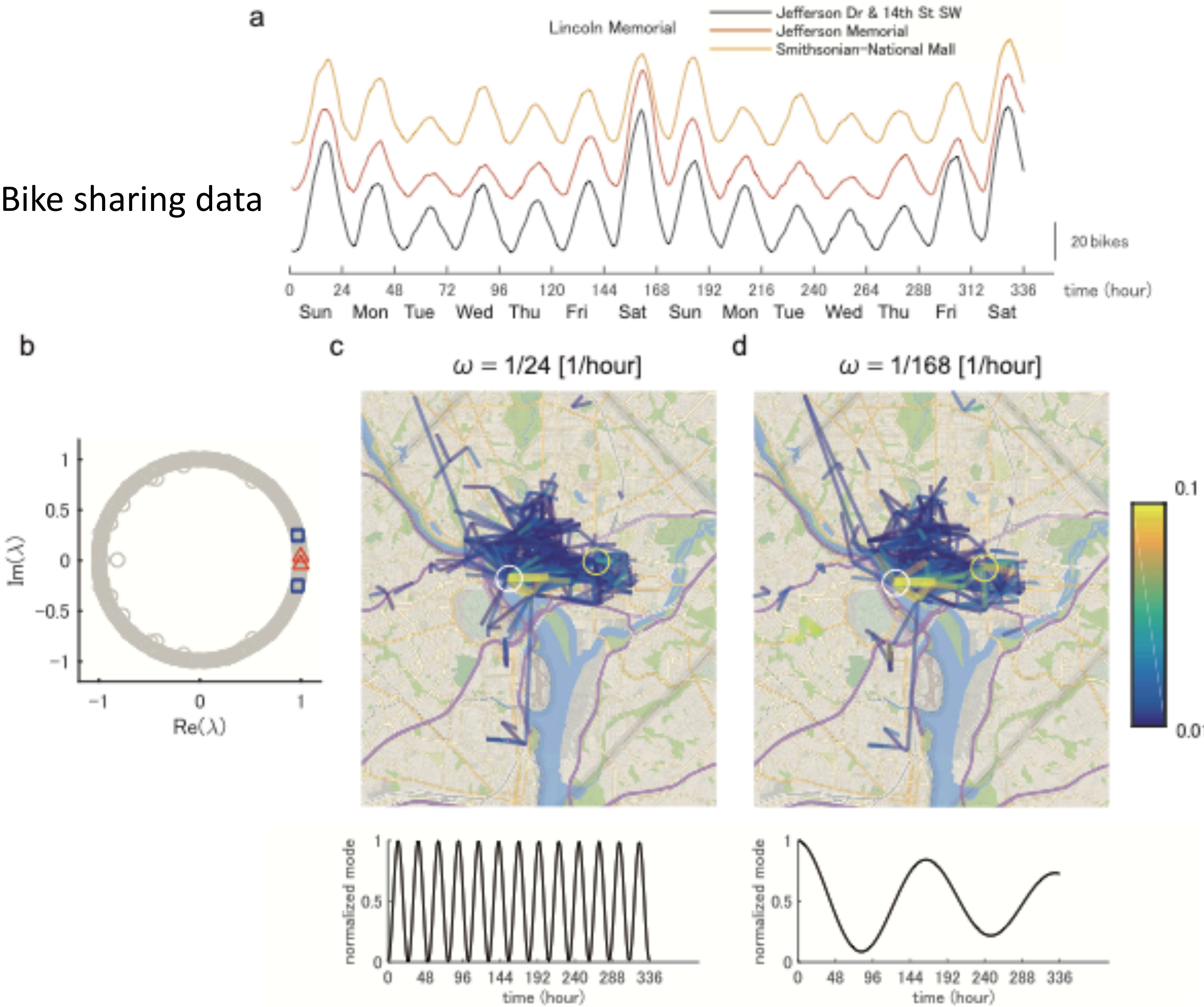


Clustering results with DMD modes as features *(with kernels defined on DMD modes* (Fujii+ 2017)*)*



(Fujii & Kawahara, Neural Networks (2019))

ex). Bike sharing data



(Fujii & Kawahara, Neural Networks (2019))

Properties and Extensibility

- Applicable to wide range of dynamical systems without preparing observables (just need to choose (*suitable*) kernel functions).
- A DMD procedure (for the naïve case) is reduced to the equivalent one of Extended DMD (Williams+ 16) (for SVD-based implementation) (Kawahara, 16).

