

# Reduced Order Modeling of Convection-Dominated Flows, Dimensionality Reduction and Stabilization

**Rambod Mojgani**

Department of Aerospace Engineering  
University of Illinois at Urbana-Champaign  
Dr. Maciej Balajewicz, Director of Research

Department of Mechanical Engineering — Rice University  
The Environmental Fluid Dynamics Group  
Prof. Pedram Hassanzadeh

Slides on: <https://github.com/rmojgani>

October 28, 2020  
MECH Seminar Series, Rice University

# Outline

- 1 Introduction
  - Fundamentals
  - Motivation
- 2 Identifying an optimal manifold
  - Physics-aware registration-based manifold
- 3 ROMs on the identified manifold
  - Projection based ROMs
  - Neural Network based ROMs
- 4 Stabilization of time-varying ROMs
  - A feedback controller approach
- 5 Summary

# Outline

## 1 Introduction

- Fundamentals
- Motivation

## 2 Identifying an optimal manifold

- Physics-aware registration-based manifold

## 3 ROMs on the identified manifold

- Projection based ROMs
- Neural Network based ROMs

## 4 Stabilization of time-varying ROMs

- A feedback controller approach

## 5 Summary

## Numerical simulation of physical systems



Figure 1: From high fidelity simulations to reduced order models

- Low cost computational models are especially required in
  - In situ real-time calculations
  - Many-query applications
    - Optimization
    - Uncertainty quantification (UQ)

[1] J. Lee and T. Zaki, Gallery of Fluid Motion, APS, 2017 [2] D-G Caprare et. al., Gallery of Fluid Motion, APS, 2017 [3] M. Balajewicz, 2017.

# Projection based reduced order models

→ Underlying mathematical premises:

1. **Compression:** solution of governing PDEs lies on a manifold of significantly lower dimension
2. **Off-line training:**
  - the manifold can be identified/learned off-line via training simulations
  - the high-fidelity model can be reformulated with respect to this manifold, i.e. training of reduced order models (ROMs)
3. **On-line prediction:** the identified ROMs are, in principle, capable of delivering new solutions with fraction of the computational cost

# Projection based reduced order models

→ Underlying mathematical premises:

1. **Compression:** solution of governing PDEs lies on a manifold of significantly lower dimension
2. **Off-line training:**
  - the manifold can be identified/learned off-line via training simulations
  - the high-fidelity model can be reformulated with respect to this manifold, i.e. training of reduced order models (ROMs)
3. **On-line prediction:** the identified ROMs are, in principle, capable of delivering new solutions with fraction of the computational cost

## Projection based reduced order models

→ Underlying mathematical premises:

1. **Compression:** solution of governing PDEs lies on a manifold of significantly lower dimension
2. **Off-line training:**
  - the manifold can be identified/learned off-line via training simulations
  - the high-fidelity model can be reformulated with respect to this manifold, i.e. training of reduced order models (ROMs)
3. **On-line prediction:** the identified ROMs are, in principle, capable of delivering new solutions with fraction of the computational cost

## Projection based reduced order models

→ Underlying mathematical premises:

1. **Compression:** solution of governing PDEs lies on a manifold of significantly lower dimension
2. **Off-line training:**
  - the manifold can be identified/learned off-line via training simulations
  - the high-fidelity model can be reformulated with respect to this manifold, i.e. training of reduced order models (ROMs)
3. **On-line prediction:** the identified ROMs are, in principle, capable of delivering new solutions with fraction of the computational cost

## Step 1: identifying the POD subspace

- Given a snapshot matrix,  $\mathbf{M} \in \mathbb{R}^{N \times K}$

$$\mathbf{M} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_3 \quad \cdots \quad \mathbf{w}_K]; \quad \mathbf{w}_i \in \mathbb{R}^N \quad (1)$$

- The low-rank approximation problem is

$$\underset{\text{rank}(\widetilde{\mathbf{M}})=k}{\text{minimize}} \quad \|\mathbf{M} - \widetilde{\mathbf{M}}\|_F \quad (2)$$

- The rank constraint is satisfied

$$\underset{\mathbf{U} \in \mathbb{R}^{N \times k}, \mathbf{V} \in \mathbb{R}^{k \times K}}{\text{minimize}} \quad \|\mathbf{M} - \mathbf{UV}\|_F \quad (3)$$

- The solution is given by SVD/POD of the snapshot matrix

## Step 1: identifying the POD subspace

- Given a snapshot matrix,  $\mathbf{M} \in \mathbb{R}^{N \times K}$

$$\mathbf{M} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_3 \quad \cdots \quad \mathbf{w}_K]; \quad \mathbf{w}_i \in \mathbb{R}^N \quad (1)$$

- The low-rank approximation problem is

$$\underset{\text{rank}(\widetilde{\mathbf{M}})=k}{\text{minimize}} \quad \|\mathbf{M} - \widetilde{\mathbf{M}}\|_F \quad (2)$$

- The rank constraint is satisfied

$$\underset{\mathbf{U} \in \mathbb{R}^{N \times k}, \mathbf{V} \in \mathbb{R}^{k \times K}}{\text{minimize}} \quad \|\mathbf{M} - \mathbf{UV}\|_F \quad (3)$$

- The solution is given by SVD/POD of the snapshot matrix

## Step 1: identifying the POD subspace

- Given a snapshot matrix,  $\mathbf{M} \in \mathbb{R}^{N \times K}$

$$\mathbf{M} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_3 \quad \cdots \quad \mathbf{w}_K]; \quad \mathbf{w}_i \in \mathbb{R}^N \quad (1)$$

- The low-rank approximation problem is

$$\underset{\text{rank}(\widetilde{\mathbf{M}})=k}{\text{minimize}} \quad \|\mathbf{M} - \widetilde{\mathbf{M}}\|_F \quad (2)$$

- The rank constraint is satisfied

$$\underset{\mathbf{U} \in \mathbb{R}^{N \times k}, \mathbf{V} \in \mathbb{R}^{k \times K}}{\text{minimize}} \quad \|\mathbf{M} - \mathbf{U}\mathbf{V}\|_F \quad (3)$$

- The solution is given by SVD/POD of the snapshot matrix

## Step 1: identifying the POD subspace

- Given a snapshot matrix,  $\mathbf{M} \in \mathbb{R}^{N \times K}$

$$\mathbf{M} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_3 \quad \cdots \quad \mathbf{w}_K]; \quad \mathbf{w}_i \in \mathbb{R}^N \quad (1)$$

- The low-rank approximation problem is

$$\underset{\text{rank}(\widetilde{\mathbf{M}})=k}{\text{minimize}} \quad \|\mathbf{M} - \widetilde{\mathbf{M}}\|_F \quad (2)$$

- The rank constraint is satisfied

$$\underset{\mathbf{U} \in \mathbb{R}^{N \times k}, \mathbf{V} \in \mathbb{R}^{k \times K}}{\text{minimize}} \quad \|\mathbf{M} - \mathbf{U}\mathbf{V}\|_F \quad (3)$$

- The solution is given by SVD/POD of the snapshot matrix

## Step 2: constructing the reduced order model

- Consider the discretized system

$$\mathbf{R}(\mathbf{w}[n]) = \mathbf{0} \quad (4)$$

- The state,  $\mathbf{w}$ , is approximated on a global trial subspace,

$$\mathbf{w}[n] \approx \tilde{\mathbf{w}}[n] = \mathbf{U}\mathbf{w}_r[n], \quad \mathbf{U} \in \mathbb{R}^{N \times k}, \quad \mathbf{w}_r[n] \in \mathbb{R}^k, \quad k \ll N \quad (5)$$

- Galerkin projection onto the test subspace  $\mathbf{U} \in \mathbb{R}^{N \times k}$ ,

$$\mathbf{U}^T \mathbf{R}(\mathbf{U}\mathbf{w}_r[n]) = \mathbf{0} \quad (6)$$

## Step 2: constructing the reduced order model

- Consider the discretized system

$$\mathbf{R}(\mathbf{w}[n]) = \mathbf{0} \quad (4)$$

- The state,  $\mathbf{w}$ , is approximated on a global trial subspace,

$$\mathbf{w}[n] \approx \tilde{\mathbf{w}}[n] = \mathbf{U}\mathbf{w}_r[n], \quad \mathbf{U} \in \mathbb{R}^{N \times k}, \quad \mathbf{w}_r[n] \in \mathbb{R}^k, \quad k \ll N \quad (5)$$

- Galerkin projection onto the test subspace  $\mathbf{U} \in \mathbb{R}^{N \times k}$ ,

$$\mathbf{U}^T \mathbf{R}(\mathbf{U}\mathbf{w}_r[n]) = \mathbf{0} \quad (6)$$

## Step 2: constructing the reduced order model

- Consider the discretized system

$$\mathbf{R}(\mathbf{w}[n]) = \mathbf{0} \quad (4)$$

- The state,  $\mathbf{w}$ , is approximated on a global trial subspace,

$$\mathbf{w}[n] \approx \tilde{\mathbf{w}}[n] = \mathbf{U}\mathbf{w}_r[n], \quad \mathbf{U} \in \mathbb{R}^{N \times k}, \quad \mathbf{w}_r[n] \in \mathbb{R}^k, \quad k \ll N \quad (5)$$

- Galerkin projection onto the test subspace  $\mathbf{U} \in \mathbb{R}^{N \times k}$ ,

$$\mathbf{U}^T \mathbf{R}(\mathbf{U}\mathbf{w}_r[n]) = \mathbf{0} \quad (6)$$

## Motivational problem 1: one-dimensional convection

- Consider the scalar linear convection equation

$$\begin{cases} \frac{\partial w}{\partial t} + \frac{\partial w}{\partial x} = 0, & (x, t) \in [0, \infty) \times [0, 1] \\ w(x, 0) = 0.8 + e^{-(x-0.3)^2/0.05^2} \\ w(0, t) = 0.8 \end{cases} \quad (7)$$

- Exact solution is  $w(x, t) = w(x - t, 0)$

## Motivational problem 1: one-dimensional convection

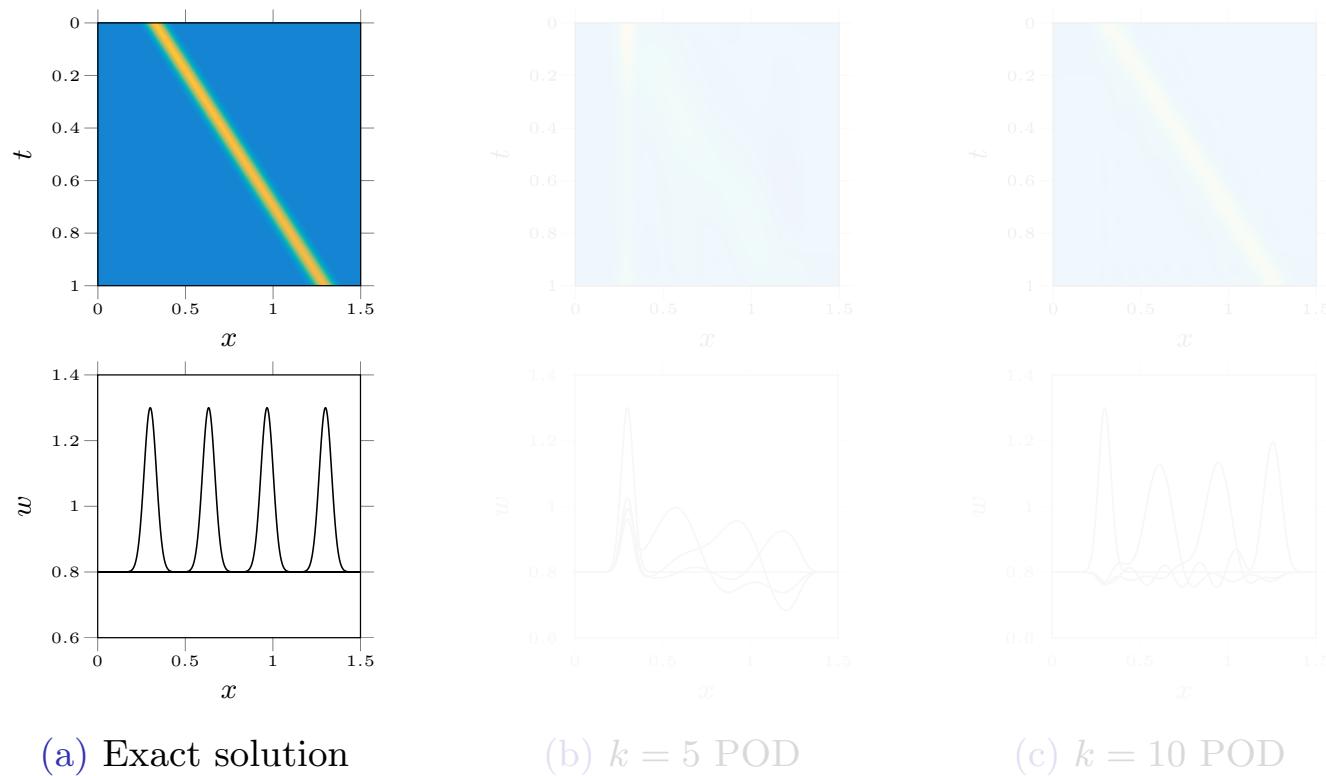


Figure 2: POD compression of the scalar convection equation

## Motivational problem 1: one-dimensional convection

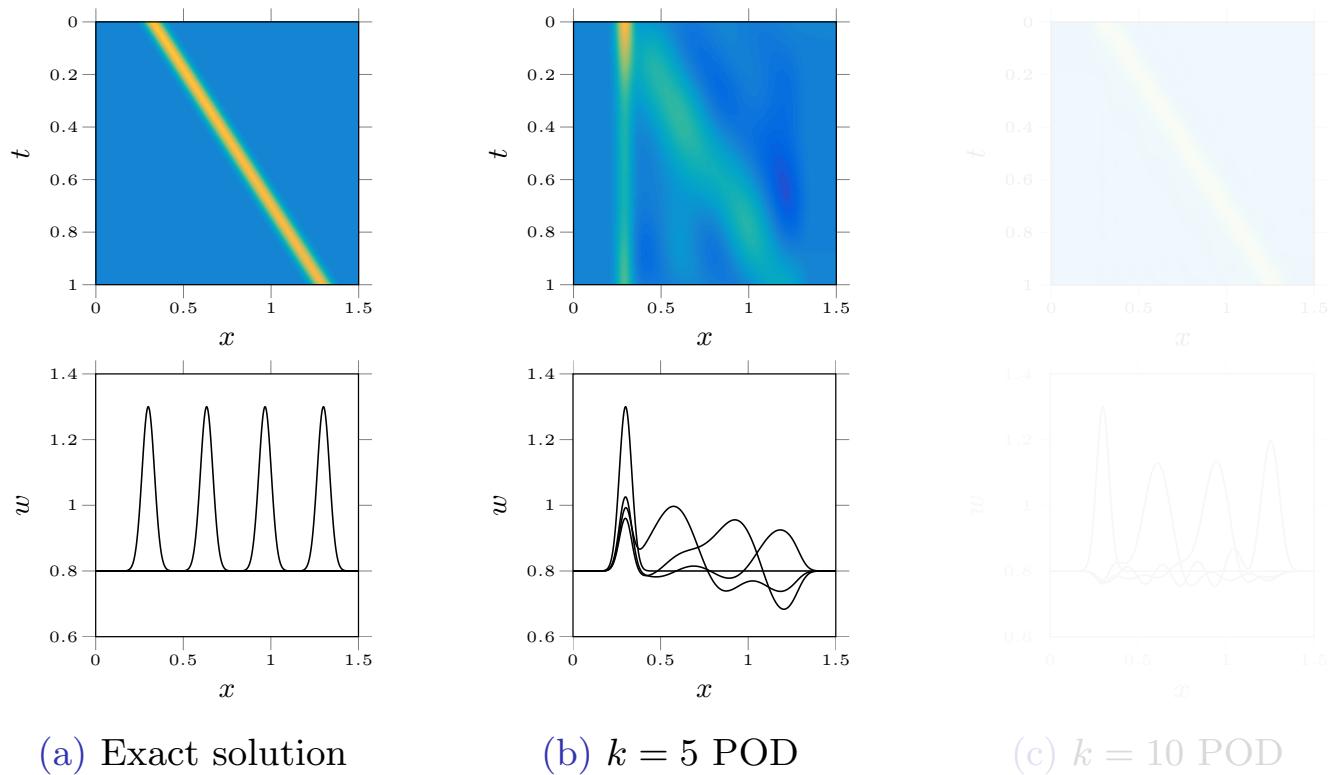


Figure 2: POD compression of the scalar convection equation

## Motivational problem 1: one-dimensional convection

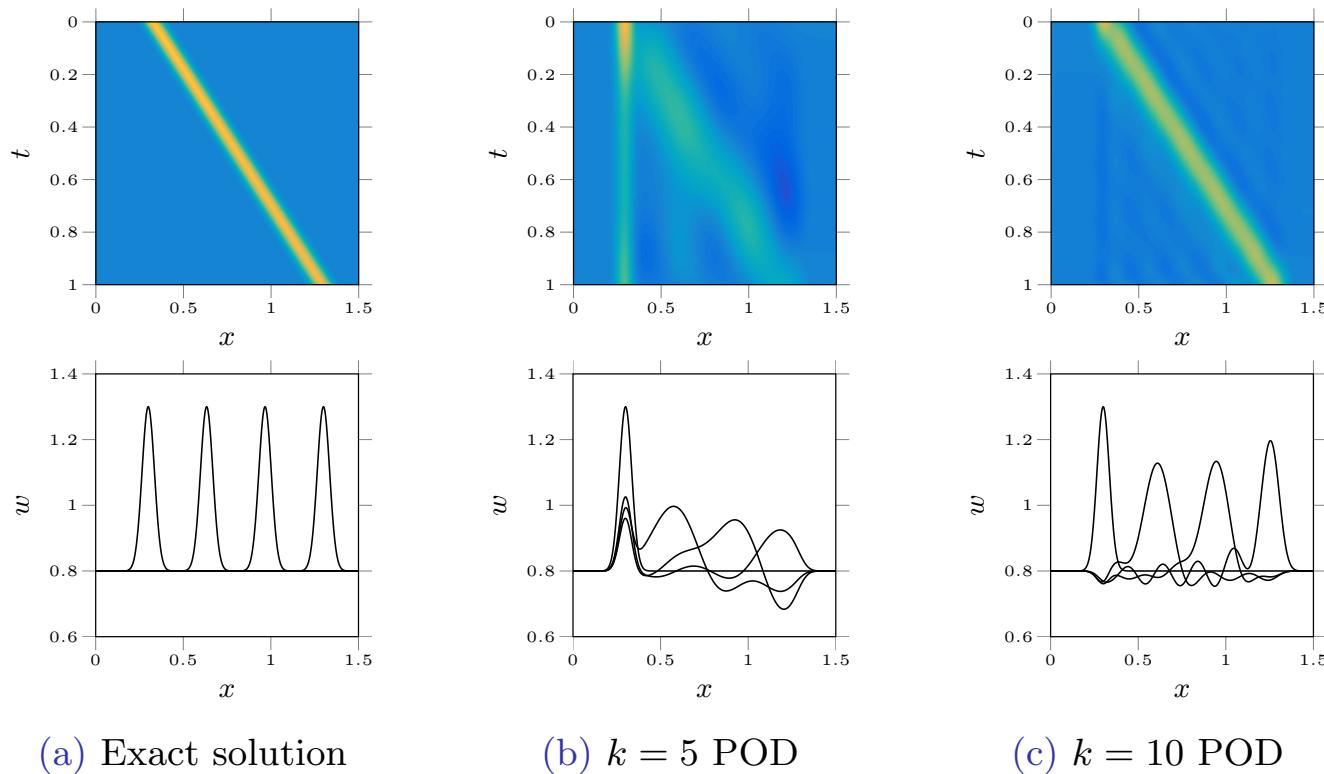


Figure 2: POD compression of the scalar convection equation

## Motivational problem 2: Rotating character “A”

- Each snapshot is a  $50 \times 50$  matrix of real numbers corresponding to greyscale values of the character “A”

