DAEDALUS
RTG 2433

Nonlinear Model Reduction for an Advection-Reaction-Diffusion Equation with a Fisher term

Philipp Krah, Stefen Büchholz, Matthias Häringer, Julius Reiss

Absolventenseminar WS21/22,
03. February 2022

Contact: `krah@math.tu-berlin.de`
Follow: `philipp137.github.io`

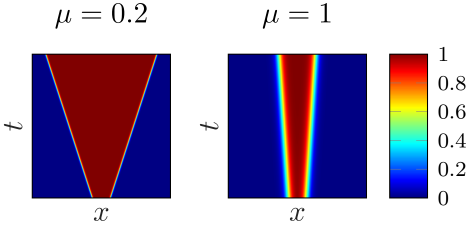
Advection-Reaction-Diffusion PDE

$$\partial_t q = -\boldsymbol{v} \cdot \boldsymbol{\nabla} q + \kappa \Delta q - \mu q^2 (q-1)$$

Advection-Reaction-Diffusion PDE

$$\partial_t q = -\boldsymbol{v} \cdot \boldsymbol{\nabla} q + \kappa \Delta q - \mu q^2 (q-1)$$

Discretize



FOM

$$\dot{\boldsymbol{q}}(t,\mu) = \boldsymbol{N}(\boldsymbol{q}, t, \mu)$$

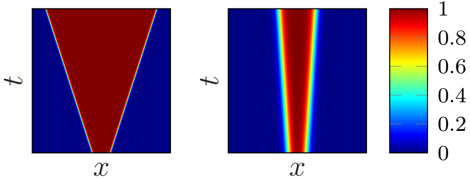
$$\boldsymbol{q}(t,\mu) \in \mathbb{R}^M, M \sim 10^6$$

Advection-Reaction-Diffusion PDE

$$\partial_t q = -\boldsymbol{v} \cdot \boldsymbol{\nabla} q + \kappa \Delta q - \mu q^2 (q - 1)$$

Discretize

$\mu = 0.2$ $\mu = 1$



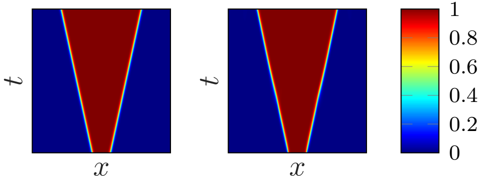
FOM

$$\dot{\boldsymbol{q}}(t, \mu) = \boldsymbol{N}(\boldsymbol{q}, t, \mu)$$

$$\boldsymbol{q}(t, \mu) \in \mathbb{R}^M, M \sim 10^6$$

Model Order Reduction $\boldsymbol{q} \approx \tilde{\boldsymbol{q}} = \boldsymbol{g}(\boldsymbol{a})$

FOM $\mu = 0.3$ ROM $\mu = 0.3$



ROM

$$\dot{\boldsymbol{a}}(t, \mu) = \hat{\boldsymbol{N}}(\boldsymbol{a}, t, \mu)$$

$$\boldsymbol{a}(t, \mu) \in \mathbb{R}^r, r \sim 10$$

POD-Galerkin Example: Advection Equation

Discretized PDE $q_t - vq_x = 0$ gives linear FOM:

$$\text{(FOM)} \quad \dot{\boldsymbol{q}}(t) = \boldsymbol{D}\boldsymbol{q}(t) \quad \boldsymbol{D} \in \mathbb{R}^{M\times M}, \boldsymbol{q} \in \mathbb{R}^{M}, \quad M \sim 10^6$$

POD-Galerkin Example: Advection Equation

Discretized PDE $q_t - vq_x = 0$ gives linear FOM:

$$\text{(FOM)} \quad \dot{\boldsymbol{q}}(t) = \boldsymbol{D}\boldsymbol{q}(t) \quad \boldsymbol{D} \in \mathbb{R}^{M\times M}, \boldsymbol{q} \in \mathbb{R}^{M}, \quad M \sim 10^6$$

1. Create low dimensional subspace with POD/SVD Truncated SVD of Snapshot matrix

$$\mathbf{Q} = [\boldsymbol{q}(t_1), \dots, \boldsymbol{q}(t_N)] \approx \mathbf{U}_r \boldsymbol{\Sigma}_r \mathbf{V}_r^T \quad \mathbf{U}_r^T \mathbf{U}_r = \mathbf{V}_r^T \mathbf{V}_r = \mathbf{I}_r$$

$$\rightarrow \tilde{\boldsymbol{q}}(t) = \boldsymbol{g}(\boldsymbol{a}) = \mathbf{U}_r \boldsymbol{a}(t)$$

POD-Galerkin Example: Advection Equation

Discretized PDE $q_t - v q_x = 0$ gives linear FOM:

$$\text{(FOM)} \quad \dot{\boldsymbol{q}}(t) = \boldsymbol{D}\boldsymbol{q}(t) \quad \boldsymbol{D} \in \mathbb{R}^{M\times M}, \boldsymbol{q} \in \mathbb{R}^{M}, \quad M \sim 10^6$$

1. Create low dimensional subspace with POD/SVD Truncated SVD of Snapshot matrix

$$\mathbf{Q} = [\boldsymbol{q}(t_1), \dots, \boldsymbol{q}(t_N)] \approx \mathbf{U}_r \boldsymbol{\Sigma}_r \mathbf{V}_r^T \quad \mathbf{U}_r^T \mathbf{U}_r = \mathbf{V}_r^T \mathbf{V}_r = \mathbf{I}_r$$