  => but, PF operators from kernels are not necessarily bounded.

- Deliver useful *extensibility* such as
  - Random systems with kernel-mean embeddings (Hashimoto et al., *under review*)
  - Structured observables (eg. Graph sequence) (Fujii & Kawahara, *Neural Networks* (in press))
  - Metric with PF operators in RKHSs (Ishikawa et al., *NeurIPS'18*)
  - and others …

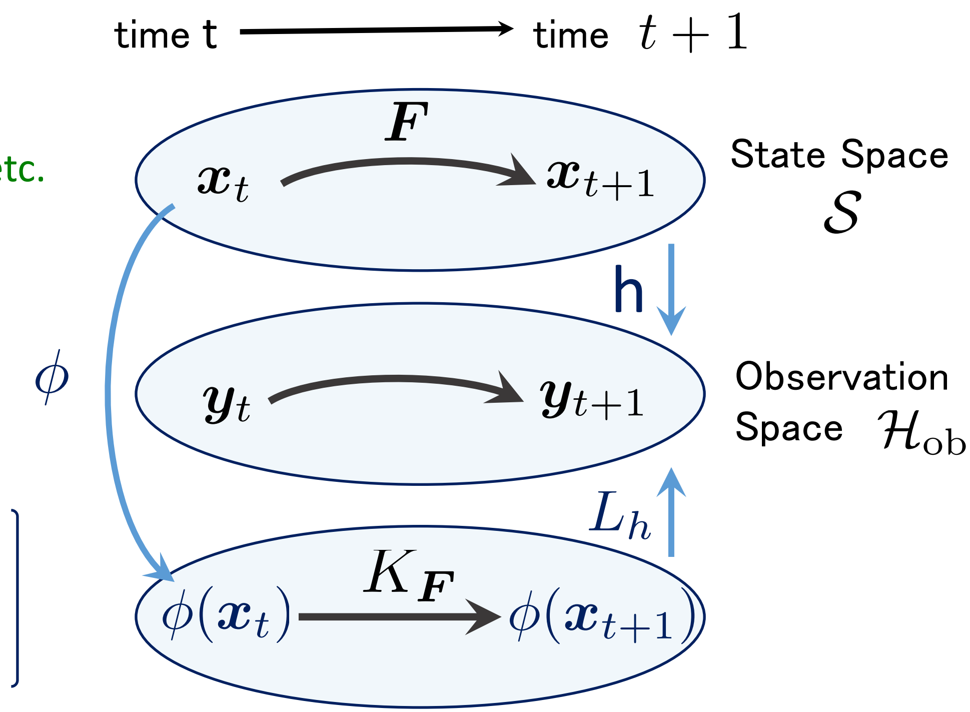
Metric on Dynamical Systems (1/2)

Metric on nonlinear dynamics with PF operator in RKHSs (Ishikawa+ 18)

➡ Generalizes (Martin 00), (Vishwanatan+ 07) etc.

Compare the properties with respect to a pair of dynamical systems:

$$D_1(\boldsymbol{F}_1, K_{\boldsymbol{F}_1}, \mathscr{I}_1) \rightleftarrows D_2(\boldsymbol{F}_2, K_{\boldsymbol{F}_2}, \mathscr{I}_2)$$



Some definitions:

$K_{\boldsymbol{F}} : \mathcal{H}_k \to \mathcal{K}_k$: Perron-Frobenius Ope.