A A A A A A A A A

(a) The snapshots



(b) The rank-1 reconstruction ( $k = 1$ ) on the POD subspace

Figure 3: POD compression of 90 degrees rotation of character “A”

## Motivational problem 2: Rotating character “A”

- Each snapshot is a  $50 \times 50$  matrix of real numbers corresponding to greyscale values of the character “A”

A A A A A A A A A

(a) The snapshots



(b) The rank-1 reconstruction ( $k = 1$ ) on the POD subspace

Figure 3: POD compression of 90 degrees rotation of character “A”

## Motivation: observations

- Despite relative simplicity of the snapshots, these particular two examples could not be approximated using a *small* number of bases using the traditional *linear* approach
- It is simply not possible to efficiently compress certain wave-like solutions using a low-rank decomposition
- A *fundamentally* different approach is required

## Motivation: observations

- Despite relative simplicity of the snapshots, these particular two examples could not be approximated using a *small* number of bases using the traditional *linear* approach
- It is simply not possible to efficiently compress certain wave-like solutions using a low-rank decomposition
- A *fundamentally* different approach is required

## Motivation: observations

- Despite relative simplicity of the snapshots, these particular two examples could not be approximated using a *small* number of bases using the traditional *linear* approach
- It is simply not possible to efficiently compress certain wave-like solutions using a low-rank decomposition
- A *fundamentally* different approach is required

## Previous attempts

1. Seeking symmetries, self-similarities, and coordinate transformations or modifying the decomposition to include the mentioned properties, i.e.  $\mathcal{L}$ -DMD, sPOD  
[Rowley and Marsden, 2000, Rowley et al., 2003, Kavousanakis et al., 2007, Rapun and Vega, 2010, Gerbeau and Lombardi, 2014, Rim et al., 2017, Mowlavi and Sapsis, 2018, Rim and Mandli, 2018, Reiss et al., 2015, Reiss et al., 2018, Schulze et al., 2018, Mendible et al., 2019, Taddei, 2020]
2. Using local bases or splitting the domain or bases  
[Amsallem et al., 2012, Lucia, 2001, Carlberg, 2015]
3. Adding temporal dependency in the reduced bases  
[Iollo and Lombardi, 2014, Gerbeau and Lombardi, 2014, Babaee et al., 2017].

## Previous attempts

1. Seeking symmetries, self-similarities, and coordinate transformations or modifying the decomposition to include the mentioned properties, i.e.  $\mathcal{L}$ -DMD, sPOD  
[Rowley and Marsden, 2000, Rowley et al., 2003, Kavousanakis et al., 2007, Rapun and Vega, 2010, Gerbeau and Lombardi, 2014, Rim et al., 2017, Mowlavi and Sapsis, 2018, Rim and Mandli, 2018, Reiss et al., 2015, Reiss et al., 2018, Schulze et al., 2018, Mendible et al., 2019, Taddei, 2020]
2. Using local bases or splitting the domain or bases  
[Amsallem et al., 2012, Lucia, 2001, Carlberg, 2015]
3. Adding temporal dependency in the reduced bases  
[Iollo and Lombardi, 2014, Gerbeau and Lombardi, 2014, Babaee et al., 2017].

## Previous attempts

1. Seeking symmetries, self-similarities, and coordinate transformations or modifying the decomposition to include the mentioned properties, i.e.  $\mathcal{L}$ -DMD, sPOD  
[Rowley and Marsden, 2000, Rowley et al., 2003, Kavousanakis et al., 2007, Rapun and Vega, 2010, Gerbeau and Lombardi, 2014, Rim et al., 2017, Mowlavi and Sapsis, 2018, Rim and Mandli, 2018, Reiss et al., 2015, Reiss et al., 2018, Schulze et al., 2018, Mendible et al., 2019, Taddei, 2020]
2. Using local bases or splitting the domain or bases  
[Amsallem et al., 2012, Lucia, 2001, Carlberg, 2015]
3. Adding temporal dependency in the reduced bases  
[Iollo and Lombardi, 2014, Gerbeau and Lombardi, 2014, Babaee et al., 2017].

# Identifying a subspace/manifold

- We propose to construct the ROMs on *nonlinear* manifolds
- Linear subspaces
  - Proper Orthogonal Decomposition (POD)
  - Dynamic Mode Decomposition (DMD)
  - ...
- Nonlinear manifolds
  - Iso-map [Tenenbaum, 1998]
  - t-distributed stochastic neighbor embedding (t-SNE) [Maaten and Hinton, 2008]
  - Auto-encoders [G.E and R.R, 2006]
  - ...
- The goals of the proposed approach
  - Data-driven
  - Compressibility on the identified manifold
  - A two-way map between the physical space and latent space
  - Interpretable low-dimensional/latent variables

# Identifying a subspace/manifold

- We propose to construct the ROMs on *nonlinear* manifolds
- Linear subspaces
  - Proper Orthogonal Decomposition (POD)
  - Dynamic Mode Decomposition (DMD)
  - ...
- Nonlinear manifolds
  - Iso-map [Tenenbaum, 1998]
  - t-distributed stochastic neighbor embedding (t-SNE) [Maaten and Hinton, 2008]
  - Auto-encoders [G.E and R.R, 2006]
  - ...
- The goals of the proposed approach
  - Data-driven
  - Compressibility on the identified manifold
  - A two-way map between the physical space and latent space
  - Interpretable low-dimensional/latent variables

# Identifying a subspace/manifold

- We propose to construct the ROMs on *nonlinear* manifolds
- Linear subspaces
  - Proper Orthogonal Decomposition (POD)
  - Dynamic Mode Decomposition (DMD)
  - ...
- Nonlinear manifolds
  - Iso-map [Tenenbaum, 1998]
  - t-distributed stochastic neighbor embedding (t-SNE) [Maaten and Hinton, 2008]
  - Auto-encoders [G.E and R.R, 2006]
  - ...
- The goals of the proposed approach
  - Data-driven
  - Compressibility on the identified manifold
  - A two-way map between the physical space and latent space
  - Interpretable low-dimensional/latent variables

## Identifying a subspace/manifold

- We propose to construct the ROMs on *nonlinear* manifolds
- Linear subspaces
  - Proper Orthogonal Decomposition (POD)
  - Dynamic Mode Decomposition (DMD)
  - ...
- Nonlinear manifolds
  - Iso-map [Tenenbaum, 1998]
  - t-distributed stochastic neighbor embedding (t-SNE) [Maaten and Hinton, 2008]
  - Auto-encoders [G.E and R.R, 2006]
  - ...
- The goals of the proposed approach
  - Data-driven
  - Compressibility on the identified manifold
  - A two-way map between the physical space and latent space
  - Interpretable low-dimensional/latent variables

# Outline

## 1 Introduction

- Fundamentals
- Motivation

## 2 Identifying an optimal manifold

- Physics-aware registration-based manifold

## 3 ROMs on the identified manifold

- Projection based ROMs
- Neural Network based ROMs

## 4 Stabilization of time-varying ROMs

- A feedback controller approach

## 5 Summary

## The proposed manifold

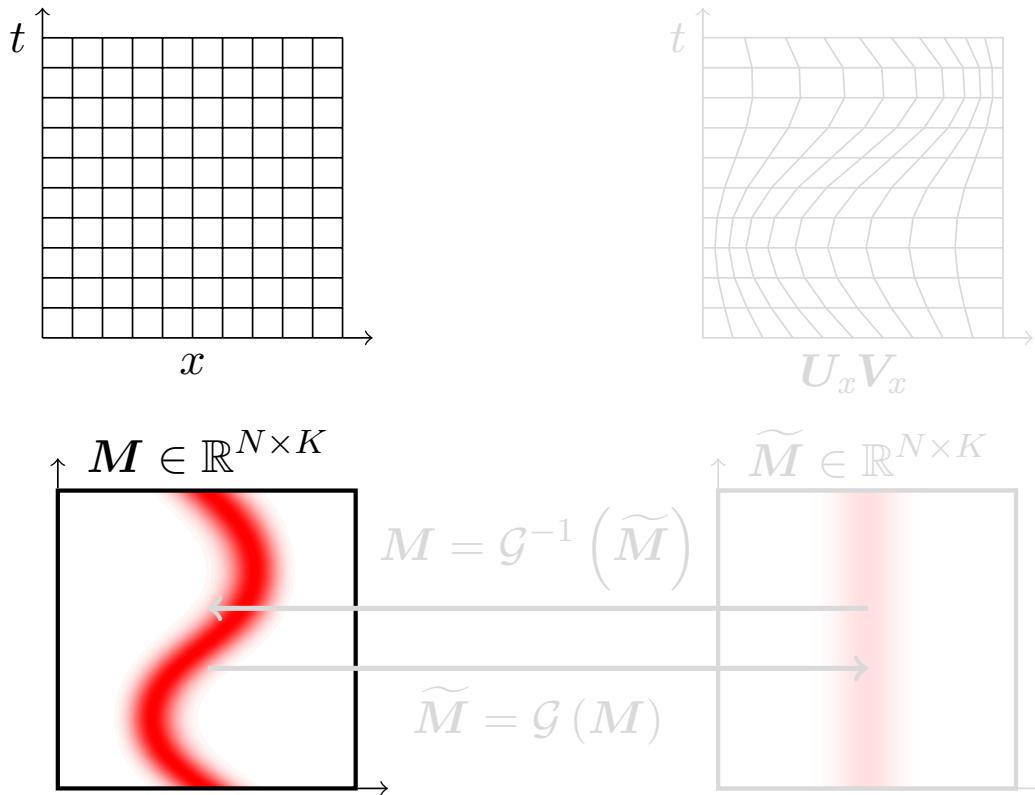


Figure 4: Illustration of the proposed approach

## The proposed manifold

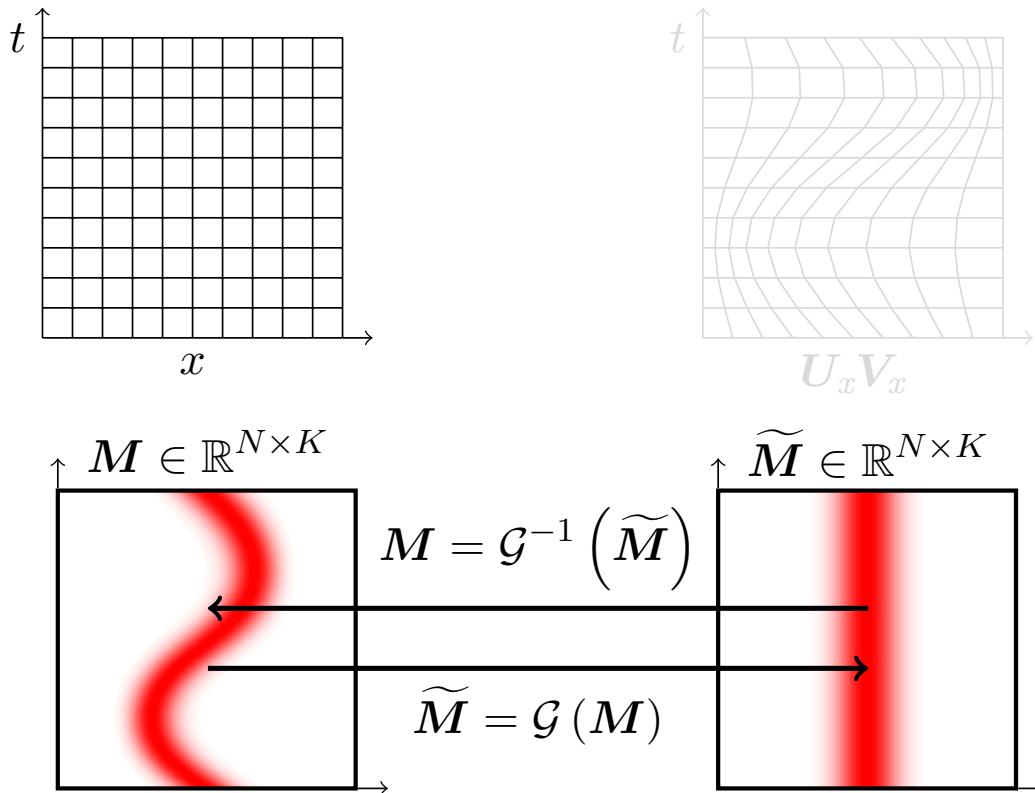


Figure 4: Illustration of the proposed approach

## The proposed manifold

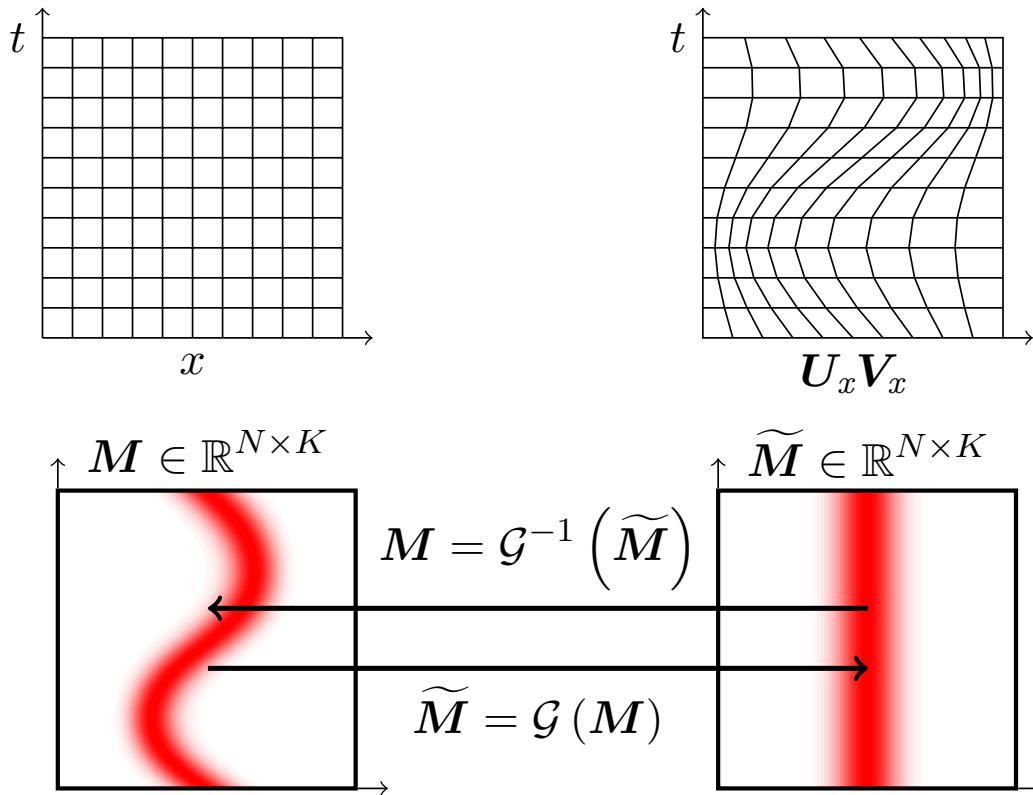


Figure 4: Illustration of the proposed approach

## The proposed manifold

- For a given snapshot matrix  $\mathbf{M} \in \mathbb{R}^{N \times K}$ , find an empirical *map* that solves the minimization problem:

$$\underset{\mathcal{G}}{\text{minimize}} \left\| \mathbf{M} - \mathcal{G}^{-1} \left( \widetilde{\mathbf{M}} \right) \right\|_F \quad (8)$$