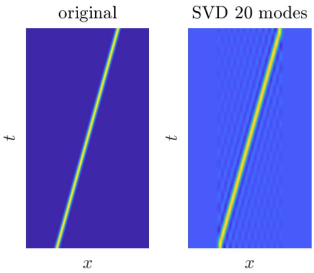
$$\rightarrow \tilde{\boldsymbol{q}}(t) = \boldsymbol{g}(\boldsymbol{a}) = \mathbf{U}_r \boldsymbol{a}(t)$$

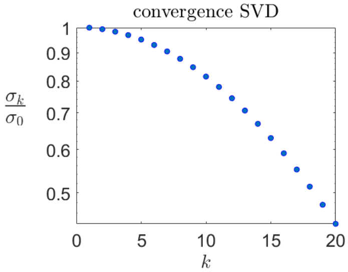
2. Plug in FOM and project onto $\mathbf{U}_r$

$$\mathbf{U}_r \dot{\boldsymbol{a}}(t) = \mathbf{D}\mathbf{U}_r \boldsymbol{a}(t) \Rightarrow \underbrace{\mathbf{U}_r^T \mathbf{U}_r}_{\mathbf{I}_r} \dot{\boldsymbol{a}}(t) = \underbrace{\mathbf{U}_r^T \mathbf{D} \mathbf{U}_r}_{\mathbf{D}_r} \boldsymbol{a}(t)$$

$$\text{(ROM)} \quad \dot{\boldsymbol{a}}(t) = \mathbf{D}_r \boldsymbol{a}(t) \quad \boldsymbol{D}_r \in \mathbb{R}^{r\times r}, \boldsymbol{a} \in \mathbb{R}^{r}, \quad r \sim 10$$

POD/SVD fails for transport dominated problems

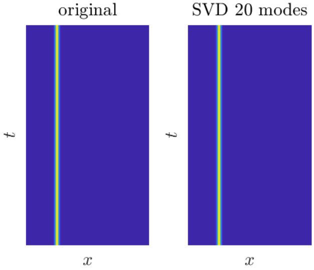


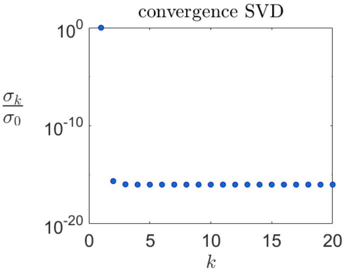


$$q(x,t) \approx \tilde{q}(x,t) = \mathbf{U}_r \boldsymbol{a}(t) = \sum_{l=1}^{r} u_l(x) a_l(t)$$

Problem: slow singular value decay σ_k

Idea: Transport compensation [Reiss et al. 2018]



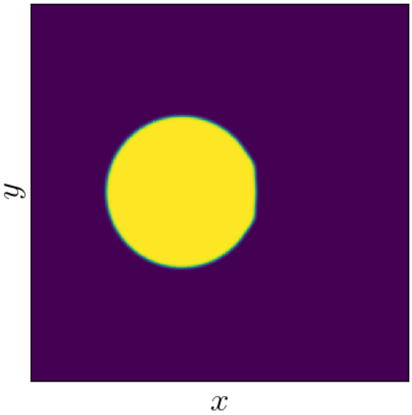


$$q(x+\Delta(t),t) \approx f(x) = \sum_{l=1}^{r} u_l(x)a_l(t)$$

$$\Rightarrow q(x,t) \approx T^{\Delta(t)} f(x) = f(x-\Delta(t))$$

Idea: apply time-dependent shift

Often no 1 to 1 mapping possible:

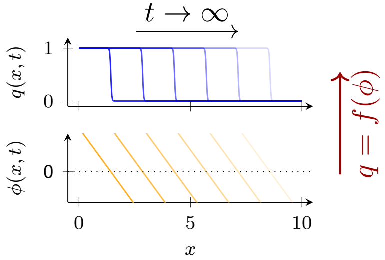


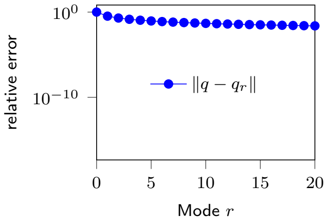
FTR - Front Transport Reduction

Idea: Front Transport Reduction (FTR)

[Krah, Sroka, and Reiss 2020]

$$q(x,t) \approx q_r(x,t) \coloneqq \sum_{k=1}^{r} u_k(x) a_k(t)$$



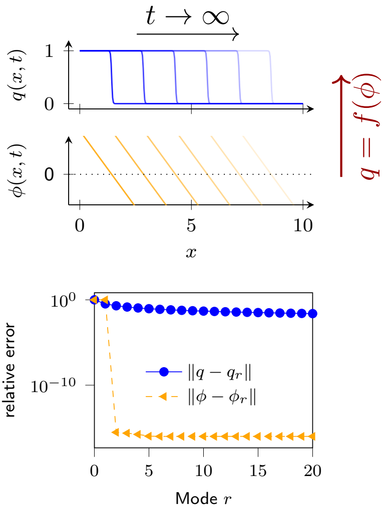


Idea: Front Transport Reduction (FTR)