$L_h : \mathcal{H}_k \to \mathcal{H}_{\text{ob}}$: Observation Ope.

$$\mathfrak{K}_m^T(D_1, D_2) := \operatorname{tr}\left(\bigwedge^m \sum_{r=0}^{T-1} \left(L_{h_2} K_{\boldsymbol{F}_2}^r \mathscr{I}_2\right)^* L_{h_1} K_{\boldsymbol{F}_1}^r \mathscr{I}_1\right) \in \mathbb{C}$$

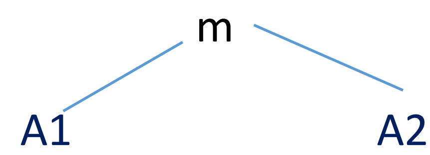
$$A_m^T(D_1, D_2) := \lim_{\epsilon \to +0} \frac{\left|\epsilon + \mathfrak{K}_m^T(D_1, D_2)\right|^2}{\left(\epsilon + \mathfrak{K}_m^T(D_1, D_1)\right)\left(\epsilon + \mathfrak{K}_m^T(D_2, D_2)\right)} \in [0, 1]$$

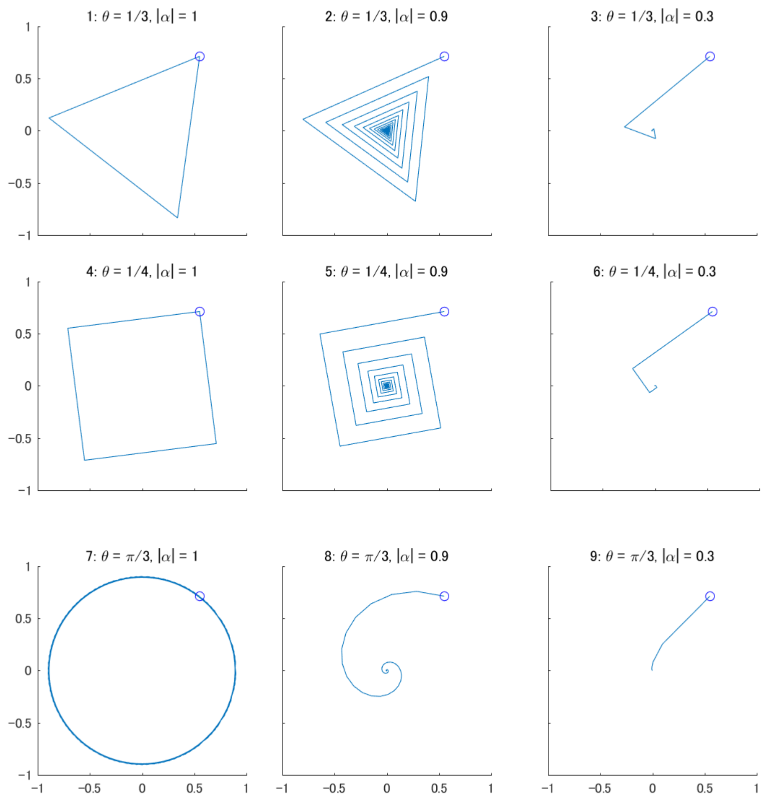
Both are positive definite kernels

=> Relation to the metrics in (Mezic 04) (Mezic+ 16) ?