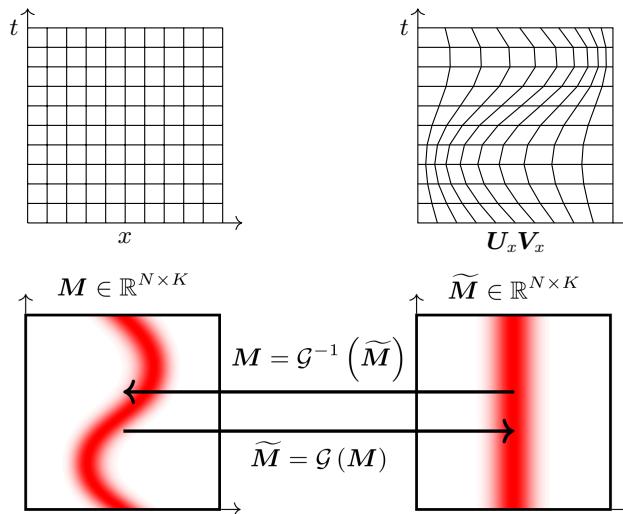


Figure 4: Illustration of the proposed approach

## The proposed manifold

- For a given snapshot matrix  $\mathbf{M} \in \mathbb{R}^{N \times K}$ , find an empirical map that solves *low-rank approximation* problem:

$$\underset{\mathcal{G}}{\text{minimize}} \|\mathbf{M} - \mathcal{G}^{-1}(\mathbf{U}\mathbf{V})\|_F \quad (9)$$

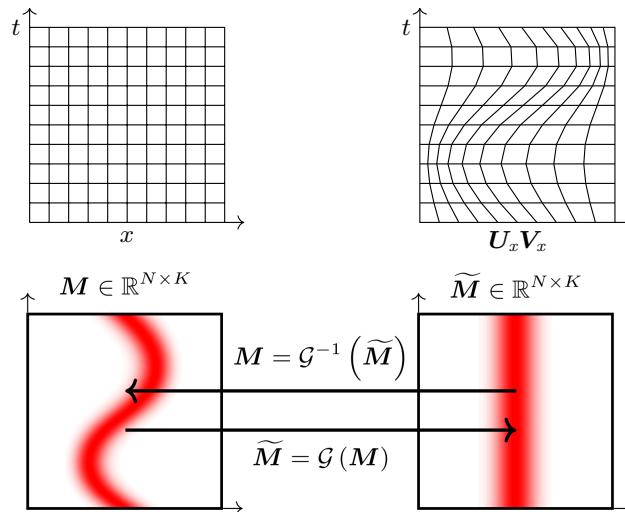


Figure 4: Illustration of the proposed approach

## The proposed manifold

- For a given snapshot matrix  $\mathbf{M} \in \mathbb{R}^{N \times K}$ , find a *low-rank grid* that solves the minimization problem:

$$\underset{\mathbf{U}_x \in \mathbb{R}^{N \times r}, \mathbf{V}_x \in \mathbb{R}^{r \times K}}{\text{minimize}} \left\| \mathbf{M} - \mathcal{G}^{-1} (\mathbf{U} \mathbf{V}) \right\|_F \quad (10)$$

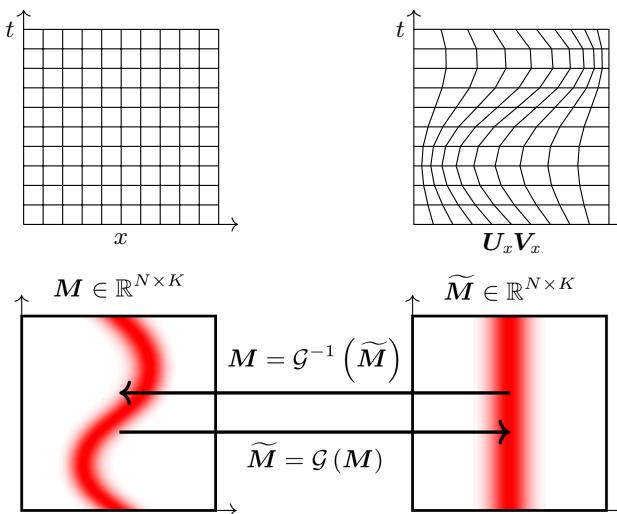


Figure 4: Illustration of the proposed approach

## Interpolation and diffeomorphism

- Impose diffeomorphism: “a map,  $\mathcal{G}$ , is said to be diffeomorphic if  $\mathcal{G}$  and  $\mathcal{G}^{-1}$  are differentiable”
- Guarantees existence and uniqueness of  $\widetilde{\mathbf{M}}$ , given a  $\mathbf{M}$  and vice versa
- For the map to be diffeomorphic
  - Bijectivity (i.e. one to oneness)
  - Continuity of the function and its inverse function
  - Smoothness

## Interpolation and diffeomorphism

- Impose diffeomorphism: “a map,  $\mathcal{G}$ , is said to be diffeomorphic if  $\mathcal{G}$  and  $\mathcal{G}^{-1}$  are differentiable”
- Guarantees existence and uniqueness of  $\widetilde{\mathbf{M}}$ , given a  $\mathbf{M}$  and vice versa
- For the map to be diffeomorphic
  - Bijectivity (i.e. one to oneness)
  - Smoothness

## Interpolation and diffeomorphism

- Impose diffeomorphism: “a map,  $\mathcal{G}$ , is said to be diffeomorphic if  $\mathcal{G}$  and  $\mathcal{G}^{-1}$  are differentiable”
- Guarantees existence and uniqueness of  $\widetilde{\mathbf{M}}$ , given a  $\mathbf{M}$  and vice versa
- For the map to be diffeomorphic
  - Bijectivity (i.e. one to oneness)
    - is enforced by ensuring that volume of all the cells remain strictly positive
  - Smoothness
    - is promoted by penalizing abrupt changes of the grid volume, both in space and parameter/time

## Interpolation and diffeomorphism

- Impose diffeomorphism: “a map,  $\mathcal{G}$ , is said to be diffeomorphic if  $\mathcal{G}$  and  $\mathcal{G}^{-1}$  are differentiable”
- Guarantees existence and uniqueness of  $\widetilde{\mathbf{M}}$ , given a  $\mathbf{M}$  and vice versa
- For the map to be diffeomorphic
  - Bijectivity (i.e. one to oneness)
    - is enforced by ensuring that volume of all the cells remain strictly positive
  - Smoothness
    - is promoted by penalizing abrupt changes of the grid volume, both in space and parameter/time

## Physics-aware registration based auto-encoder

- The final minimization problem:

$$\begin{aligned}
 & \underset{\mathbf{U}_x, \mathbf{V}_x}{\text{minimize}} \quad \|\mathbf{M} - \mathcal{G}^{-1}(\mathbf{U}\mathbf{V})\|_F + \|\boldsymbol{\Gamma}_1 \mathbf{U}_x\|_F + \|\mathbf{V}_x \boldsymbol{\Gamma}_2^T\|_F \\
 & \text{subject to } \mathbf{v}[n] \geq v_{\min}, \text{ for } n = 1, \dots, K \\
 & \quad \mathbf{x}[n]|_{\partial\Omega} = \mathbf{x}|_{\partial\Omega}, \text{ for } n = 1, \dots, K
 \end{aligned} \tag{11}$$

- $\mathbf{U} \in \mathbb{R}^{N \times k_r}$ ,  $\mathbf{V} \in \mathbb{R}^{k_r \times K}$
- $\mathbf{U}_x \in \mathbb{R}^{N \times r}$ ,  $\mathbf{V}_x \in \mathbb{R}^{r \times K}$
- $\boldsymbol{\Gamma}_1 \in \mathbb{R}^{N \times N}$ ,  $\boldsymbol{\Gamma}_2 \in \mathbb{R}^{K \times K}$
- $\mathbf{v}[n] \in \mathbb{R}^N$  is a vector of cell volumes of the time-varying grid
- $\mathbf{x}[n]|_{\partial\Omega}$  and  $\mathbf{x}|_{\partial\Omega}$  are boundary points of the time-varying and Eulerian grids

## Physics-aware registration based auto-encoder

- The final minimization problem:

$$\begin{aligned}
 & \underset{\mathbf{U}_x, \mathbf{V}_x}{\text{minimize}} \quad \|\mathbf{M} - \mathcal{G}^{-1}(\mathbf{U}\mathbf{V})\|_F + \|\boldsymbol{\Gamma}_1 \mathbf{U}_x\|_F + \|\mathbf{V}_x \boldsymbol{\Gamma}_2^T\|_F \\
 & \text{subject to } \mathbf{v}[n] \geq v_{\min}, \text{ for } n = 1, \dots, K \\
 & \quad \mathbf{x}[n]|_{\partial\Omega} = \mathbf{x}|_{\partial\Omega}, \text{ for } n = 1, \dots, K
 \end{aligned} \tag{11}$$

- $\mathbf{U} \in \mathbb{R}^{N \times k_r}$ ,  $\mathbf{V} \in \mathbb{R}^{k_r \times K}$
- $\mathbf{U}_x \in \mathbb{R}^{N \times r}$ ,  $\mathbf{V}_x \in \mathbb{R}^{r \times K}$
- $\boldsymbol{\Gamma}_1 \in \mathbb{R}^{N \times N}$ ,  $\boldsymbol{\Gamma}_2 \in \mathbb{R}^{K \times K}$
- $\mathbf{v}[n] \in \mathbb{R}^N$  is a vector of cell volumes of the time-varying grid
- $\mathbf{x}[n]|_{\partial\Omega}$  and  $\mathbf{x}|_{\partial\Omega}$  are boundary points of the time-varying and Eulerian grids

## Physics-aware registration based auto-encoder

- The final minimization problem:

$$\begin{aligned}
 & \underset{\mathbf{U}_x, \mathbf{V}_x}{\text{minimize}} \quad \|\mathbf{M} - \mathcal{G}^{-1}(\mathbf{U}\mathbf{V})\|_F + \|\boldsymbol{\Gamma}_1 \mathbf{U}_x\|_F + \|\mathbf{V}_x \boldsymbol{\Gamma}_2^T\|_F \\
 & \text{subject to } \mathbf{v}[n] \geq v_{\min}, \text{ for } n = 1, \dots, K \\
 & \quad \mathbf{x}[n]|_{\partial\Omega} = \mathbf{x}|_{\partial\Omega}, \text{ for } n = 1, \dots, K
 \end{aligned} \tag{11}$$

- $\mathbf{U} \in \mathbb{R}^{N \times k_r}$ ,  $\mathbf{V} \in \mathbb{R}^{k_r \times K}$
- $\mathbf{U}_x \in \mathbb{R}^{N \times r}$ ,  $\mathbf{V}_x \in \mathbb{R}^{r \times K}$
- $\boldsymbol{\Gamma}_1 \in \mathbb{R}^{N \times N}$ ,  $\boldsymbol{\Gamma}_2 \in \mathbb{R}^{K \times K}$
- $\mathbf{v}[n] \in \mathbb{R}^N$  is a vector of cell volumes of the time-varying grid
- $\mathbf{x}[n]|_{\partial\Omega}$  and  $\mathbf{x}|_{\partial\Omega}$  are boundary points of the time-varying and Eulerian grids

## Physics-aware registration based auto-encoder

- The final minimization problem:

$$\begin{aligned}
 & \underset{\mathbf{U}_x, \mathbf{V}_x}{\text{minimize}} \quad \|\mathbf{M} - \mathcal{G}^{-1}(\mathbf{U}\mathbf{V})\|_F + \|\boldsymbol{\Gamma}_1 \mathbf{U}_x\|_F + \|\mathbf{V}_x \boldsymbol{\Gamma}_2^T\|_F \\
 & \text{subject to } \mathbf{v}[n] \geq v_{\min}, \text{ for } n = 1, \dots, K \\
 & \quad \mathbf{x}[n]|_{\partial\Omega} = \mathbf{x}|_{\partial\Omega}, \text{ for } n = 1, \dots, K
 \end{aligned} \tag{11}$$

- $\mathbf{U} \in \mathbb{R}^{N \times k_r}$ ,  $\mathbf{V} \in \mathbb{R}^{k_r \times K}$
- $\mathbf{U}_x \in \mathbb{R}^{N \times r}$ ,  $\mathbf{V}_x \in \mathbb{R}^{r \times K}$
- $\boldsymbol{\Gamma}_1 \in \mathbb{R}^{N \times N}$ ,  $\boldsymbol{\Gamma}_2 \in \mathbb{R}^{K \times K}$
- $\mathbf{v}[n] \in \mathbb{R}^N$  is a vector of cell volumes of the time-varying grid
- $\mathbf{x}[n]|_{\partial\Omega}$  and  $\mathbf{x}|_{\partial\Omega}$  are boundary points of the time-varying and Eulerian grids

## Numerical Example: Rotating character “A”

- Each snapshot is a  $50 \times 50$  matrix of real numbers corresponding to greyscale values of the character A

A A A A A A A A A

(a) The snapshots



(b) The rank-1 reconstruction ( $k = 1$ ) on the POD subspace



(c) The rank-1 grid corresponding to the identified manifold

A A A A A A A A A

(d) The rank-1 reconstruction ( $k_r = 1$ ) on the rank-1 grid ( $r = 1$ )

Figure 5: 90 degrees rotation of character “A”

## Numerical Example: Rotating character “A”

- Each snapshot is a  $50 \times 50$  matrix of real numbers corresponding to greyscale values of the character A

A A A A A A A A A

(a) The snapshots



(b) The rank-1 reconstruction ( $k = 1$ ) on the POD subspace



(c) The rank-1 grid corresponding to the identified manifold

A A A A A A A A A

(d) The rank-1 reconstruction ( $k_r = 1$ ) on the rank-1 grid ( $r = 1$ )

Figure 5: 90 degrees rotation of character “A”

## Numerical Example: Rotating character “A”

- Each snapshot is a  $50 \times 50$  matrix of real numbers corresponding to greyscale values of the character A

A A A ↗ ↗ ↗ ↗ ↗ ↗ ↗ ↗ ↗

(a) The snapshots



(b) The rank-1 reconstruction ( $k = 1$ ) on the POD subspace



(c) The rank-1 grid corresponding to the identified manifold



(d) The rank-1 reconstruction ( $k_r = 1$ ) on the rank-1 grid ( $r = 1$ )

Figure 5: 90 degrees rotation of character “A”

## Numerical Example: Rotating character “A”

- Each snapshot is a  $50 \times 50$  matrix of real numbers corresponding to greyscale values of the character A

A A A A A A A A A

(a) The snapshots



(b) The rank-1 reconstruction ( $k = 1$ ) on the POD subspace



(c) The rank-1 grid corresponding to the identified manifold

A A A A A A A A A

(d) The rank-1 reconstruction ( $k_r = 1$ ) on the rank-1 grid ( $r = 1$ )

Figure 5: 90 degrees rotation of character “A”

## Numerical Example: Rotating character “A”

- Each snapshot is a  $50 \times 50$  matrix of real numbers corresponding to greyscale values of the character A

A A A A A A A A A

(a) The snapshots



(b) The rank-1 reconstruction ( $k = 1$ ) on the POD subspace



(c) The rank-2 grid corresponding to the identified manifold

A A A A A A A A A

(d) The rank-1 reconstruction ( $k_r = 1$ ) on the rank-2 grid ( $r = 2$ )

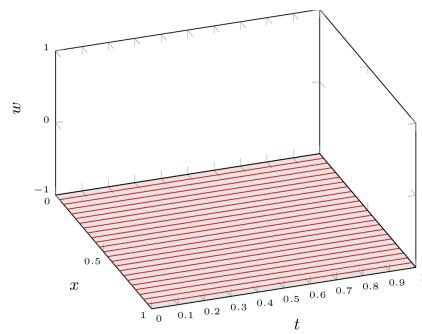
Figure 5: 90 degrees rotation of character “A”

## Numerical Example: Second-order wave equation

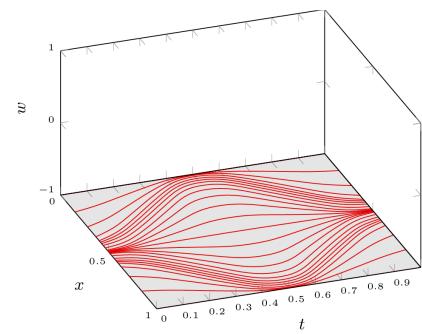
- Consider the second-order wave equation

$$\begin{cases} \frac{\partial^2 w(x, t)}{\partial t^2} - \frac{\partial^2 w(x, t)}{\partial x^2} = 0, & (x, t) \in [0, 1] \times [0, 1] \\ w(x, 0) = e^{-((x-0.5)/0.05)^2} \\ \partial_t w(x, 0) = 0 \\ w(0, t) = 0 \\ w(1, t) = 0 \end{cases} \quad (12)$$