[Krah, Sroka, and Reiss 2020]

$$q(x,t) \approx q_r(x,t) := \sum_{k=1}^{r} u_k(x) a_k(t)$$

$$\phi(x,t) \approx \phi_r(x,t) := \sum_{k=1}^{r} \psi_k(x) \tilde{a}_k(t)$$



Idea: Front Transport Reduction (FTR)

[Krah, Sroka, and Reiss 2020]

$$q(x,t) \approx q_r(x,t) := \sum_{k=1}^{r} u_k(x) a_k(t)$$

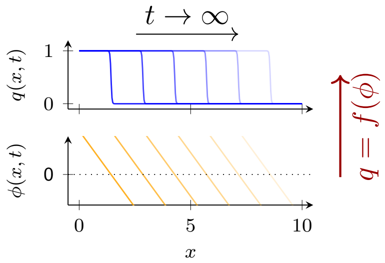
$$q(x,t) = f(\phi(x,t))$$

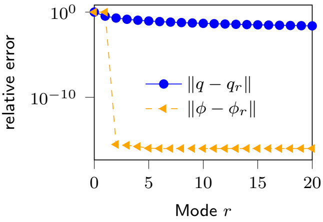
$$\phi(x,t) \approx \phi_r(x,t) := \sum_{k=1}^{r} \psi_k(x) \tilde{a}_k(t)$$

Transport compensation

$$\begin{aligned} q(x,t) &= T^{\Delta(t)}(f(x)) \\ &= f(x - \Delta(t)) \\ &= f(\phi(x,t)) \end{aligned}$$

$\Delta(t)$ front location





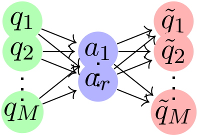
Front Transport Reduction For a given snapshot matrix $\mathbf{Q} \in \mathbb{R}^{M \times N_t}$ with $\mathbf{Q}_{ij} = q(\boldsymbol{x}_i, t_j) \in [0,1]$ and nonlinear smooth monotone increasing function $f\colon \mathbb{R} \to [0,1]$, find a rank r matrix $\mathbf{\Phi} \in \mathbb{R}^{M \times N_t}$, such that the error $\|\mathbf{Q} - \tilde{\mathbf{Q}}\|_{\mathrm{F}}^2$ for $\tilde{\mathbf{Q}}_{ij} = f(\mathbf{\Phi}_{ij})$ is minimized.

Algorithm 1 FTR as iterativ thresholding

Require: $\mathbf{Q} \in \mathbb{R}^{M \times N_t}$ data $\mathbf{Q}_{ij} = q(\boldsymbol{x}_i, t_j)$, τ step size, r rank,
front $f\colon \mathbb{R} \to [0,1]$

```
init Φ^k = 0
while not converged do
    residual R = f(Φ^k) − Q
    Φ^{k+1/2} = Φ^k − τR
    decompose and truncate
    Φ^{k+1} = svd(Φ^{k+1/2}, r)
    k ← k + 1
end while
return Φ^k
```

Algorithm 2 - FTR-NN Autoencoder



$$\tilde{\boldsymbol{q}} = g_{\mathsf{dec}}(g_{\mathsf{enc}}(\boldsymbol{q}))$$

Encoder $g_{\mathsf{enc}} : \mathbb{R}^M \to \mathbb{R}^r$, $\boldsymbol{q} \mapsto \boldsymbol{a} = g_{\mathsf{enc}}(\boldsymbol{q})$, mapping the input data $\boldsymbol{q}$ onto points $\boldsymbol{a}$ in a learned lower dimensional latent space and the

Decoder $g_{\mathsf{dec}} : \mathbb{R}^r \to \mathbb{R}^M, \boldsymbol{a} \mapsto g_{\mathsf{dec}}(\boldsymbol{a}) = \tilde{\boldsymbol{q}}$, mapping the latent representation back to input space.

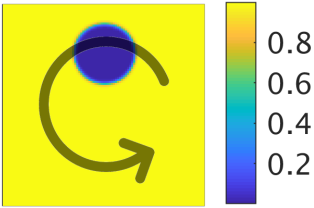
The task of the optimization procedure is now to determine $g_{\mathsf{dec}}, g_{\mathsf{enc}}$, such that the reconstruction error over the training data $\mathbf{Q} = [\boldsymbol{q}_1, \ldots, \boldsymbol{q}_{N_t}]$:

$$\mathcal{L}_{\mathrm{FTR}} = \|\mathbf{Q} - \tilde{\mathbf{Q}}\| = \sum_{i=1}^{N_t} \|\boldsymbol{q}_i - g_{\mathsf{dec}}(g_{\mathsf{enc}}(\boldsymbol{q}_i))\|_{\mathrm{F}}^2$$

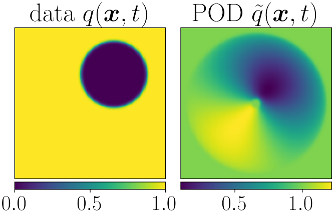
$$\tilde{\boldsymbol{q}} = g_{\mathsf{dec}}(\boldsymbol{a}) = f(\boldsymbol{\Psi}\boldsymbol{a}), \quad \boldsymbol{\Psi} \in \mathbb{R}^{M \times r},$$