Metric on Dynamical Systems (2/2)

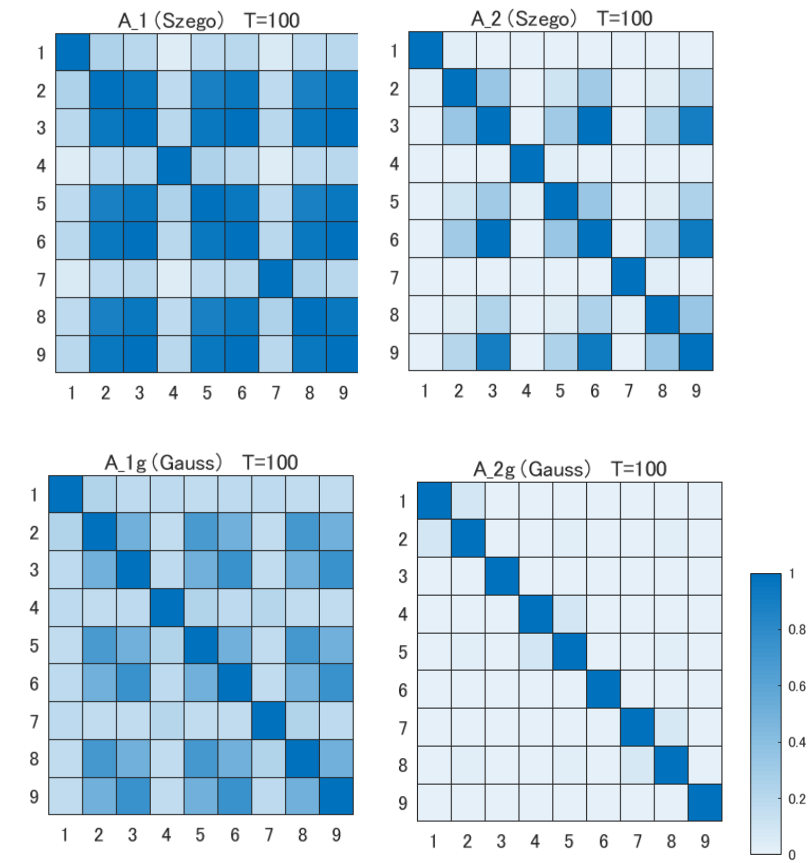
Empirical Example: Rotation in a unit circle





Szego Kernel

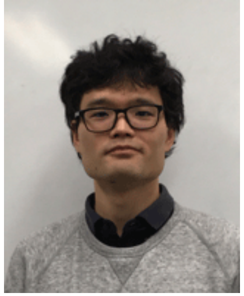
Gaussian Kernel



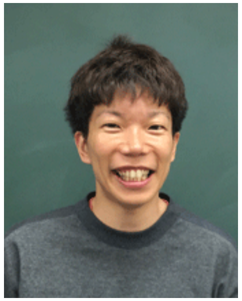
Summary

- Introduce analysis with transfer operators of dynamical systems using reproducing kernels, and describe some related recent works such as
  - Extension to random systems with kernel-mean embeddings
  - Extension to DMD for relation dynamics with vector-valued RKHSs
  - Metric on nonlinear dynamical systems with PF operators in RKHSs

Acknoledgements: I would like to thank all of my collaborators (shown below), and acknowledge supports by IMI, Kyushu Univ. and RIKEN AIP Center.