## Numerical Example: Second-order wave equation



(a) Eulerian grid



(b) Rank-2 grid

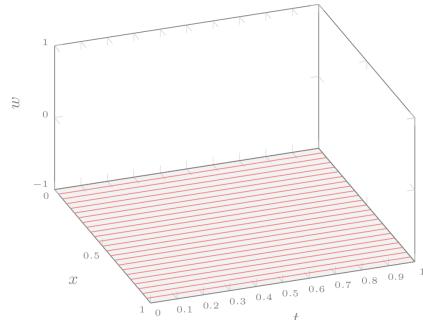
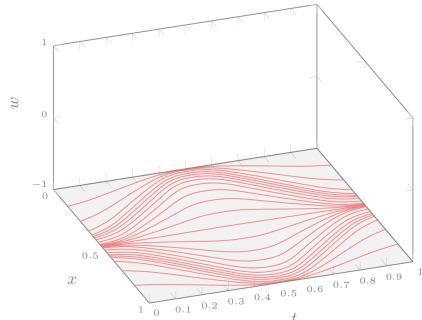
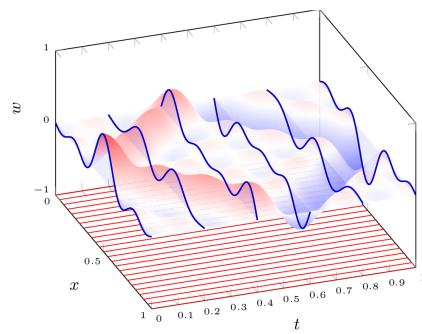
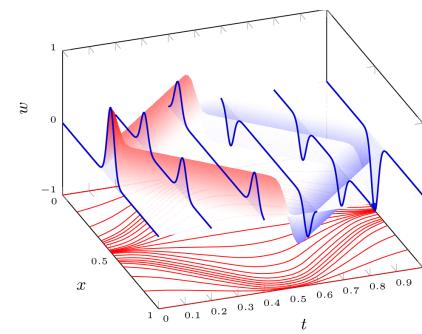
(c) Eulerian grid,  $k = 8$  POD(d) Rank-2 grid,  $k_r = 8$ 

Figure 6: Reconstruction of second-order wave equation

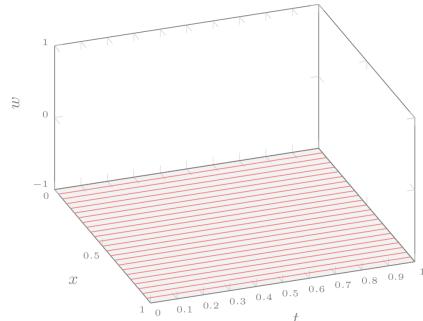
## Numerical Example: Second-order wave equation



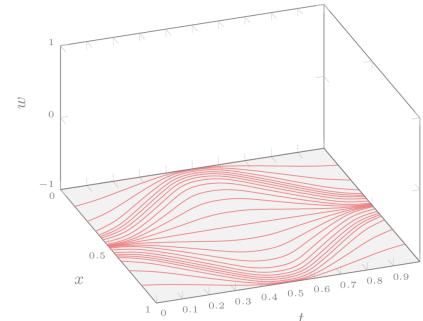
(a) Eulerian grid,  $k = 4$  POD



(b) Rank-2 grid,  $k_r = 4$



(c) Eulerian grid,  $k = 8$  POD



(d) Rank-2 grid,  $k_r = 8$

Figure 6: Reconstruction of second-order wave equation

## Numerical Example: Second-order wave equation

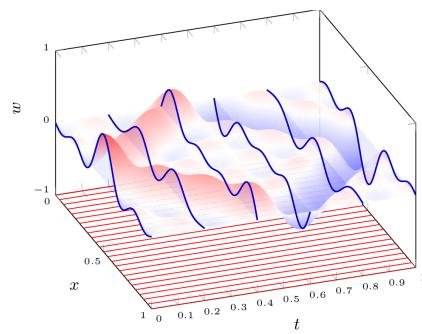
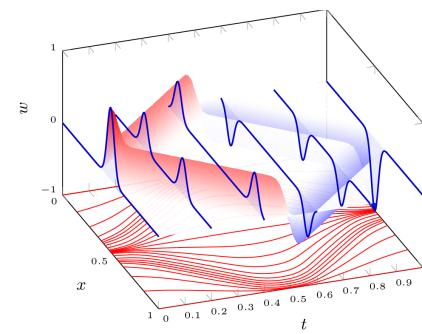
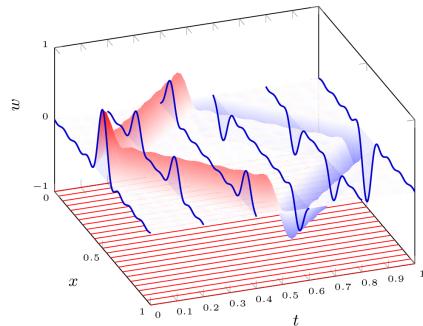
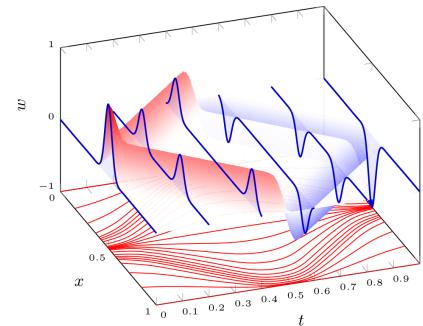
(a) Eulerian grid,  $k = 4$  POD(b) Rank-2 grid,  $k_r = 4$ (c) Eulerian grid,  $k = 8$  POD(d) Rank-2 grid,  $k_r = 8$ 

Figure 6: Reconstruction of second-order wave equation

## Numerical Example: Second-order wave equation

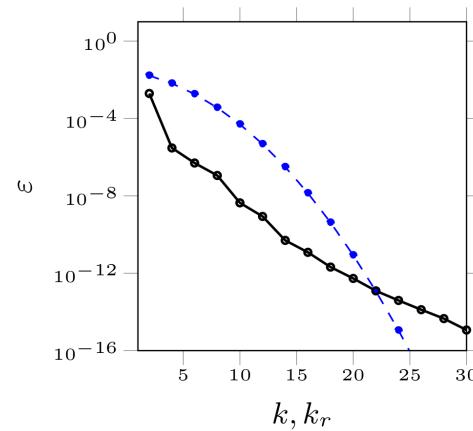


Figure 7: Reconstruction error in second-order wave problem. The rank- $k$  reconstruction on traditional POD subspace (dashed blue line), and rank- $k_r$  reconstruction on the identified manifold of rank-2 grid (solid black line)

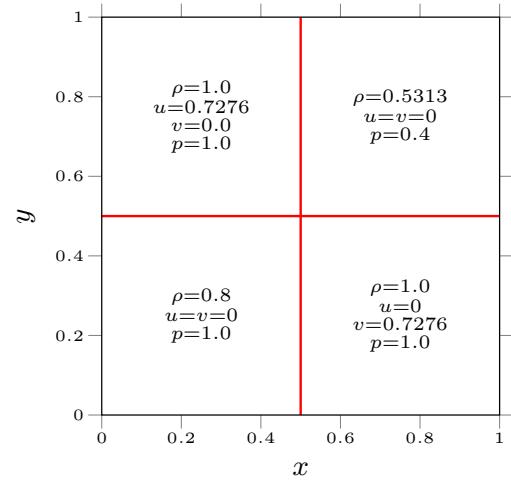
## Numerical Example: Two-dimensional Riemann problem

- Consider the Euler equations

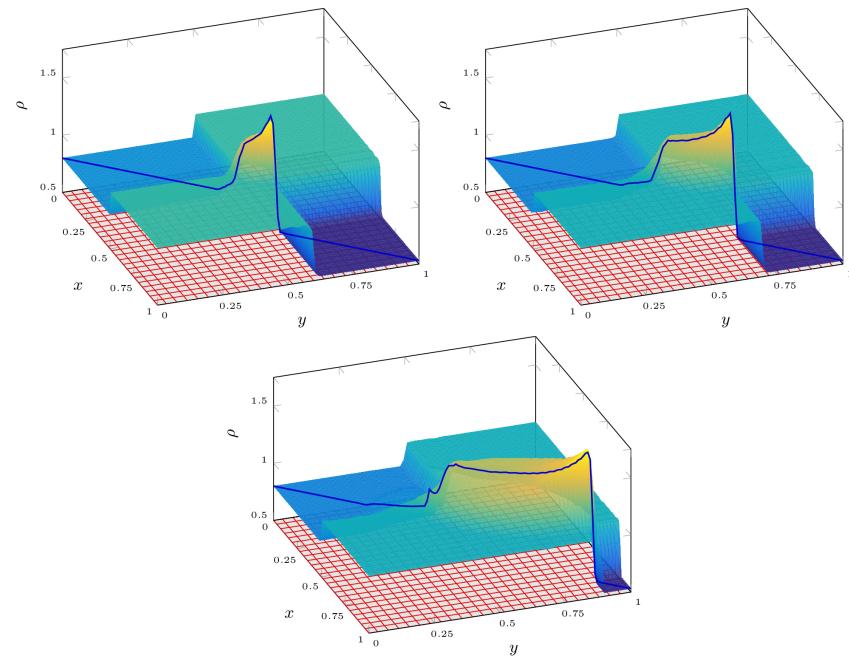
$$\left\{ \begin{array}{lcl} \frac{\partial}{\partial t} \mathbf{q} + \frac{\partial}{\partial x} \mathbf{f}_x + \frac{\partial}{\partial y} \mathbf{f}_y & = & 0, \quad (x, y, t) \in [0, 1] \times [0, 1] \times [0, t_{\max}] \\ \mathbf{q} & = & [\rho, \rho u, \rho v, \rho e]^T \\ \mathbf{f}_x & = & [\rho u, \rho u^2 + p, \rho uv, \rho uH]^T \\ \mathbf{f}_y & = & [\rho v, \rho uv + p, \rho v^2 + p, \rho vH]^T \\ H & = & e + p/\rho \\ p & = & \rho (\gamma - 1) (e - 0.5 (u^2 + v^2)) \end{array} \right. \quad (13)$$

- Stabilized using high-order artificial viscosity

## Two-dimensional Riemann problem



(a) Initial condition,  
configuration 12  
[Lax and Liu, 1998]



(b) Density at  $t \in \{0.0625, 0.125, 0.25\}$

Figure 8: Two-dimensional Riemann problem

## Two-dimensional Riemann problem

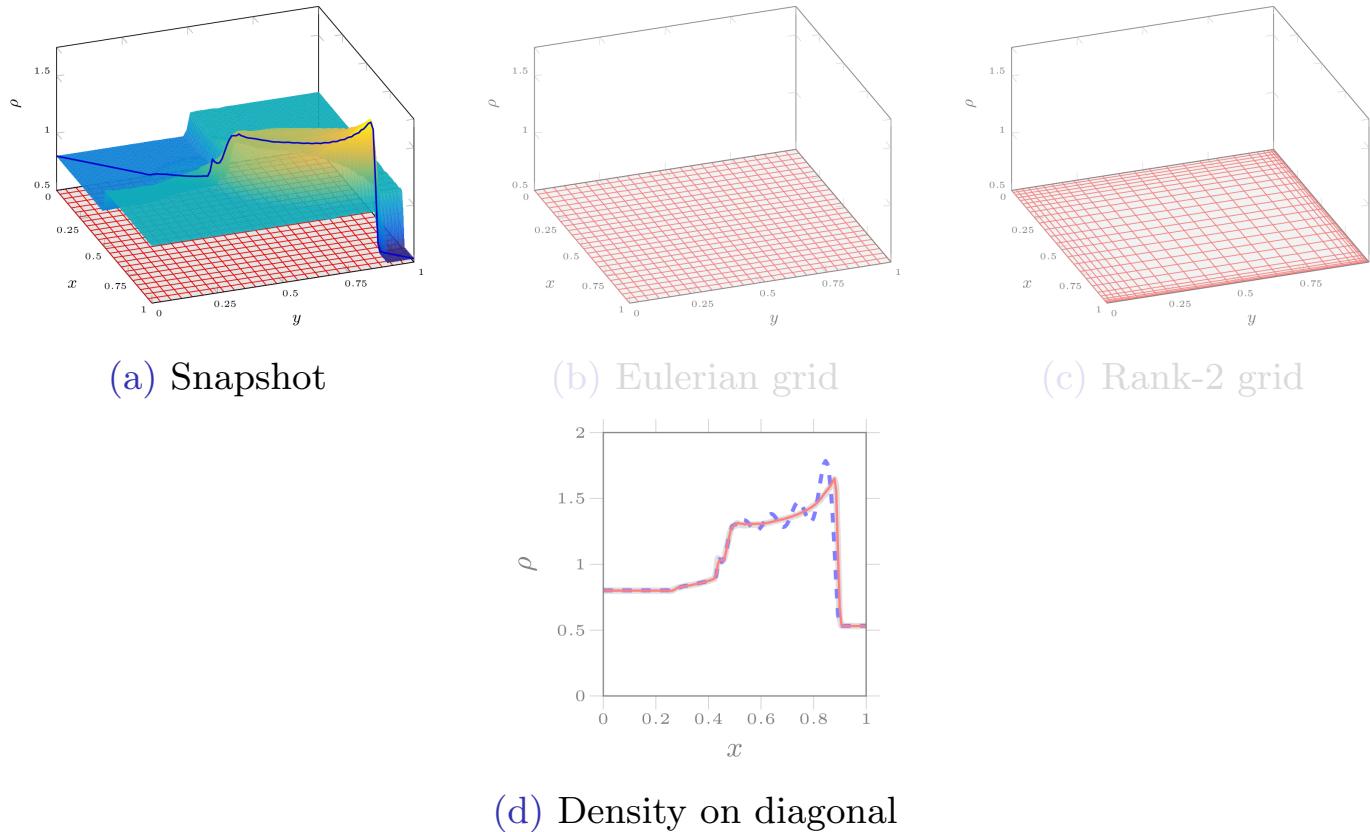
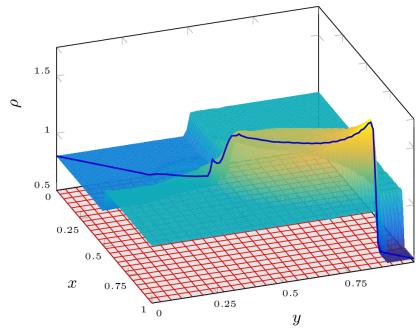
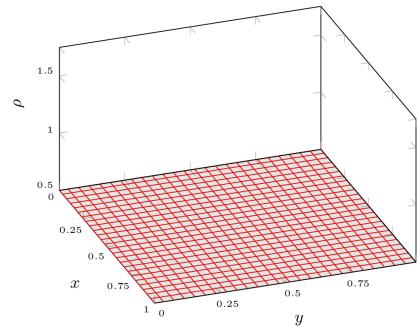


Figure 9: Density at  $t = 0.25$

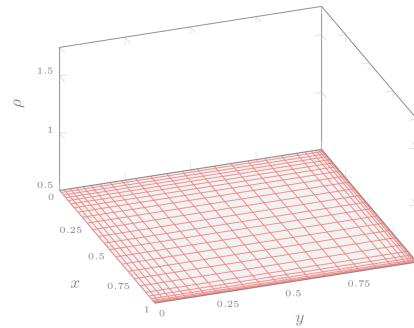
## Two-dimensional Riemann problem



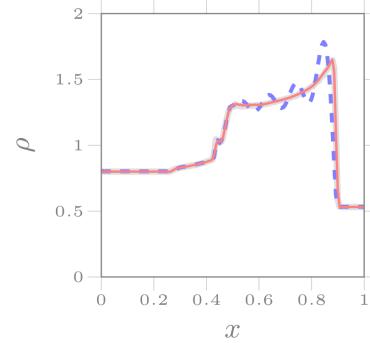
(a) Snapshot



(b) Eulerian grid



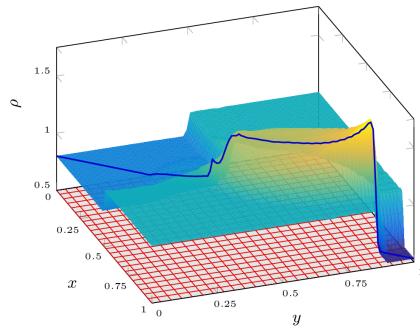
(c) Rank-2 grid



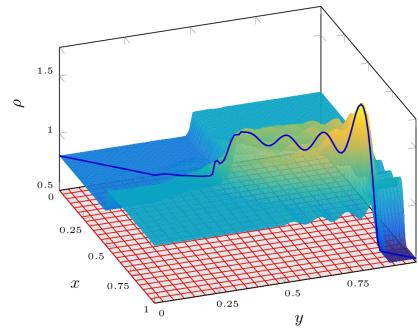
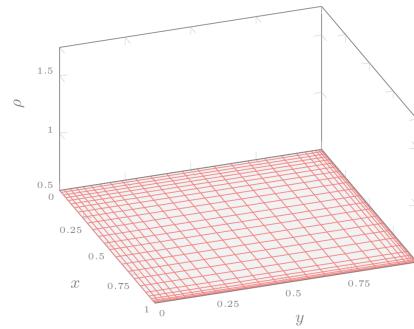
(d) Density on diagonal