2D Example - Moving Disc

Setting: disc of fixed radius R moves in a circle



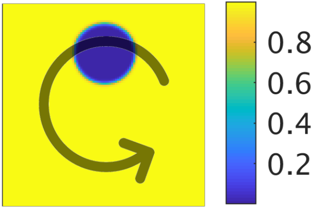
$$\begin{cases} q_t - \boldsymbol{v}(t) \cdot \nabla q = 0 & t \in [0,1] \\ \boldsymbol{v}(t) = \begin{pmatrix} -\sin(2\pi t) \\ \cos(2\pi t) \end{pmatrix} & \end{cases}$$



2D Example - Moving Disc

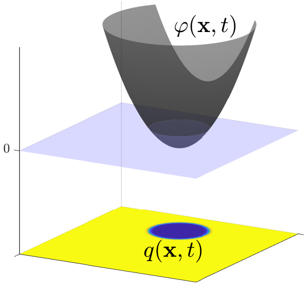
Setting: disc of fixed radius R moves in a circle

$$\begin{cases} q_t - \boldsymbol{v}(t) \cdot \nabla q = 0 & t \in [0,1] \\ \boldsymbol{v}(t) = \begin{pmatrix} -\sin(2\pi t) \\ \cos(2\pi t) \end{pmatrix} \end{cases}$$



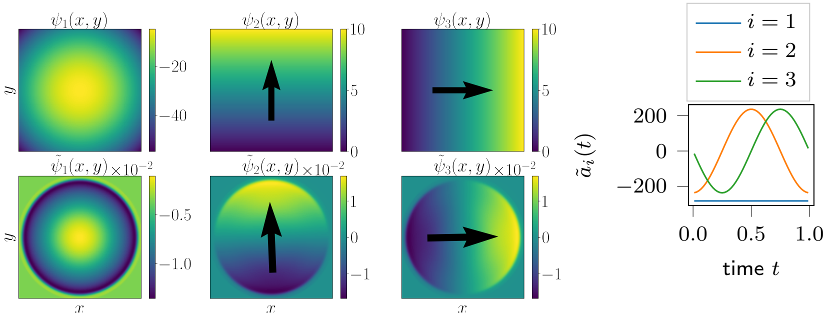
FTR: $$q(\boldsymbol{x},t) = f(\phi(\boldsymbol{x},t))$$
$$f(x) = \text{sigmoid}(x)$$

$$\phi(\boldsymbol{x},t) = \|\boldsymbol{x} - \boldsymbol{x}_0(t)\|_2^2 - R^2$$

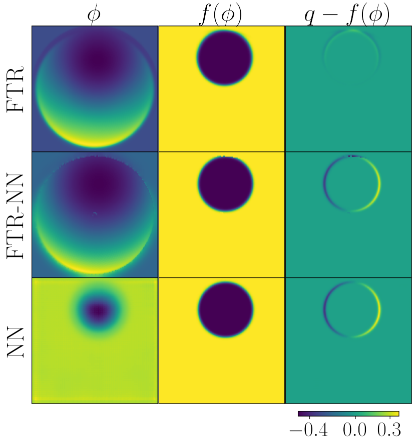


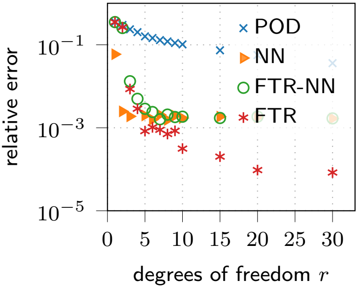
Moving Disc

$$q(\boldsymbol{x}, t) = f(\phi(\boldsymbol{x}, t))$$
$$\phi(\boldsymbol{x}, t) = \|\boldsymbol{x} - \boldsymbol{x}_0(t)\|_2^2 - R^2$$
$$= a_1(t)\psi_1(x, y) + a_2(t)\psi_2(x, y) + a_3(t)\psi_3(x, y)$$