Dr. Isao Ishikawa



Dr. Masahiro Ikeda

Dr. Keisuke Fujii

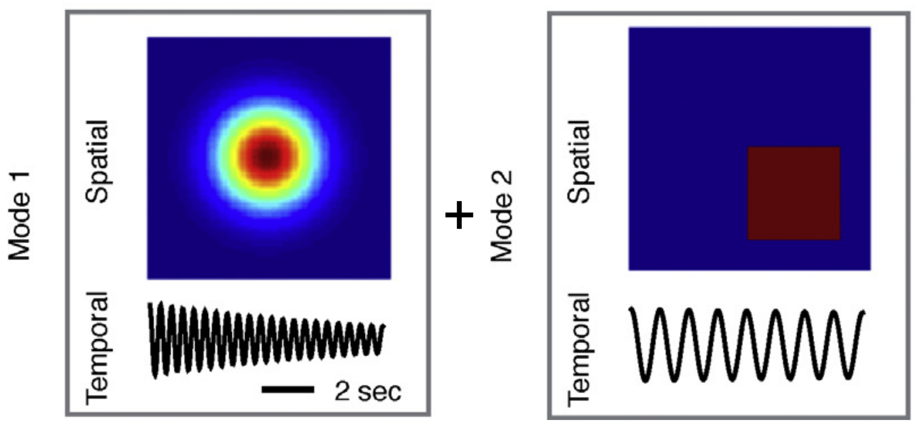
Dr. Naoya Takeishi

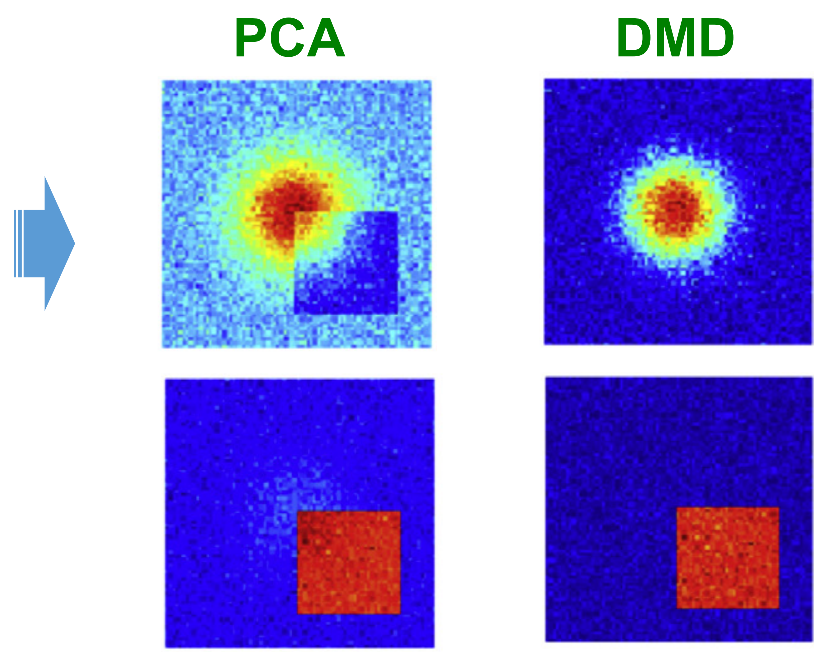
Ms. Yuka Hashimoto

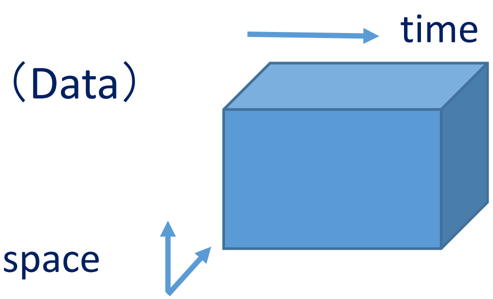
Interpretation of Dynamic Mode

DMD mode $\boldsymbol{v}_j$ gives the contribution of the corresponding dynamics to each observable:

$$\boldsymbol{g}(\boldsymbol{x}_{t+c\Delta t}) = \sum_{j=1}^{q} \lambda_j^c \varphi_j(\boldsymbol{x}_t) \boxed{\boldsymbol{v}_j}$$