Figure 9: Density at  $t = 0.25$

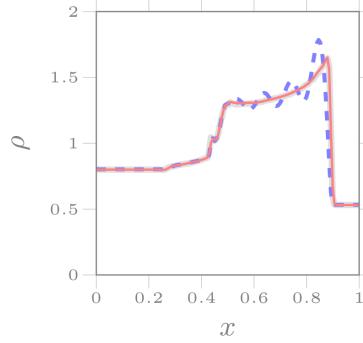
## Two-dimensional Riemann problem



(a) Snapshot

(b) Eulerian grid,  $k = 8$ 

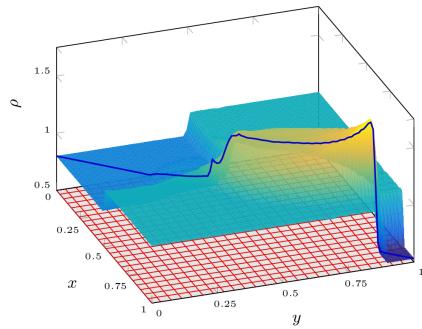
(c) Rank-2 grid



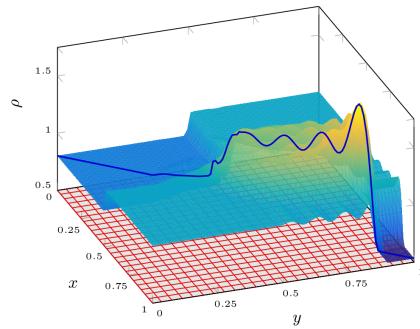
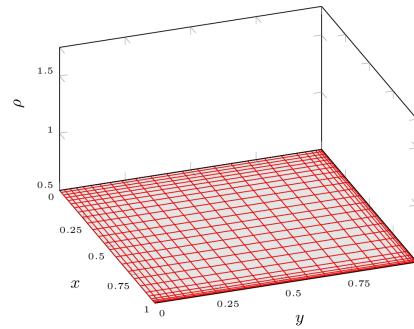
(d) Density on diagonal

Figure 9: Density at  $t = 0.25$

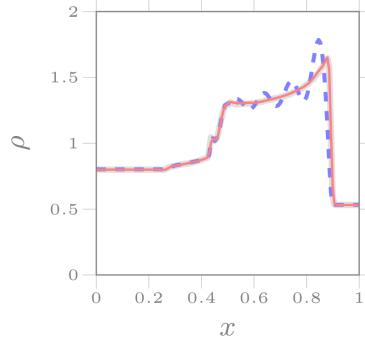
## Two-dimensional Riemann problem



(a) Snapshot

(b) Eulerian grid,  $k = 8$ 

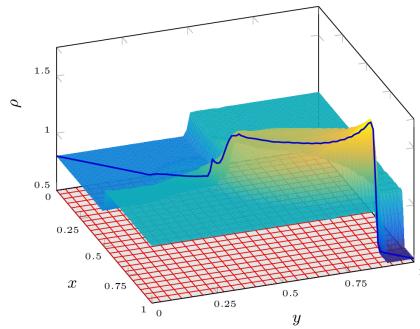
(c) Rank-2 grid



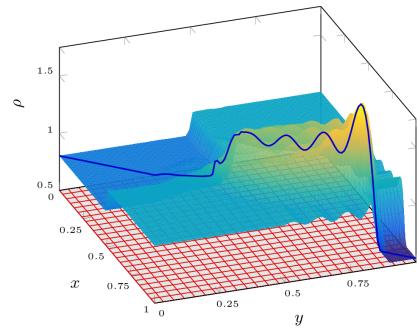
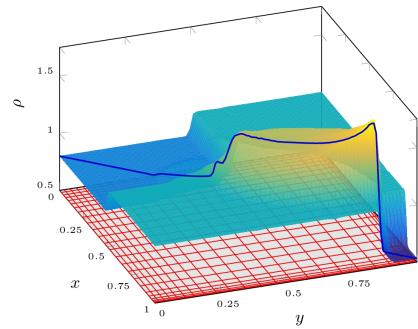
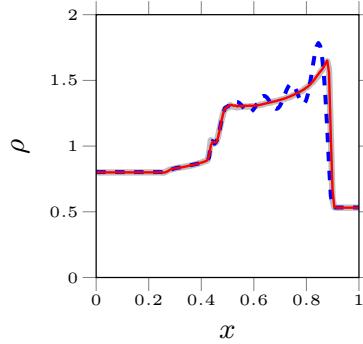
(d) Density on diagonal

Figure 9: Density at  $t = 0.25$

## Two-dimensional Riemann problem



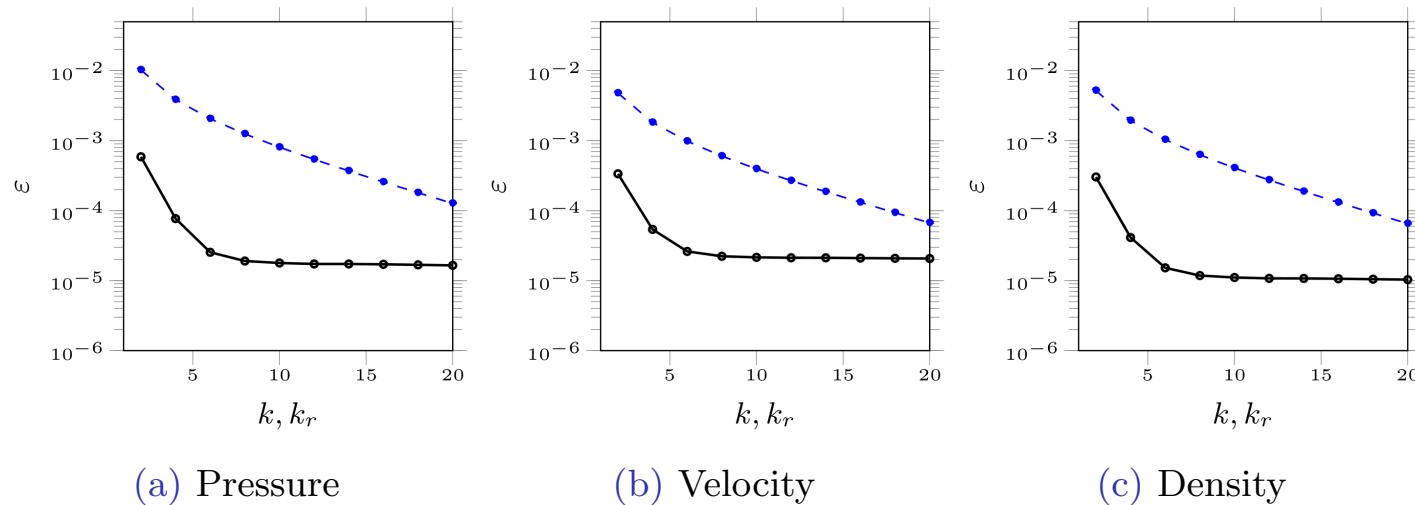
(a) Snapshot

(b) Eulerian grid,  $k = 8$ (c) Rank-2 grid,  $k_r = 8$ 

(d) Density on diagonal

Figure 9: Density at  $t = 0.25$

## Two-dimensional Riemann problem



**Figure 10:** Reconstruction error. The rank- $k$  reconstruction on POD subspace (dashed blue line), and rank- $k_r$  reconstruction on the identified manifold of  $r = 2$  (solid black line)

## Summary

- Snapshots featuring convection cannot be efficiently compressed using traditional approaches
- A low-rank grid can be identified to efficiently compress/reduce such snapshots
- The proposed manifold approach *efficiently registers the output sequence of the PDEs on a low-rank time-varying grid*

## Summary

- Snapshots featuring convection cannot be efficiently compressed using traditional approaches
- A low-rank grid can be identified to efficiently compress/reduce such snapshots
- The proposed manifold approach *efficiently registers the output sequence of the PDEs on a low-rank time-varying grid*

## Summary

- Snapshots featuring convection cannot be efficiently compressed using traditional approaches
- A low-rank grid can be identified to efficiently compress/reduce such snapshots
- The proposed manifold approach *efficiently registers the output sequence of the PDEs on a low-rank time-varying grid*

# Outline

## 1 Introduction

- Fundamentals
- Motivation

## 2 Identifying an optimal manifold

- Physics-aware registration-based manifold

## 3 ROMs on the identified manifold

- Projection based ROMs
- Neural Network based ROMs

## 4 Stabilization of time-varying ROMs

- A feedback controller approach

## 5 Summary

## Constructing the ROMs

- Constructing the dynamical systems on the identified manifolds
  - Governing equations are available:
    - Projection-based reduced order models
  - Governing equations are *not* available:
    - System identification: Volterra series, Nonlinear Auto-Regressive Moving Average with eXogenous input (NARMAX), ...
    - Neural Networks: Recurrent Neural Networks (RNNs), ...

## Constructing the ROMs

- Constructing the dynamical systems on the identified manifolds
  - Governing equations are available:
    - Projection-based reduced order models
  - Governing equations are *not* available:
    - System identification: Volterra series, Nonlinear Auto-Regressive Moving Average with eXogenous input (NARMAX), ...
    - Neural Networks: Recurrent Neural Networks (RNNs), ...

## Step 1: identifying the POD subspace

- Given a snapshot matrix,  $\mathbf{M} \in \mathbb{R}^{N \times K}$

$$\mathbf{M} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_3 \quad \cdots \quad \mathbf{w}_K]; \quad \mathbf{w}_i \in \mathbb{R}^N \quad (14)$$

- The low-rank approximation problem is

$$\underset{\text{rank}(\widetilde{\mathbf{M}})=k}{\text{minimize}} \quad \|\mathbf{M} - \widetilde{\mathbf{M}}\|_F \quad (15)$$

- The rank constraint is satisfied

$$\underset{\mathbf{U} \in \mathbb{R}^{N \times k}, \mathbf{V} \in \mathbb{R}^{k \times K}}{\text{minimize}} \quad \|\mathbf{M} - \mathbf{UV}\|_F \quad (16)$$

- The solution is given by SVD/POD of the snapshot matrix

## Step 1: identifying the POD subspace

- Given a snapshot matrix,  $\mathbf{M} \in \mathbb{R}^{N \times K}$

$$\mathbf{M} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_3 \quad \cdots \quad \mathbf{w}_K]; \quad \mathbf{w}_i \in \mathbb{R}^N \quad (14)$$

- The low-rank approximation problem is

$$\underset{\text{rank}(\widetilde{\mathbf{M}})=k}{\text{minimize}} \quad \|\mathbf{M} - \widetilde{\mathbf{M}}\|_F \quad (15)$$

- The rank constraint is satisfied

$$\underset{\mathbf{U} \in \mathbb{R}^{N \times k}, \mathbf{V} \in \mathbb{R}^{k \times K}}{\text{minimize}} \quad \|\mathbf{M} - \mathbf{UV}\|_F \quad (16)$$

- The solution is given by SVD/POD of the snapshot matrix

## Step 1: identifying the POD subspace

- Given a snapshot matrix,  $\mathbf{M} \in \mathbb{R}^{N \times K}$

$$\mathbf{M} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_3 \quad \cdots \quad \mathbf{w}_K]; \quad \mathbf{w}_i \in \mathbb{R}^N \quad (14)$$

- The low-rank approximation problem is

$$\underset{\text{rank}(\widetilde{\mathbf{M}})=k}{\text{minimize}} \quad \|\mathbf{M} - \widetilde{\mathbf{M}}\|_F \quad (15)$$

- The rank constraint is satisfied

$$\underset{\mathbf{U} \in \mathbb{R}^{N \times k}, \mathbf{V} \in \mathbb{R}^{k \times K}}{\text{minimize}} \quad \|\mathbf{M} - \mathbf{UV}\|_F \quad (16)$$

- The solution is given by SVD/POD of the snapshot matrix

## Step 1: identifying the POD subspace

- Given a snapshot matrix,  $\mathbf{M} \in \mathbb{R}^{N \times K}$

$$\mathbf{M} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_3 \quad \cdots \quad \mathbf{w}_K]; \quad \mathbf{w}_i \in \mathbb{R}^N \quad (14)$$

- The low-rank approximation problem is

$$\underset{\text{rank}(\widetilde{\mathbf{M}})=k}{\text{minimize}} \quad \|\mathbf{M} - \widetilde{\mathbf{M}}\|_F \quad (15)$$

- The rank constraint is satisfied

$$\underset{\mathbf{U} \in \mathbb{R}^{N \times k}, \mathbf{V} \in \mathbb{R}^{k \times K}}{\text{minimize}} \quad \|\mathbf{M} - \mathbf{UV}\|_F \quad (16)$$

- The solution is given by SVD/POD of the snapshot matrix

## Step 2: constructing the reduced order model

- Consider the discretized system

$$\mathbf{R}(\mathbf{w}[n]) = \mathbf{0} \quad (17)$$

- The state,  $\mathbf{w}$ , is approximated on a global trial subspace,

$$\mathbf{w}[n] \approx \tilde{\mathbf{w}}[n] = \mathbf{U}\mathbf{w}_r[n], \quad \mathbf{U} \in \mathbb{R}^{N \times k}, \quad \mathbf{w}_r[n] \in \mathbb{R}^k, \quad k \ll N \quad (18)$$

- Galerkin projection onto the test subspace  $\mathbf{U} \in \mathbb{R}^{N \times k}$ ,

$$\mathbf{U}^T \mathbf{R}(\mathbf{U}\mathbf{w}_r[n]) = \mathbf{0} \quad (19)$$

## Step 2: constructing the reduced order model

- Consider the discretized system

$$\mathbf{R}(\mathbf{w}[n]) = \mathbf{0} \quad (17)$$

- The state,  $\mathbf{w}$ , is approximated on a global trial subspace,

$$\mathbf{w}[n] \approx \tilde{\mathbf{w}}[n] = \mathbf{U}\mathbf{w}_r[n], \quad \mathbf{U} \in \mathbb{R}^{N \times k}, \quad \mathbf{w}_r[n] \in \mathbb{R}^k, \quad k \ll N \quad (18)$$

- Galerkin projection onto the test subspace  $\mathbf{U} \in \mathbb{R}^{N \times k}$ ,

$$\mathbf{U}^T \mathbf{R}(\mathbf{U}\mathbf{w}_r[n]) = \mathbf{0} \quad (19)$$

## Step 2: constructing the reduced order model

- Consider the discretized system

$$\mathbf{R}(\mathbf{w}[n]) = \mathbf{0} \quad (17)$$

- The state,  $\mathbf{w}$ , is approximated on a global trial subspace,

$$\mathbf{w}[n] \approx \tilde{\mathbf{w}}[n] = \mathbf{U}\mathbf{w}_r[n], \quad \mathbf{U} \in \mathbb{R}^{N \times k}, \quad \mathbf{w}_r[n] \in \mathbb{R}^k, \quad k \ll N \quad (18)$$

- Galerkin projection onto the test subspace  $\mathbf{U} \in \mathbb{R}^{N \times k}$ ,

$$\mathbf{U}^T \mathbf{R}(\mathbf{U}\mathbf{w}_r[n]) = \mathbf{0} \quad (19)$$

## The proposed ROM

- Recall

$$\mathbf{w}[n] \approx \mathbf{U}\mathbf{w}_r[n] \quad (20)$$

- In the proposed new approach,

$$\mathbf{w}[n] \approx \mathbf{U}[n]\mathbf{w}_r[n] \quad (21)$$

where  $\mathcal{G}^{-1}(\mathbf{U}) = \{\mathbf{U}[0], \mathbf{U}[1], \dots\}$

- For Galerkin ROM on the proposed manifold

$$\mathbf{U}^T[n]\mathbf{R}(\mathbf{U}[n]\mathbf{w}_r[n]) = \mathbf{0} \quad (22)$$

## The proposed ROM

- Recall

$$\mathbf{w}[n] \approx \mathbf{U}\mathbf{w}_r[n] \quad (20)$$

- In the proposed new approach,

$$\mathbf{w}[n] \approx \mathbf{U}[n]\mathbf{w}_r[n] \quad (21)$$

where  $\mathcal{G}^{-1}(\mathbf{U}) = \{\mathbf{U}[0], \mathbf{U}[1], \dots\}$

- For Galerkin ROM on the proposed manifold

$$\mathbf{U}^T[n]\mathbf{R}(\mathbf{U}[n]\mathbf{w}_r[n]) = \mathbf{0} \quad (22)$$

## The proposed ROM

- Recall

$$\mathbf{w}[n] \approx \mathbf{U}\mathbf{w}_r[n] \quad (20)$$

- In the proposed new approach,

$$\mathbf{w}[n] \approx \mathbf{U}[n]\mathbf{w}_r[n] \quad (21)$$

where  $\mathcal{G}^{-1}(\mathbf{U}) = \{\mathbf{U}[0], \mathbf{U}[1], \dots\}$

- For Galerkin ROM on the proposed manifold

$$\mathbf{U}^T[n]\mathbf{R}(\mathbf{U}[n]\mathbf{w}_r[n]) = \mathbf{0} \quad (22)$$

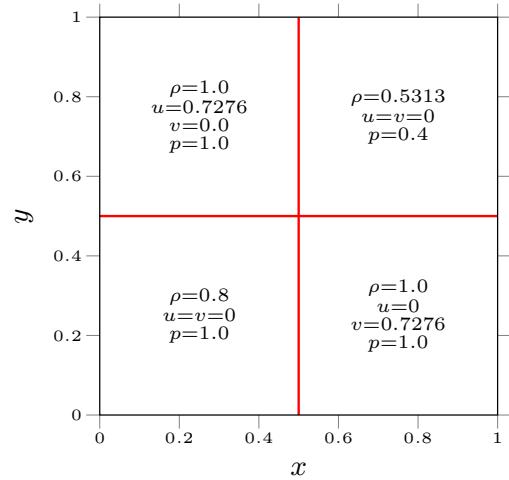
## Numerical Example: Two-dimensional Riemann problem

- Consider the Euler equations

$$\left\{ \begin{array}{lcl} \frac{\partial}{\partial t} \mathbf{q} + \frac{\partial}{\partial x} \mathbf{f}_x + \frac{\partial}{\partial y} \mathbf{f}_y & = & 0, \quad (x, y, t) \in [0, 1] \times [0, 1] \times [0, t_{\max}] \\ \mathbf{q} & = & [\rho, \rho u, \rho v, \rho e]^T \\ \mathbf{f}_x & = & [\rho u, \rho u^2 + p, \rho uv, \rho uH]^T \\ \mathbf{f}_y & = & [\rho v, \rho uv + p, \rho v^2 + p, \rho vH]^T \\ H & = & e + p/\rho \\ p & = & \rho (\gamma - 1) (e - 0.5 (u^2 + v^2)) \end{array} \right. \quad (23)$$