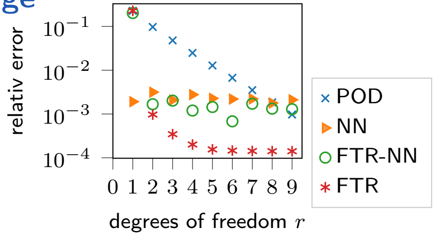


Moving Disc - FTR vs. Neural Autoencoder Networks





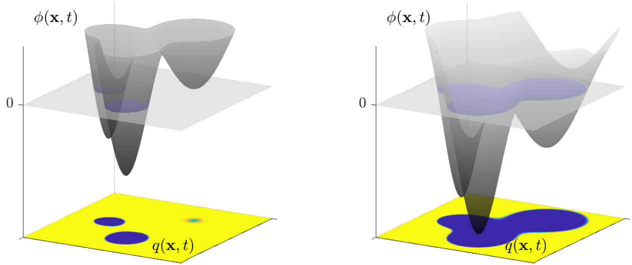
Advection with topology change



$q(\boldsymbol{x},t) = f(\phi(\boldsymbol{x},t))$ build from the level-set field

$$\varphi(\boldsymbol{x},t) = \sum_{k=1}^{3} -A_k e^{-\sigma_k r_k} - t$$

$$r_k = \|\boldsymbol{x} - \boldsymbol{x}_i\|_2 \,,$$



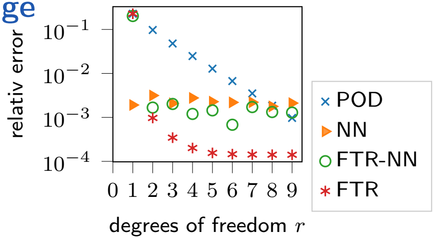
data $t = 0$ **data** $t = 0.4$

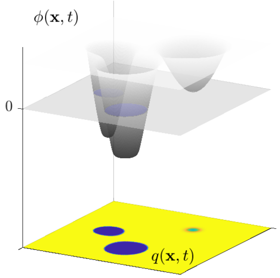
Advection with topology change

$q(\boldsymbol{x},t) = f(\phi(\boldsymbol{x},t))$ build from the level-set field

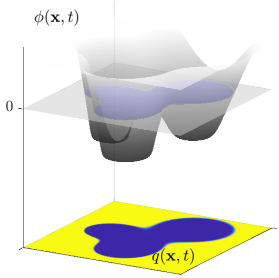
$$\varphi(\boldsymbol{x},t) = \sum_{k=1}^{3} -A_k e^{-\sigma_k r_k} - t$$

$$r_k = \|\boldsymbol{x} - \boldsymbol{x}_i\|_2 \,,$$





FTR $t = 0$, rank $r = 2$



FTR $t = 0.4$, rank $r = 2$

Nonlinear ROM - online prediction

How to generate a reduced order model from this?

$$\text{(FOM)} \quad \begin{cases} \dot{\boldsymbol{q}}(t,\mu) = \boldsymbol{N}(\boldsymbol{q},t,\mu) \\ \boldsymbol{q}(0,\mu) = \boldsymbol{q}_0(\mu) \end{cases} \longrightarrow \text{(ROM) ???}$$

How to generate a reduced order model from this?

$$\text{(FOM)} \quad \begin{cases} \dot{\boldsymbol{q}}(t,\mu) = \boldsymbol{N}(\boldsymbol{q},t,\mu) \\ \boldsymbol{q}(0,\mu) = \boldsymbol{q}_0(\mu) \end{cases} \longrightarrow \text{(ROM) ???}$$

Manifold Galerkin Projections [Lee,Carlberg,2020]

(*Project-then-discretize-approach*)

- Plug $\tilde{\boldsymbol{q}} = \boldsymbol{g}(\boldsymbol{a})$ into FOM
- Time derivative becomes

$$\dot{\boldsymbol{q}}(t,\mu) \approx \frac{\mathrm{d}}{\mathrm{d}t} g(\boldsymbol{a}(t,\mu)) = \boldsymbol{J}_g(\boldsymbol{a})\dot{\boldsymbol{a}}(t,\mu)$$

- Reduced order model

$$\text{(ROM)} \quad \begin{cases} \dot{\boldsymbol{a}}(t,\mu) = \boldsymbol{J}_g(\boldsymbol{a})^+ \boldsymbol{N}(g(\boldsymbol{a}),t,\mu) \\ \boldsymbol{a}(0,\mu) = \boldsymbol{a}_0(\mu) \end{cases}$$

Solution of the time continuous residual minimization:

$$\dot{\tilde{\boldsymbol{q}}}(t,\mu) = \underset{\dot{\boldsymbol{q}} \in \mathcal{T}\tilde{\boldsymbol{q}}(t,\mu)}{\operatorname{argmin}} \|\boldsymbol{r}(\boldsymbol{q},\dot{\boldsymbol{q}},t,\mu)\|_2^2 \quad \text{with} \quad \boldsymbol{r}(\boldsymbol{q},\dot{\boldsymbol{q}},t,\mu) \coloneqq \dot{\boldsymbol{q}} - \boldsymbol{N}(\boldsymbol{q},t,\mu)$$

Advection PDE: $\partial_t q - v\partial_x q = 0$

$$\text{(FOM)} \quad \begin{cases} \dot{\boldsymbol{q}}(t,\mu) = \mathbf{L}\boldsymbol{q} \\ \boldsymbol{q}(0) = \boldsymbol{q}_0 \,. \end{cases}$$

POD

Linear Subspace:

$$\tilde{\boldsymbol{q}}(t) = \mathbf{U}\boldsymbol{a}(t)$$

⇒

FTR

Manifold:

$$\tilde{\boldsymbol{q}}(t) = f(\boldsymbol{\Psi}\boldsymbol{a}(t))$$

Galerkin

$$\dot{\boldsymbol{a}}(t) = \mathbf{U}^T\mathbf{L}\mathbf{U}\boldsymbol{a}$$

⇒

Manifold-Galerkin

$$\dot{\boldsymbol{a}}(t) = \mathbf{J}(\boldsymbol{a})^{+}\mathbf{L}f(\boldsymbol{\Psi}\boldsymbol{a})$$

Level-set field features same transport as q!