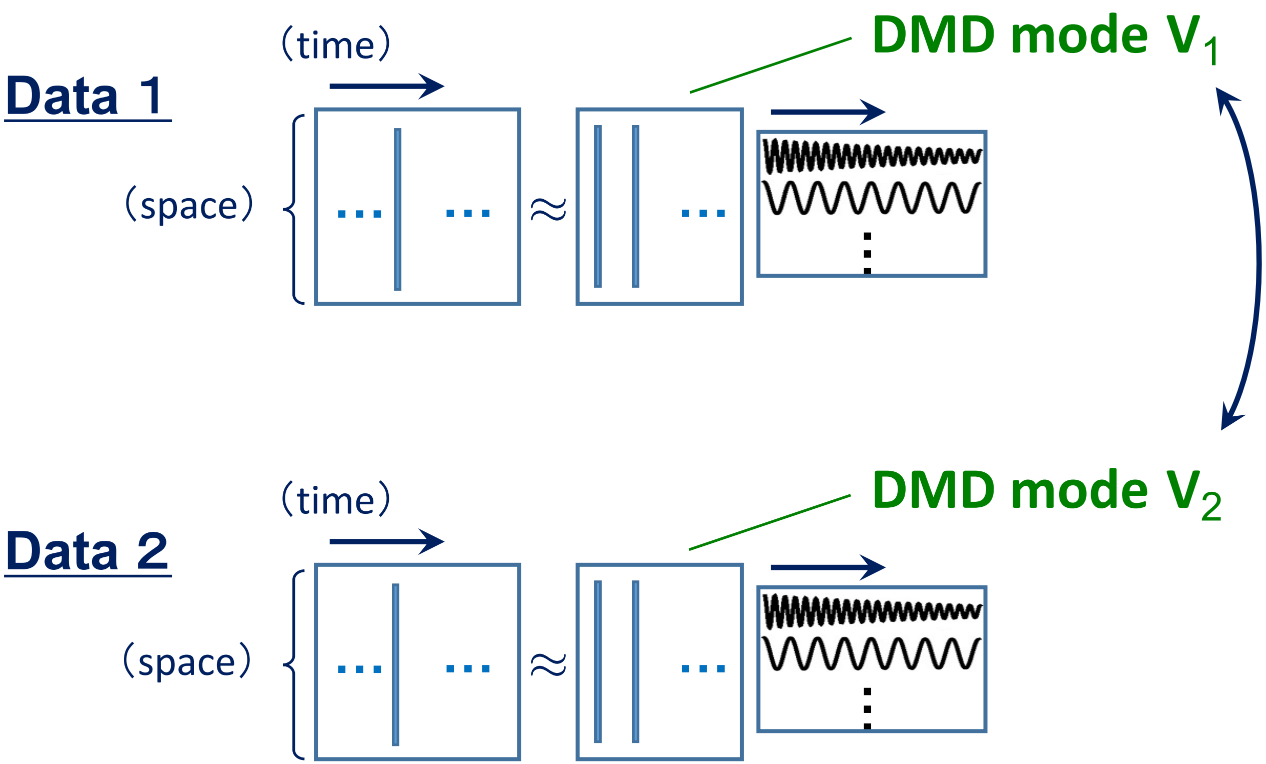




Modified from (Brunton+ 16)

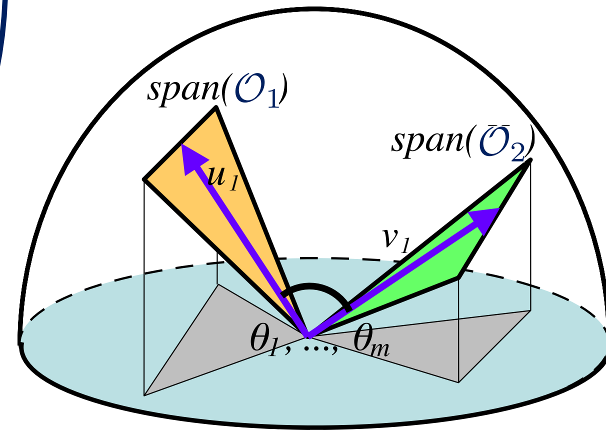
Metrics on Nonlinear Dynamics

For example, a kernel comparing spatial coherence of dynamics between two time-series data is defined with subspace angles:



$$k(\mathcal{O}_1, \mathcal{O}_2)$$

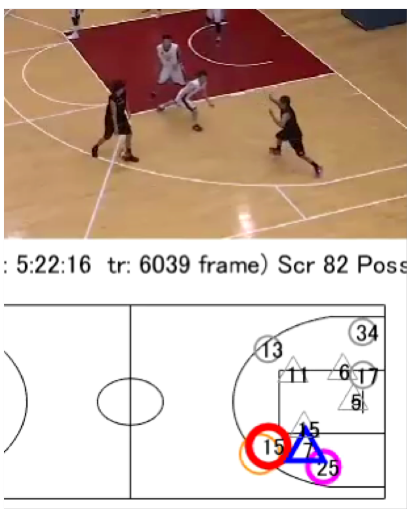
$$\left[\ \text{ex.)}\ = \prod_i (\cos\theta_i)^2\ \right]$$