- Stabilized using high-order artificial viscosity

## Numerical Example: Two-dimensional Riemann problem



(a) Initial condition,  
configuration 12  
[Lax and Liu, 1998]

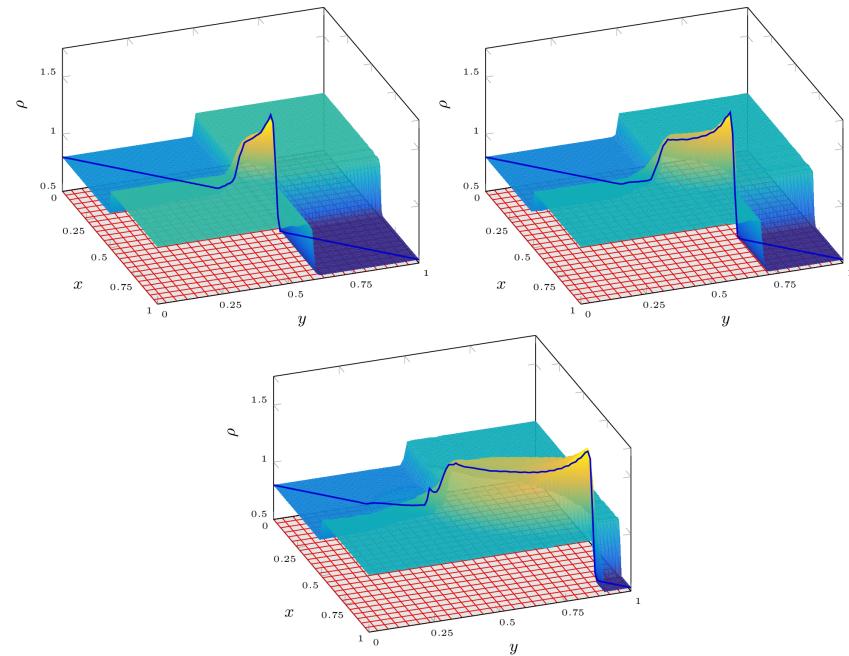
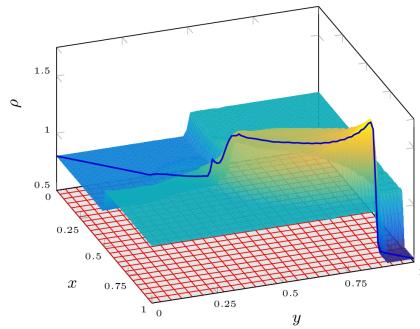
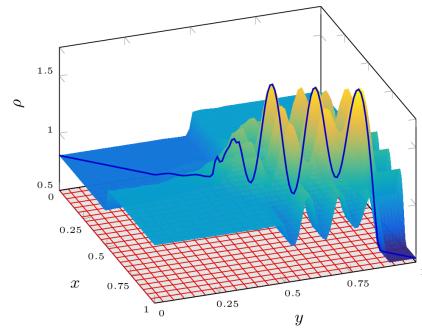
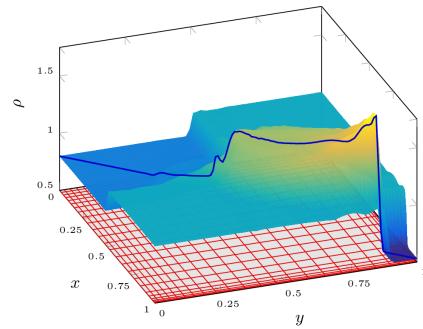
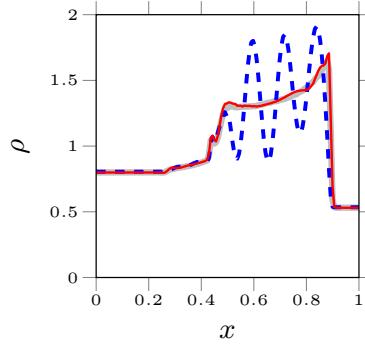


Figure 11: Two-dimensional Riemann problem

## Numerical Example: Two-dimensional Riemann problem



(a) Snapshot

(b) Eulerian grid,  $k = 8$ (c) Rank-2 grid,  $k_r = 8$ 

(d) Density on diagonal

Figure 12: Density at  $t = 0.25$

## Numerical Example: Two-dimensional Riemann problem

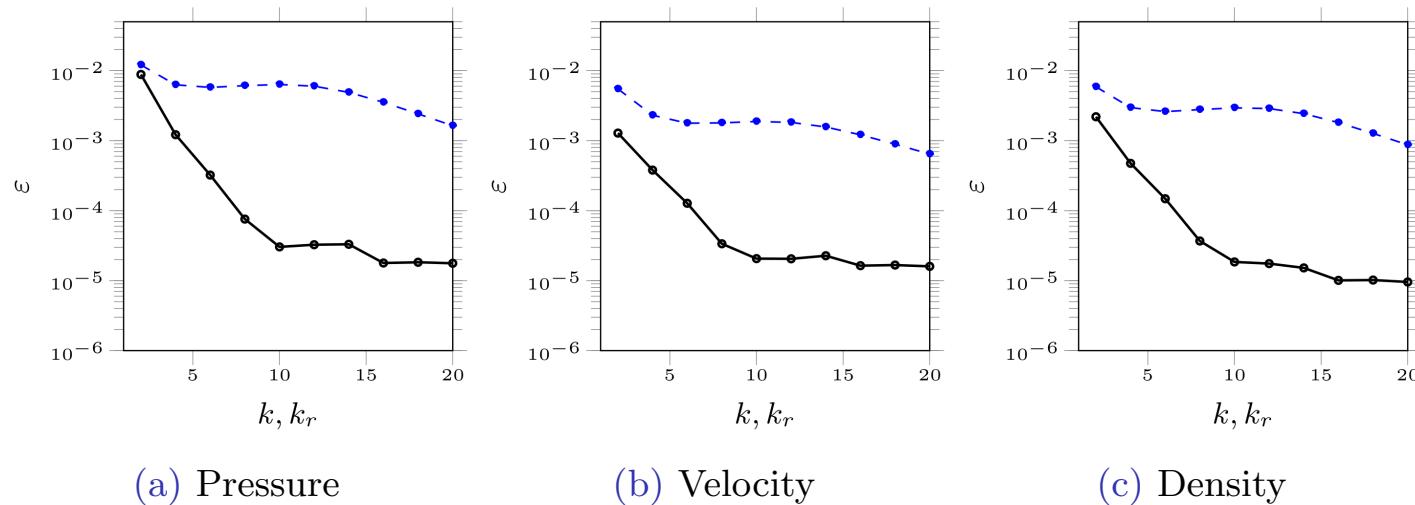
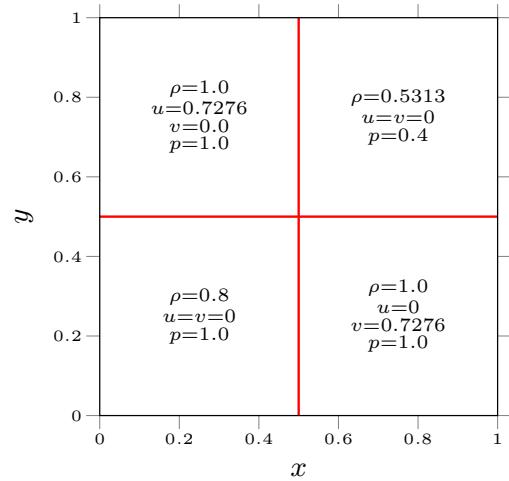


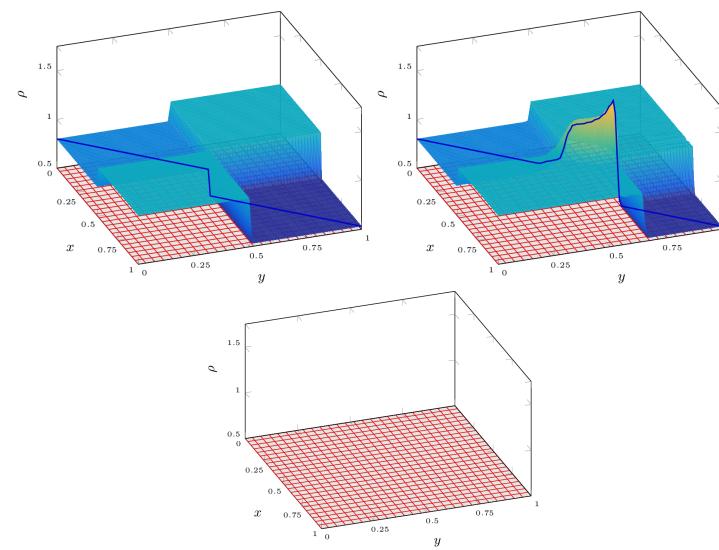
Figure 13: ROM error. The rank- $k$  on POD subspace (dashed blue line), and rank- $k_r$  on the identified rank-2 manifold (solid black line)

## Numerical Example: Two-dimensional Riemann problem - Prediction

- The ROMs are evolved beyond the training range



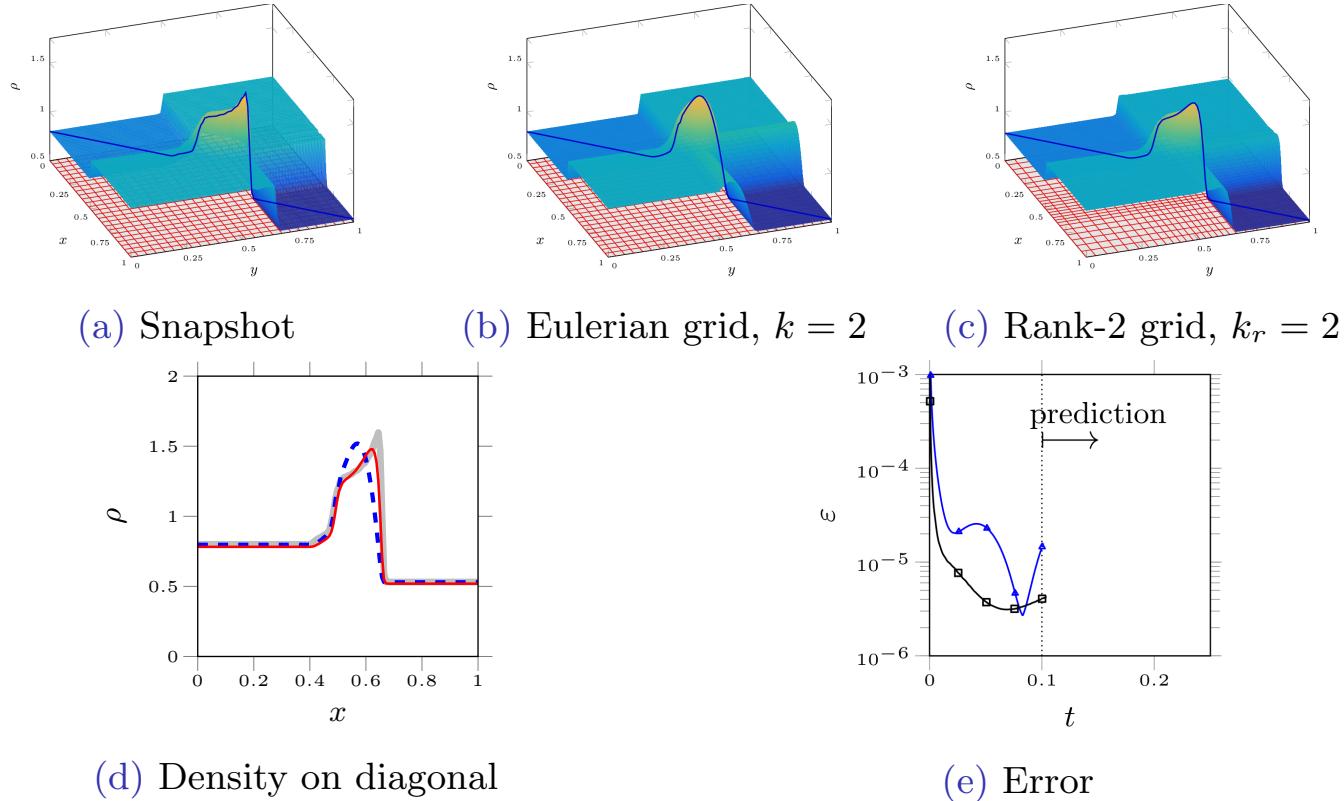
(a) Initial condition,  
configuration 12  
[Lax and Liu, 1998]



(b) Density at  $t \in \{0.0, 0.10\}$

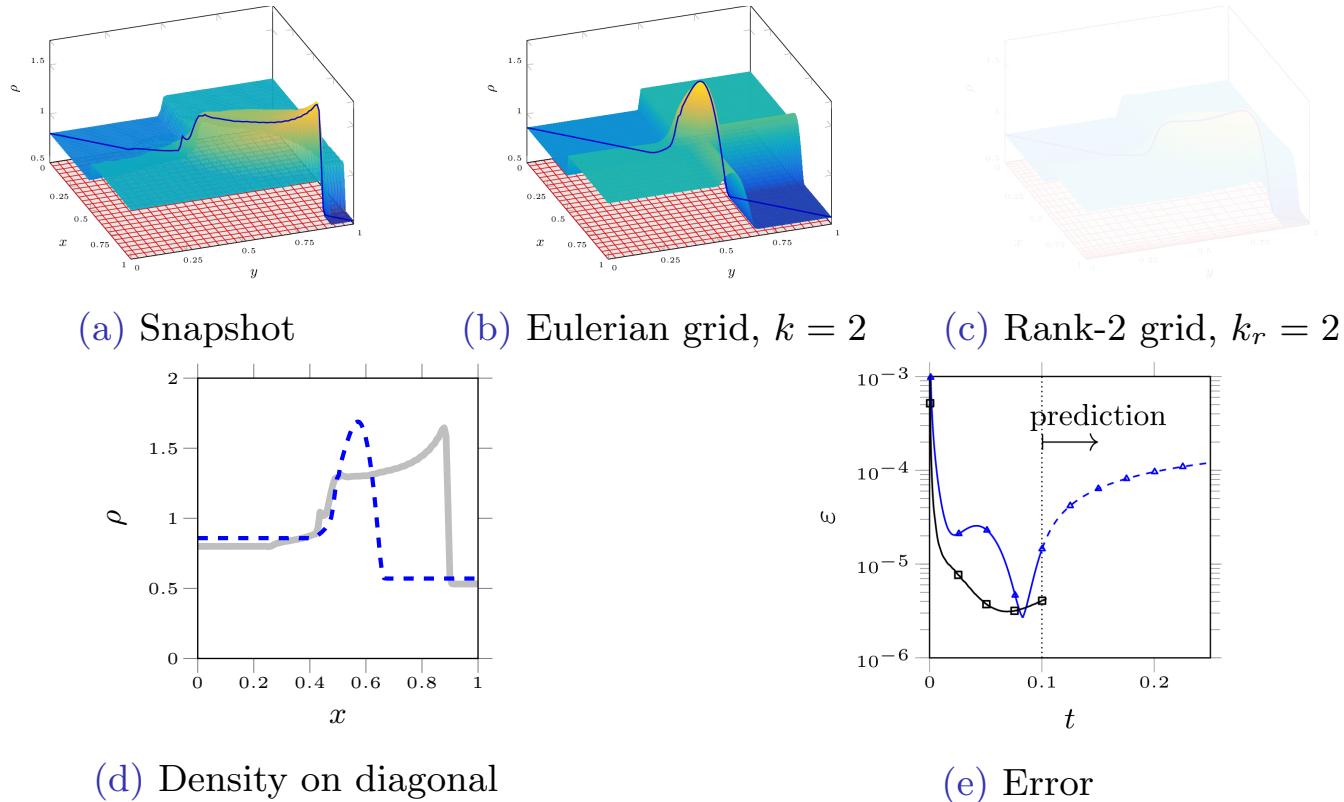
Figure 14: Two-dimensional Riemann problem

## Numerical Example: Two-dimensional Riemann problem - Prediction



**Figure 15:** Density at  $t = 0.1$ , rank-2 ROM error on POD subspace (blue line) and rank-2 grid of the proposed manifold (black line)

## Numerical Example: Two-dimensional Riemann problem - Prediction



**Figure 16:** Density at  $t = 0.25$ , rank-2 ROM error on POD subspace (blue line) and rank-2 grid of the proposed manifold (black line)