- Since for the FTR $q = f(\phi)$

$$\partial_t q - v\partial_x q = f'(\phi)(\partial_t \phi - v\partial_x \phi) = 0 \rightsquigarrow \partial_t \phi - v\partial_x \phi = 0$$

- Solve ODE for $\boldsymbol{\phi}$ instead of $\boldsymbol{q}$

$$\boldsymbol{q} = \mathbf{L}\boldsymbol{q} \rightsquigarrow \boldsymbol{\phi} = \mathbf{L}\boldsymbol{\phi}$$

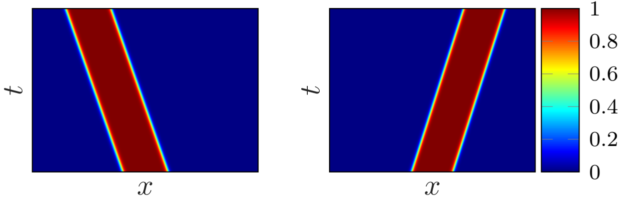
- Solve ROM for $\boldsymbol{\phi} = \boldsymbol{\Psi}\boldsymbol{a}$ instead of $\boldsymbol{q} = \mathbf{U}\boldsymbol{a}$

$$\dot{\boldsymbol{a}}(t) = \mathbf{U}^T\mathbf{L}\mathbf{U}\boldsymbol{a} \rightsquigarrow \dot{\boldsymbol{a}}(t) = \boldsymbol{\Psi}^T\mathbf{L}\boldsymbol{\Psi}\boldsymbol{a}$$

training

$$\partial_t q + u(t)\partial_x q = 0$$

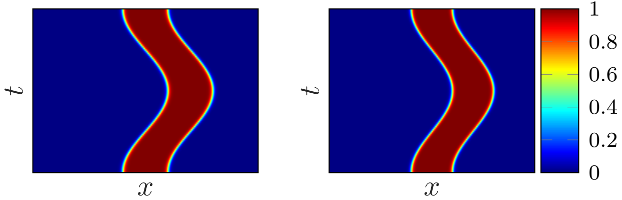
$u(t) = -2$ $u(t) = 2$



testing

$$\partial_t q + 5\sin(2\pi t/T)\partial_x q = 0$$

FOM **FTR - ROM**



ARD PDE: $\partial_t q(\boldsymbol{x}, t, \mu) = -\boldsymbol{v} \cdot \boldsymbol{\nabla} q + \kappa \Delta q - \mu q^2 (q-1)$

$$\text{(FOM)} \quad \begin{cases} \dot{\boldsymbol{q}}(t, \mu) = \boldsymbol{N}(\boldsymbol{q}, t, \mu) \\ \boldsymbol{q}(0) = \boldsymbol{q}_0 \,. \end{cases}$$

POD

Linear Subspace:

$$\tilde{\boldsymbol{q}}(t) = \mathbf{U}\boldsymbol{a}(t)$$

⇒

FTR

Manifold:

$$\tilde{\boldsymbol{q}}(t) = f(\boldsymbol{\Psi}\boldsymbol{a}(t))$$

Galerkin

$$\dot{\boldsymbol{a}}(t) = \mathbf{U}^T \boldsymbol{N}(\mathbf{U}\boldsymbol{a}, t, \mu)$$

⇒

Manifold-Galerkin

$$\dot{\boldsymbol{a}}(t) = \mathbf{J}(\boldsymbol{a})^+ \boldsymbol{N}(f(\boldsymbol{\Psi}\boldsymbol{a}), t, \mu)$$

ARD PDE: $\partial_t q(\boldsymbol{x}, t, \mu) = -\boldsymbol{v} \cdot \boldsymbol{\nabla} q + \kappa \Delta q - \mu q^2 (q-1)$

$$\text{(FOM)} \quad \begin{cases} \dot{\boldsymbol{q}}(t,\mu) = \boldsymbol{N}(\boldsymbol{q}, t, \mu) \\ \boldsymbol{q}(0) = \boldsymbol{q}_0 \,. \end{cases}$$

POD

Linear Subspace:

$\tilde{\boldsymbol{q}}(t) = \mathbf{U}\boldsymbol{a}(t)$

⇒

FTR

Manifold:

$\tilde{\boldsymbol{q}}(t) = f(\boldsymbol{\Psi}\boldsymbol{a}(t))$

Galerkin

$\dot{\boldsymbol{a}}(t) = \mathbf{U}^T \boldsymbol{N}(\mathbf{U}\boldsymbol{a}, t, \mu)$

⇒

Manifold-Galerkin

$\dot{\boldsymbol{a}}(t) = \mathbf{J}(\boldsymbol{a})^+ \boldsymbol{N}(f(\boldsymbol{\Psi}\boldsymbol{a}), t, \mu)$

!!Hyperreduction!!

How to generate a reduced order model from this?

$$\text{(FOM)} \quad \begin{cases} \dot{\boldsymbol{q}}(t,\mu) = \boldsymbol{N}(\boldsymbol{q},t,\mu) \\ \boldsymbol{q}(0,\mu) = \boldsymbol{q}_0(\mu) \end{cases} \longrightarrow \text{(ROM) ???}$$

Manifold Galerkin Projections [Lee,Carlberg,2020]
(*Project-then-discretize-approach*)

- Plug $\tilde{\boldsymbol{q}} = \boldsymbol{g}(\boldsymbol{a})$ into FOM
- Time derivative becomes

$$\dot{\boldsymbol{q}}(t,\mu) \approx \frac{\mathrm{d}}{\mathrm{d}t} g(\boldsymbol{a}(t,\mu)) = \boldsymbol{J}_g(\boldsymbol{a})\dot{\boldsymbol{a}}(t,\mu)$$

- Reduced order model $\mathbf{P} \in \mathbb{R}^{p\times M}, r \leq p \ll M$

$$\text{(hyperreduced-ROM)} \quad \begin{cases} \dot{\boldsymbol{a}}(t,\mu) = [\mathbf{P}\boldsymbol{J}_g(\boldsymbol{a})]^+\mathbf{P}\boldsymbol{N}(g(\boldsymbol{a}),t,\mu) \\ \boldsymbol{a}(0,\mu) = \boldsymbol{a}_0(\mu) \end{cases}$$