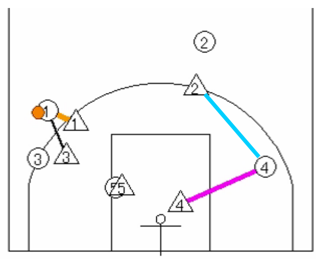


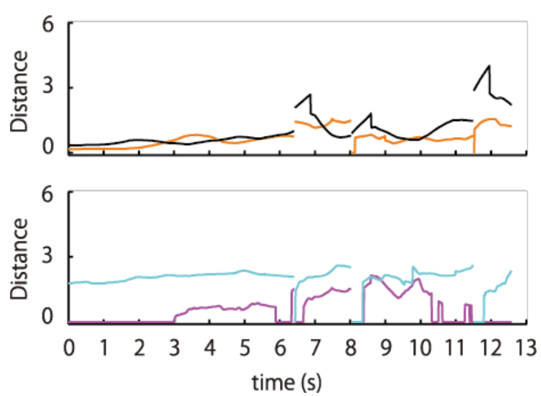
(Fujii et al., ECML-PKDD'17)
(Ishikawa et al., NIPS'18)

Embedding of Nonlinear Dynamics

Dynamics include various scales, like indivisuals => groups => team

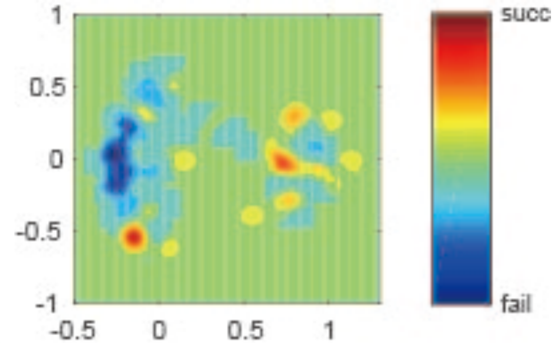




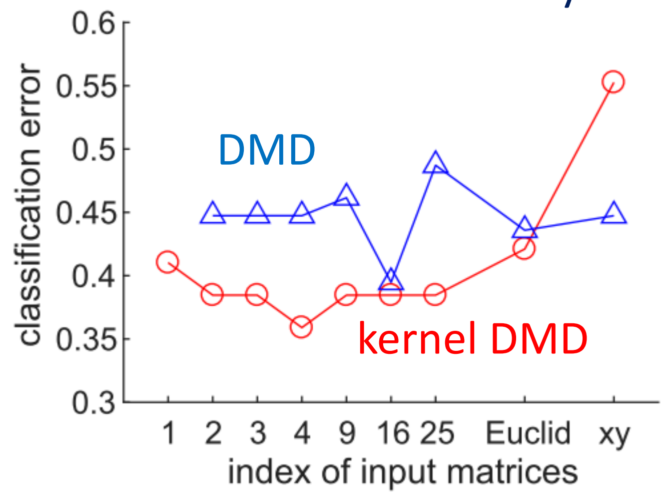


Applying DMD＋Dynamics comparison

Embedding with kDMD modes＋principal angles



Classification errors by kNN



K. Fujii, Y. Inaba and Y. Kawahara, "Koopman spectral kernels for comparing complex dynamics: Application to multi-agent sport plays," in *Proc. of ECML-PKDD'17*, pp.127-139, 2017.