## Numerical Example: Two-dimensional Riemann problem - Prediction

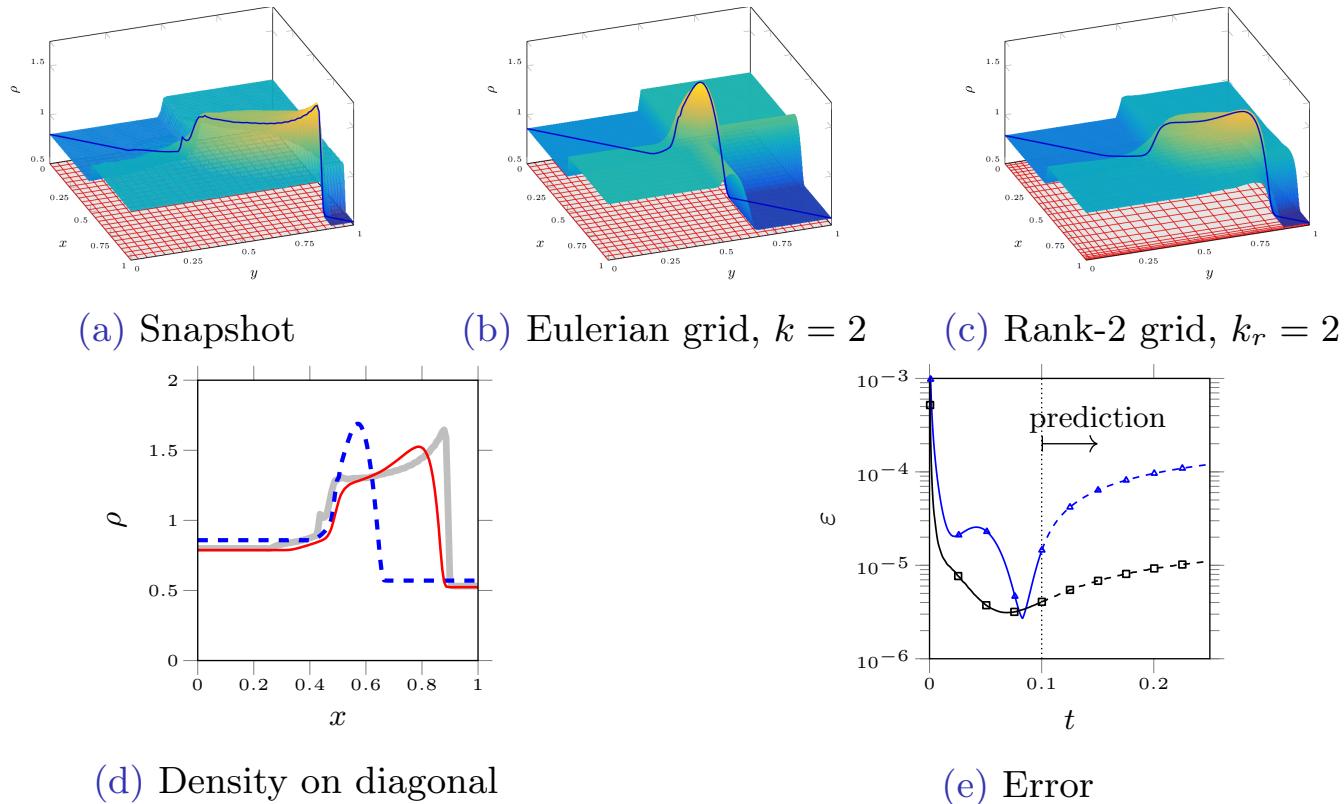


Figure 16: Density at  $t = 0.25$ , rank-2 ROM error on POD subspace (blue line) and rank-2 grid of the proposed manifold (black line)

## Constructing the ROMs

- Constructing the dynamical systems on the identified manifolds
  - Governing equations are available:
    - Projection-based reduced order modeling
  - Governing equations are *not* available:
    - System identification: Volterra series, Nonlinear Auto-Regressive Moving Average with eXogenous input (NARMAX), ...
    - Neural Networks: [Recurrent Neural Networks \(RNNs\)](#), ...

## Neural network-based ROMs

- Auto-encoders are trained to map to the low-dimensional space
- Recurrent neural networks (RNNs) are trained to approximate the dynamics on the low-dimensional space

$$\mathbf{h}[n] = f_{RNN}(\mathbf{h}[n-1]), \mathbf{h}[n] \in \mathbb{R}^k \quad (24)$$

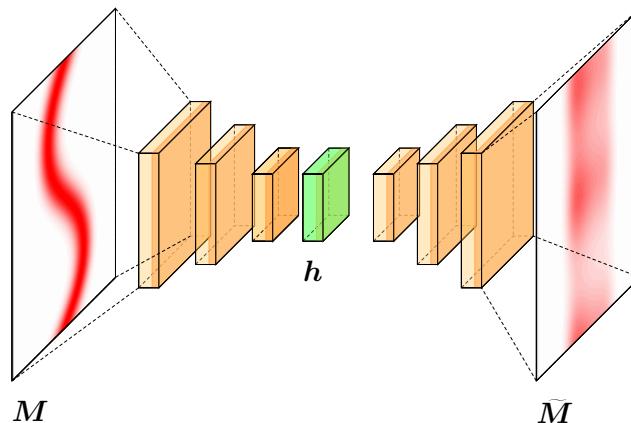


Figure 17: Neural network architecture with traditional auto-encoder

## Neural network-based ROMs

- Auto-encoders are trained to map to the low-dimensional space
- Recurrent neural networks (RNNs) are trained to approximate the dynamics on the low-dimensional space

$$\boldsymbol{h}[n] = f_{RNN}(\boldsymbol{h}[n-1]), \boldsymbol{h}[n] \in \mathbb{R}^k \quad (24)$$

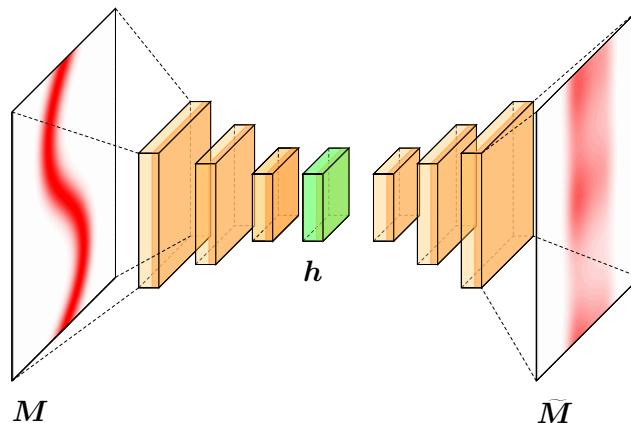
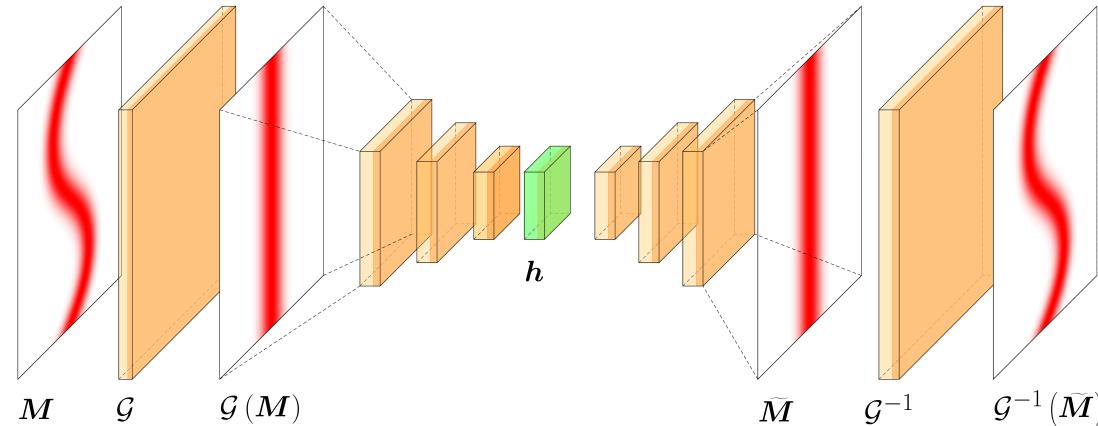


Figure 17: Neural network architecture with traditional auto-encoder

## Neural network-based ROMs

- The traditional architecture is wrapped with the proposed manifold

$$\mathbf{h}[n] = f_{RNN}(\mathbf{h}[n-1]), \mathbf{h}[n] \in \mathbb{R}^{k_r} \quad (25)$$



**Figure 18:** Neural network architecture with traditional auto-encoder wrapped with the proposed physics-aware auto-encoder

## Numerical Example: Burgers' equation

- Consider the viscous Burgers' equation

$$\begin{cases} \frac{\partial w(x,t)}{\partial t} + w \frac{\partial w(x,t)}{\partial x} - 10^{-3} \frac{\partial^2 w(x,t)}{\partial x^2} = 0, & (x,t) \in [0, 2.5] \times [0, 1] \\ w(x,0) = 0.8 + 0.5e^{-(x-0.5)/0.1^2} \\ w(0,t) = 0.8 \\ w(2.5,t) = 0.8 \end{cases} \quad (26)$$