Solution of the time continuous residual minimization:

$$\dot{\tilde{\boldsymbol{q}}}(t,\mu) = \underset{\dot{\boldsymbol{q}}\in\mathcal{T}\tilde{\boldsymbol{q}}(t,\mu)}{\operatorname{argmin}} \|\boldsymbol{r}(\boldsymbol{q},\dot{\boldsymbol{q}},t,\mu)\|^2_{\mathbf{P}^2} \quad \text{with} \quad \boldsymbol{r}(\boldsymbol{q},\dot{\boldsymbol{q}},t,\mu) := \dot{\boldsymbol{q}} - \boldsymbol{N}(\boldsymbol{q},t,\mu)$$

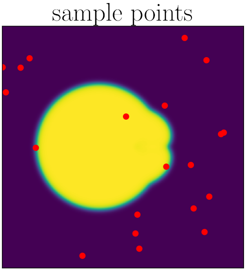
Hyperreduction

Preselection of sample points via gappy POD [Carlberg et al. 2013; Kim et al. 2021]:

$$\dot{\tilde{\boldsymbol{q}}}(t) = \underset{\dot{\boldsymbol{q}} \in \mathcal{T}\tilde{\boldsymbol{q}}(t,\mu)}{\operatorname{argmin}} \|\boldsymbol{r}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t, \mu)\|^2_{\mathsf{P}^2} \quad \text{with} \quad \boldsymbol{r}(\boldsymbol{q}, \dot{\boldsymbol{q}}, t, \mu) \coloneqq \dot{\boldsymbol{q}} - \boldsymbol{N}(\boldsymbol{q}, t, \mu)$$

(hyperreduced-ROM)
$$\begin{cases} \mathsf{P}\boldsymbol{J}(\boldsymbol{a})\dot{\boldsymbol{a}}(t,\mu) = \mathsf{P}\boldsymbol{N}(f(\boldsymbol{a}), t, \mu) \\ \boldsymbol{a}(0,\mu) = \boldsymbol{a}_0(\mu) \end{cases}$$

$$\mathsf{P} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 1 & 0 & \cdots & 0 \\ \vdots & & & & & \vdots \\ 0 & \cdots & 0 & 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{p \times M}$$

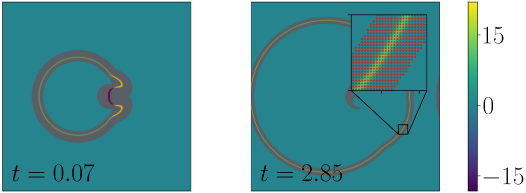


Hyperreduction

Adaptive selection of sample points

$$\text{(hyperreduced-ROM)} \quad \begin{cases} \mathsf{P}_a \boldsymbol{J}(\boldsymbol{a})\dot{\boldsymbol{a}}(t,\mu) = \mathsf{P}_a \boldsymbol{N}(f(\boldsymbol{a}), t, \mu) \\ \boldsymbol{a}(0,\mu) = \boldsymbol{a}_0(\mu) \end{cases}$$

$\mathbf{P}_a$ selects the p first grid points $\{i_1, ..., i_p\}$ close to the reacting front.



Take p smallest values of the level-set field: $\boldsymbol{\phi} = \boldsymbol{\Psi}\boldsymbol{a} \in \mathbb{R}^M$

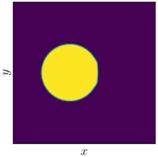
$$\left|\phi_{i_1}\right| \leq \left|\phi_{i_2}\right| \leq \cdots \leq \left|\phi_{i_p}\right| \leq \left|\phi_{i_{p+k}}\right| \quad \forall k = 1, \ldots, M-p$$

Advection-diffusion-reaction PDE (+ Periodic BC)

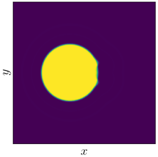
$$\begin{cases} \partial_t q(\boldsymbol{x}, t, \mu) & = -\boldsymbol{v} \cdot \boldsymbol{\nabla} q + \kappa \Delta q - \mu q^2 (q-1) \\ q(\boldsymbol{x}, 0, \mu) & = q_0(\boldsymbol{x}) \end{cases}$$

with velocity field $\boldsymbol{v}(\boldsymbol{x}, t)$. Tuned to include topology change.

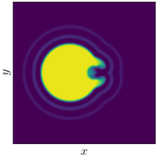
FOM
DoFs $= 512^2$:



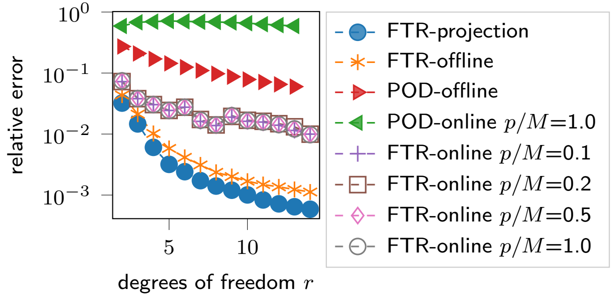
FTR-ROM
DoFs: $= 6$

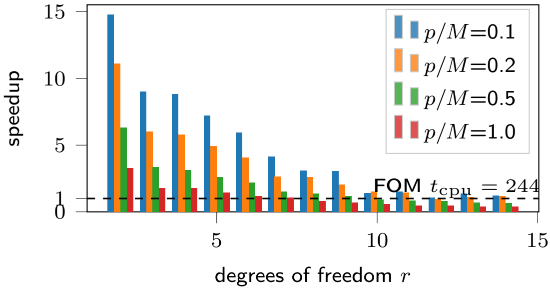


POD-ROM
DoFs: $= 6$



Error vs Speedup





Summary - Front Transport Reduction

- ϕ low rank even if $q = f(\phi)$ is not
- embeds 1D fronts in 2D, 3D
- handles topology changes
- is similar to one layer neural net
- resulting basis is well suited for dynamical models

Thank you for your attention!

References:

- Carlberg, K. et al. (2013). "The GNAT method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows". In: *Journal of Computational Physics* 242, pp. 623–647.
- Kim, Y. et al. (2021). "A fast and accurate physics-informed neural network reduced order model with shallow masked autoencoder". In: *Journal of Computational Physics*, p. 110841.
- Krah, P., M. Sroka, and J. Reiss (2020). "Model Order Reduction of Combustion Processes with Complex Front Dynamics". In: *ENUMATH 2019*, 803–811.
- Lee, K. and K. T. Carlberg (2020). "Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders". In: *Journal of Computational Physics* 404, p. 108973. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2019.108973. URL: http://www.sciencedirect.com/science/article/pii/S0021999119306783.

2. Method Least Squares Petrov Galerkin Projections [Lee, Carlberg,2020] (*discretize-then-project-Approach*)

- FOM is first discretized in time: $\boldsymbol{q}^{n+1} = \boldsymbol{q}^n + \tau \boldsymbol{N}(\boldsymbol{q}^{n+1}, t^{n+1})$
- The minimization problem for the discretized residual $\boldsymbol{r}^n$ of the implicit Euler is

$$\boldsymbol{a}^{n+1} = \operatorname*{argmin}_{a\in\mathbb{R}^r} \|\boldsymbol{r}^n(\mathsf{f}(\boldsymbol{a}))\|_2^2 \quad \text{with} \quad \boldsymbol{r}^n(\boldsymbol{\xi}) := \boldsymbol{\xi} - \boldsymbol{q}^n - \tau \boldsymbol{N}(\boldsymbol{\xi}, t^{n+1})\,.$$

- Necessary condition $\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{a}}\|\boldsymbol{r}^n\|_2^2 = 2\langle \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{a}}\boldsymbol{r}^n, \boldsymbol{r}^n\rangle = 0$ results in

$$\text{(ROM)} \quad \begin{cases} \Psi^n(\boldsymbol{a}^{n+1})^T \boldsymbol{r}^n(\mathsf{f}(\boldsymbol{a}^{n+1})) = 0 \\ \boldsymbol{a}(0) = \boldsymbol{a}_0 \end{cases}$$

where

$$\Psi^n(\boldsymbol{a}) := \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{a}} \boldsymbol{r}^n(\mathsf{f}(\boldsymbol{a})) = \left[\mathbf{I}_M - \tau \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{q}} \boldsymbol{N}(\boldsymbol{q}, t^{n+1})\right]_{q=\mathsf{f}(a)} \mathbf{J}(\boldsymbol{a})\,.$$

2. **Method** Least Squares Petrov Galerkin Projections [Lee, Carlberg,2020] (*discretize-then-project-Approach*)

- FOM is first discretized in time: $\boldsymbol{q}^{n+1} = \boldsymbol{q}^n + \tau \boldsymbol{N}(\boldsymbol{q}^{n+1}, t^{n+1})$
- The minimization problem for the discretized residual $\boldsymbol{r}^n$ of the implicit Euler is

$$\boldsymbol{a}^{n+1} = \underset{a \in \mathbb{R}^r}{\operatorname{argmin}} \|\boldsymbol{r}^n(\mathbf{f}(\boldsymbol{a}))\|^2_{\mathbf{P}^2} \quad \text{with} \quad \boldsymbol{r}^n(\boldsymbol{\xi}) := \boldsymbol{\xi} - \boldsymbol{q}^n - \tau \boldsymbol{N}(\boldsymbol{\xi}, t^{n+1}) \,.$$

- Necessary condition $\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{a}} \|\boldsymbol{r}^n\|^2_{\mathbf{P}^2} = 2\langle \mathbf{P} \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{a}} \boldsymbol{r}^n, \mathbf{P}\boldsymbol{r}^n \rangle = 0$ results in

$$(\text{hyperreduced-ROM}) \quad \begin{cases} \Psi^n(\boldsymbol{a}^{n+1})^T \mathbf{P} \boldsymbol{r}^n(\mathbf{f}(\boldsymbol{a}^{n+1})) = 0 \\ \boldsymbol{a}(0) = \boldsymbol{a}_0 \end{cases}$$

where

$$\Psi^n(\boldsymbol{a}) := \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{a}} \boldsymbol{r}^n(\mathbf{f}(\boldsymbol{a})) = \mathbf{P} \left[\mathbf{I}_M - \tau \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{q}} \boldsymbol{N}(\boldsymbol{q}, t^{n+1}) \right]_{q = \mathbf{f}(a)} \mathbf{J}(\boldsymbol{a}) \,.$$