## Numerical Example: Burgers' equation

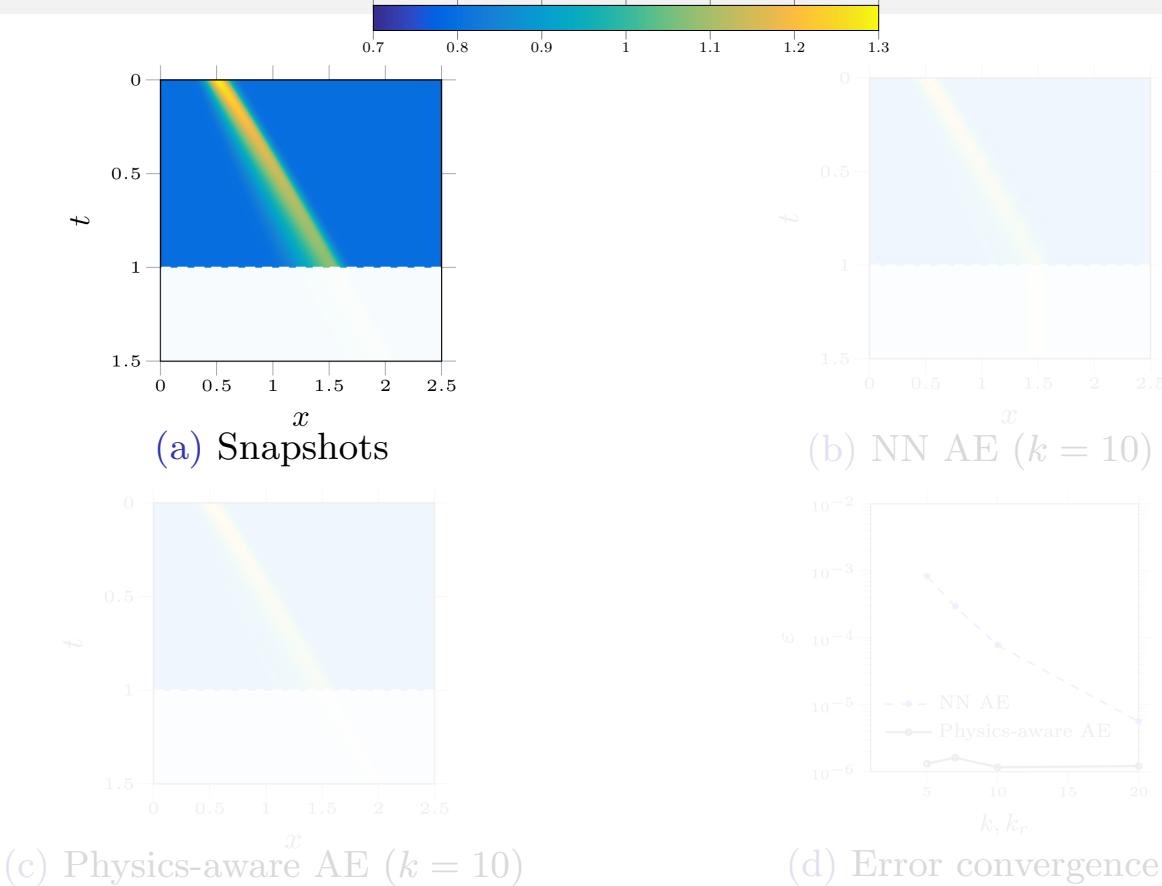


Figure 19: RNN for the Burgers' equation

## Numerical Example: Burgers' equation

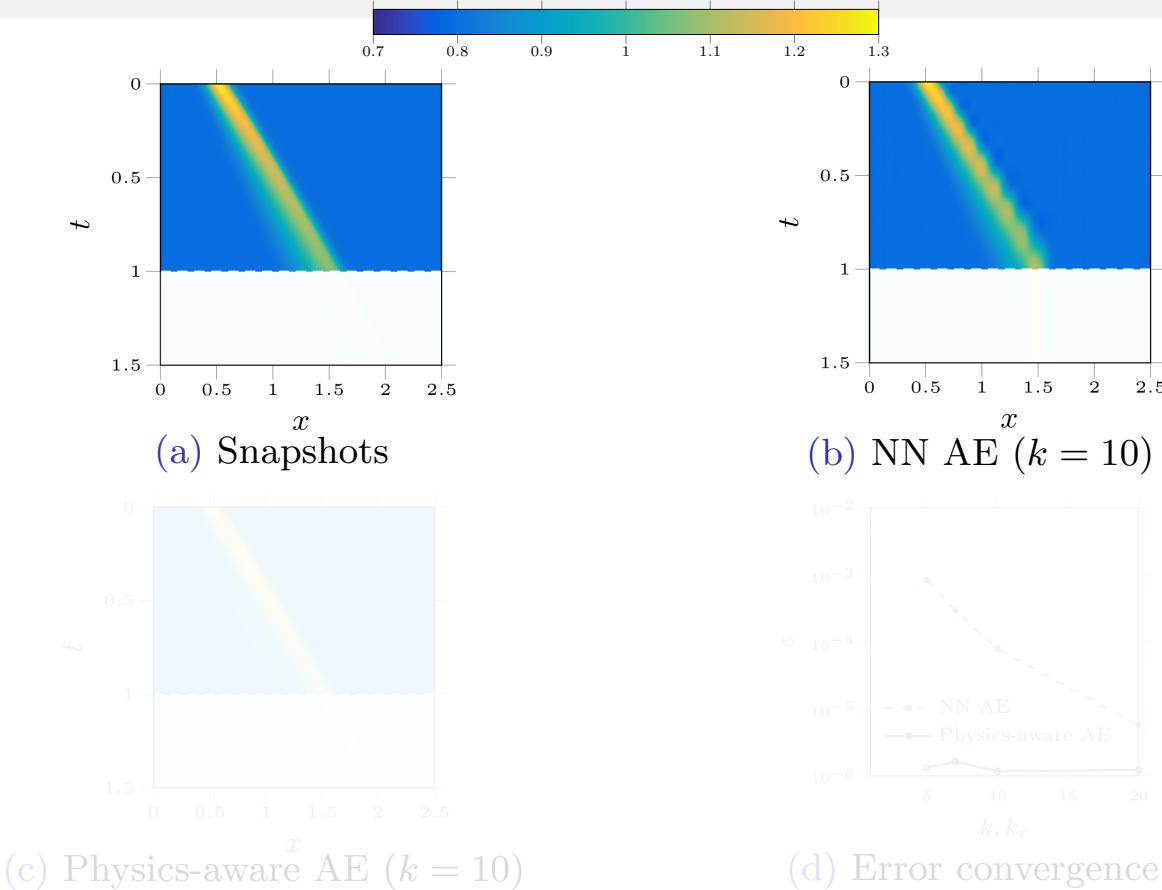


Figure 19: RNN for the Burgers' equation

## Numerical Example: Burgers' equation

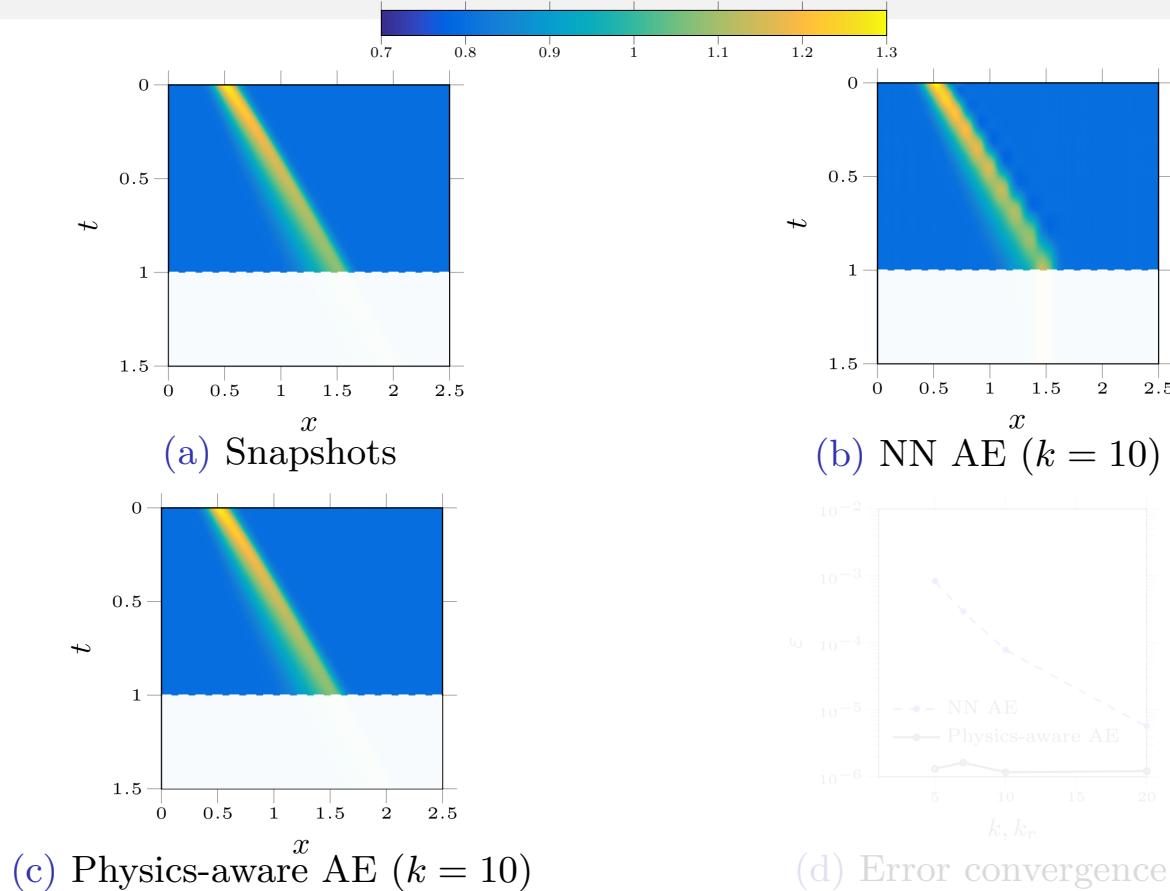


Figure 19: RNN for the Burgers' equation

## Numerical Example: Burgers' equation

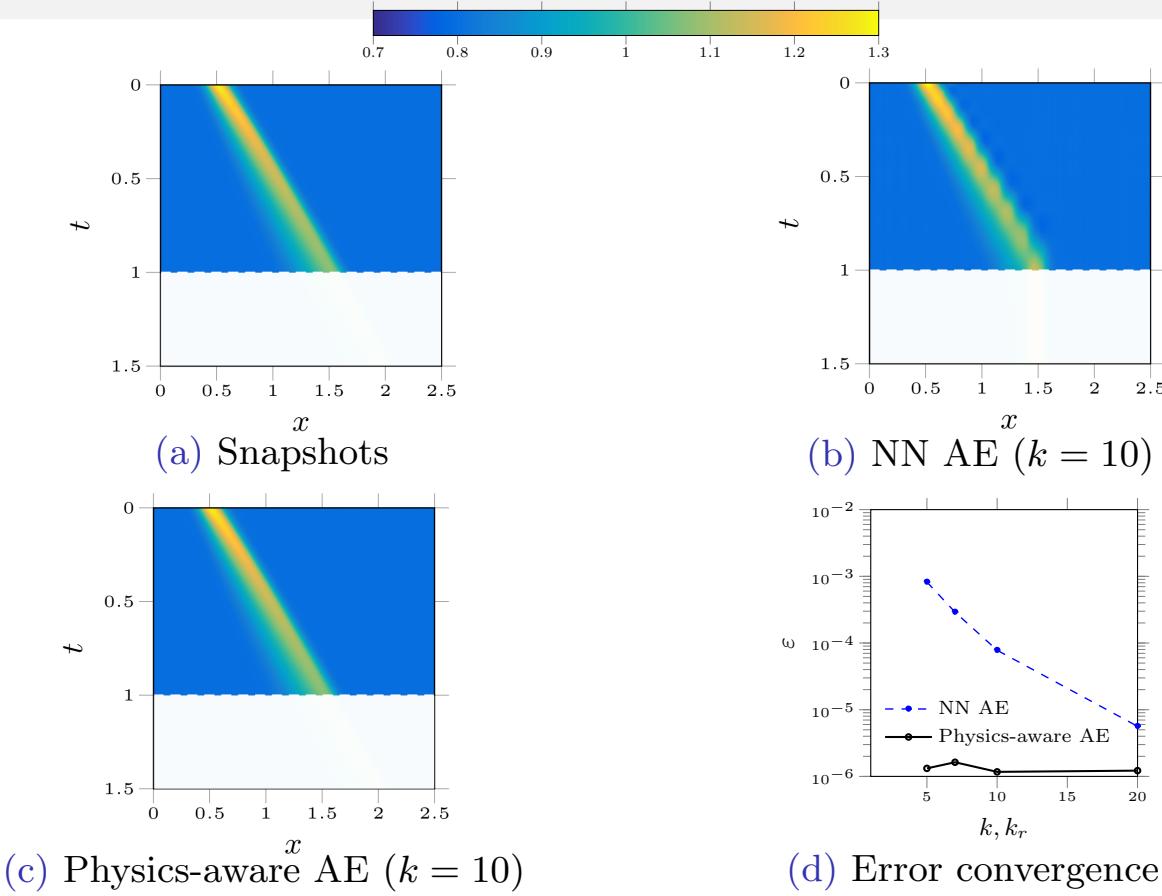


Figure 19: RNN for the Burgers' equation

## Numerical Example: Burgers' equation

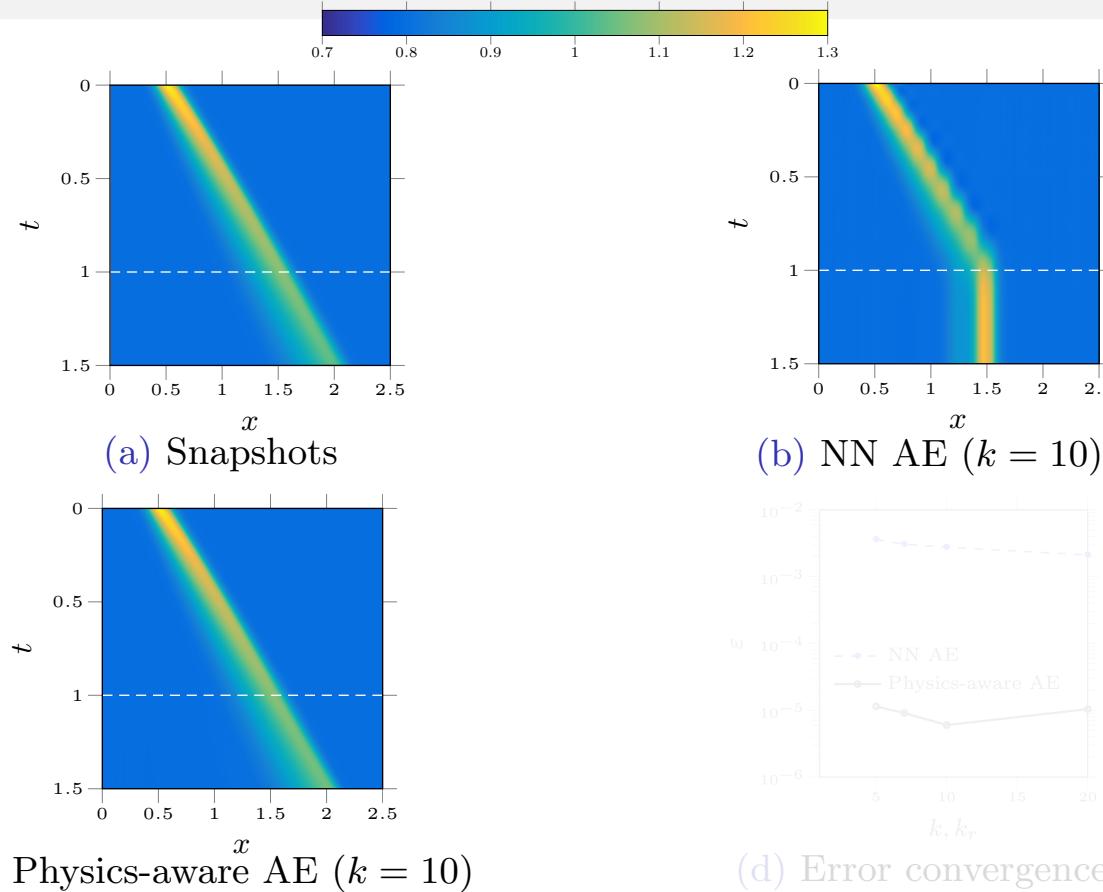


Figure 19: RNN for the Burgers' equation

## Numerical Example: Burgers' equation

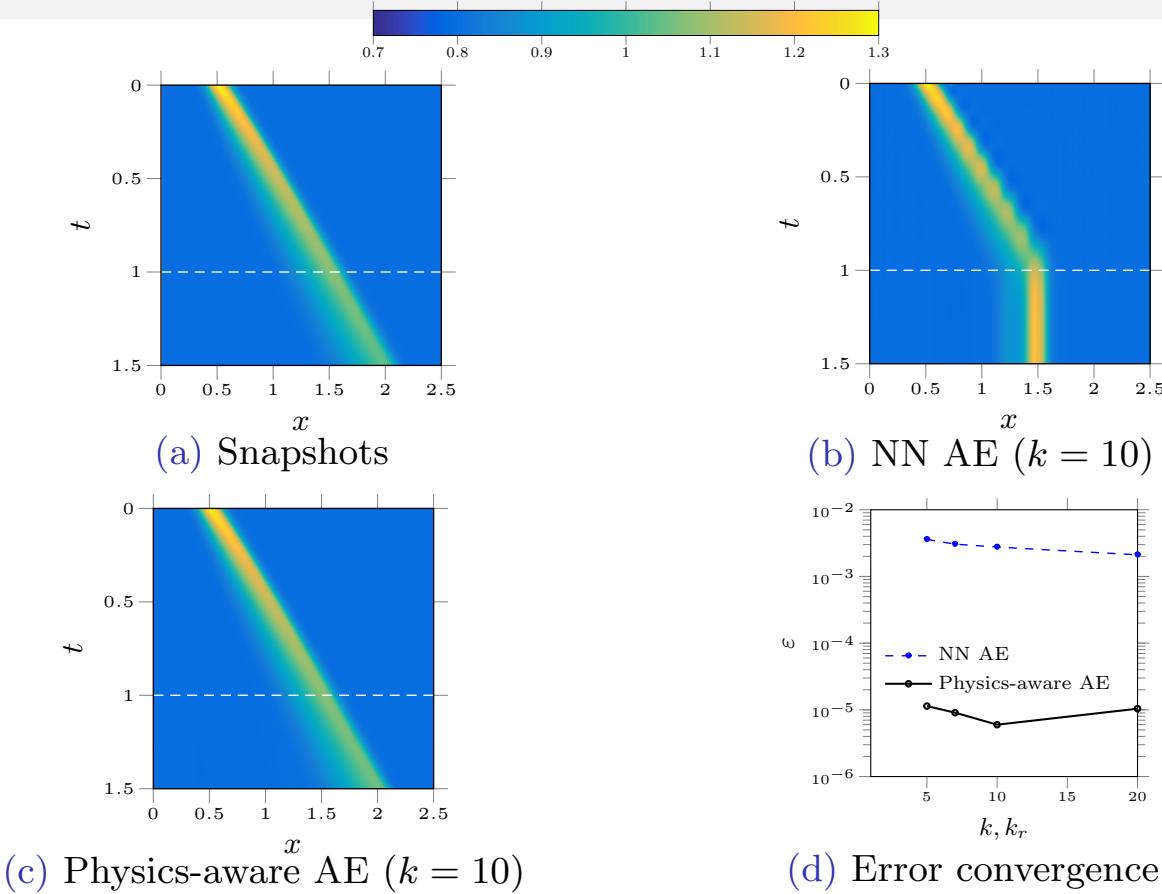


Figure 19: RNN for the Burgers' equation

## Summary

- Non-intrusive ROMs on the proposed manifold can efficiently and accurately approximate the nonlinear convection dominated phenomena
- The proposed ROMs on the identified manifolds enable prediction
- Mojgani, R. and Balajewicz, M. “Projection based reduced order models on physics-aware registration based manifolds”, [under preparation].
- Mojgani, R. and Balajewicz, M. “Physics-aware registration based auto-encoder for convection dominated PDEs”, *AAAI 2021*, [under phase II review].
- Mojgani, R. and Balajewicz, M. “Dimensionality reduction of convection-dominated flows on an optimally morphing grid”, *APS Annual Meeting*, 2019.
- Mojgani, R. and Balajewicz, M. “Arbitrary Lagrangian Eulerian framework for efficient projection-based reduction of convection dominated nonlinear flows”, *APS Division of Fluid Dynamics Meeting*, 2017.
- Mojgani, R. and Balajewicz, M. “Lagrangian basis method for dimensionality reduction of convection dominated nonlinear flows”, *SIAM Conference on Computational Science and Engineering* 2017.
- Mojgani, R. and Balajewicz, M. “Lagrangian basis method for dimensionality reduction of convection dominated nonlinear flows”, *arXiv:1701.04343*.

# Outline

## 1 Introduction

- Fundamentals
- Motivation

## 2 Identifying an optimal manifold

- Physics-aware registration-based manifold

## 3 ROMs on the identified manifold

- Projection based ROMs
- Neural Network based ROMs

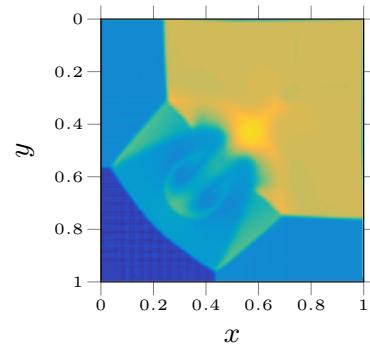
## 4 Stabilization of time-varying ROMs

- A feedback controller approach

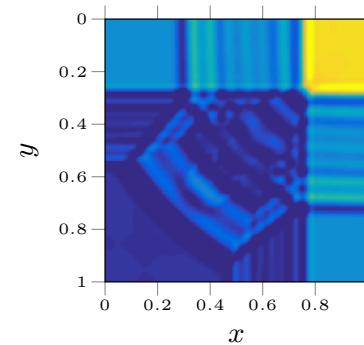
## 5 Summary

## Stability of ROMs

- A projection-based ROM does not inherit the stability characteristics of the high fidelity model



(a) High fidelity model,  $t = 0.80$



(b) ROM,  $t = 0.15$

Figure 20: Two-dimensional Riemann problem, configuration 3 [Lax and Liu, 1998]

## Stabilization of time-varying systems

- The stabilization problem

$$\begin{aligned}
 & \underset{\sigma_i[n]}{\text{minimize}} \quad \sum_{n=0}^{N_t} \|\mathbf{y}[n] - \hat{\mathbf{y}}_r[n]\|_2, \\
 & \text{subject to} \quad 1. \sigma_i^\Delta[n] \in \mathbb{R}^+, \text{ for } n \in \{0, \dots, N_t\}, \\
 & \quad 2. \sup_{n \geq n_0} \left\| \boldsymbol{\phi}_{\hat{\mathbf{A}}_r}[n, n_0] \right\|_2 \leq c[n_0],
 \end{aligned} \tag{27}$$

where  $\sigma_i^\Delta[n]$  are change in the singular values of reduced system matrix

- Mojgani, R. and Balajewicz, M. "Stabilization of linear time-varying reduced order models, a feedback controller approach". *International Journal for Numerical Methods in Engineering*, [Accepted]

# Outline

## 1 Introduction

- Fundamentals
- Motivation

## 2 Identifying an optimal manifold

- Physics-aware registration-based manifold

## 3 ROMs on the identified manifold

- Projection based ROMs
- Neural Network based ROMs

## 4 Stabilization of time-varying ROMs

- A feedback controller approach

## 5 Summary

## Summary

- No reasonable low-rank approximation exists for solutions featuring moving interfaces, discontinuities, or sharp gradients
- An interpretable manifold learning approach successfully reduces the dimensionality of convection dominated flows
- The identified low-rank manifold enables ROMs with predictive capabilities
- The non-intrusive ROMs constructed on the identified manifolds are time-varying dynamical systems
- The time-varying ROMs can be stabilized by a feedback controller and their predictive capabilities are preserved

## Summary

- No reasonable low-rank approximation exists for solutions featuring moving interfaces, discontinuities, or sharp gradients
- An interpretable manifold learning approach successfully reduces the dimensionality of convection dominated flows
- The identified low-rank manifold enables ROMs with predictive capabilities
- The non-intrusive ROMs constructed on the identified manifolds are time-varying dynamical systems
- The time-varying ROMs can be stabilized by a feedback controller and their predictive capabilities are preserved

## Summary

- No reasonable low-rank approximation exists for solutions featuring moving interfaces, discontinuities, or sharp gradients
- An interpretable manifold learning approach successfully reduces the dimensionality of convection dominated flows
- The identified low-rank manifold enables ROMs with predictive capabilities
- The non-intrusive ROMs constructed on the identified manifolds are time-varying dynamical systems
- The time-varying ROMs can be stabilized by a feedback controller and their predictive capabilities are preserved

## Summary

- No reasonable low-rank approximation exists for solutions featuring moving interfaces, discontinuities, or sharp gradients
- An interpretable manifold learning approach successfully reduces the dimensionality of convection dominated flows
- The identified low-rank manifold enables ROMs with predictive capabilities
- The non-intrusive ROMs constructed on the identified manifolds are time-varying dynamical systems
- The time-varying ROMs can be stabilized by a feedback controller and their predictive capabilities are preserved

## Summary

- No reasonable low-rank approximation exists for solutions featuring moving interfaces, discontinuities, or sharp gradients
- An interpretable manifold learning approach successfully reduces the dimensionality of convection dominated flows
- The identified low-rank manifold enables ROMs with predictive capabilities
- The non-intrusive ROMs constructed on the identified manifolds are time-varying dynamical systems
- The time-varying ROMs can be stabilized by a feedback controller and their predictive capabilities are preserved

## Summary

- No reasonable low-rank approximation exists for solutions featuring moving interfaces, discontinuities, or sharp gradients
- An interpretable manifold learning approach successfully reduces the dimensionality of convection dominated flows
- The identified low-rank manifold enables ROMs with predictive capabilities
- The non-intrusive ROMs constructed on the identified manifolds are time-varying dynamical systems
- The time-varying ROMs can be stabilized by a feedback controller and their predictive capabilities are preserved

This material is based upon work supported by

- Grainger College of Engineering, Computational Science and Engineering fellowship [2017-2018]
- The National Science Foundation under Grant No. CMMI-1435474
- The Air Force Office of Scientific Research under Grant No. FA9550-17-1-0203