Tracking Distributed Parameters System Dynamics with Recursive Dynamic Mode Decomposition with control

Authors:

Taleb BOU HAMDAN

Guillaume Mercère

Patrick Coirault

Thibault Dairay

Raphael Meunier

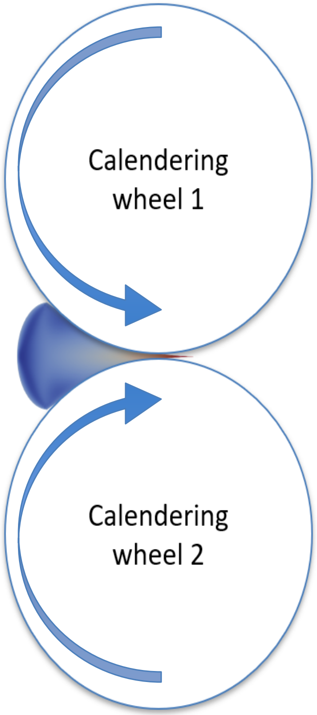
Pascal Tremblay

Calendering Process

Calendering process:

- Calendered material pressed between wheels
- 2 calendering wheels are turning at a varying speed
- Nonlinear thermo-mechanical coupling:
  - $\boldsymbol{Temperature} = f(velocity, \boldsymbol{viscosity})$
  - $\boldsymbol{viscosity} = g(velocity, \boldsymbol{temperature})$

Aim: <u>Low-dimensional representative model</u>



System Identification

- Our system dynamics are governed by nonlinear Partial Differential Equations (PDEs)

➢ Solution: Data driven modeling along with model order reduction

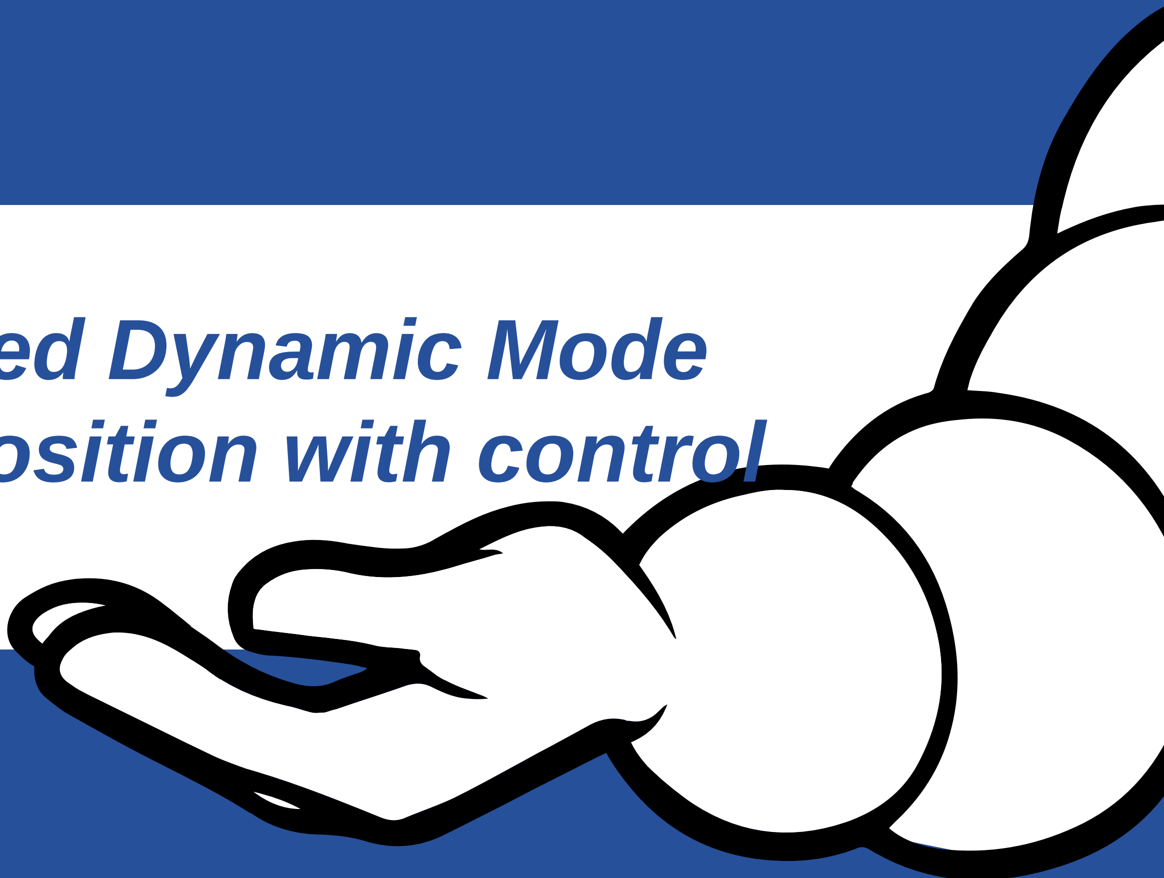
- Discrete-Time system model:

$$X(k+1) = AX(k) + BQ(k)$$

- Introduction to reduced Dynamic Mode Decomposition with control:
  - Dynamic Mode Decomposition with control
  - Proper Orthogonal Decomposition
- Recursive Proper Orthogonal Decomposition
- Recursive reduced Dynamic Mode Decomposition with control
- Conclusion and Prospects

MICHELIN

Reduced Dynamic Mode Decomposition with control



Dynamic Mode Decomposition (1/2)

DMD: Dynamic Mode Decomposition

Nonlinear model:

$$\frac{d\widetilde{\boldsymbol{X}}}{dt} = N(\widetilde{\boldsymbol{X}})$$

Koopman theory* states that any nonlinear system can be formulated via an infinite dimensional linear model

$$\frac{dX}{dt} = L(X)$$

Euler discretization:

$$X(k+1) = AX(k)$$

*: Rowley, C.W., Mezi´c, I., Bagheri, S., Schlatter, P., Henningson, D.S., 2009. Spectralanalysis of nonlinear flows. J. Fluid Mech. 641, 115–127

Dynamic Mode Decomposition (2/2)

- Consider the snapshot matrix of the states over N_t instants,
$$X_{snap} = [X(1), X(2), \dots, X(N_t)]$$

- Split X_{snap} into $X1$ and $X2$ such that:
$$X1 = [X(1), \dots, X(N_{t-1})] \; and \; X2 = [X(2), \dots, X(N_t)]$$

- $X2$ is a one step ahead version of $X1$:
$$X2 = AX1$$

- A can be estimated using the pseudo inverse of $X1$, $X1^+$ (*)
$$A = X2\, X1^+$$

- $X1^+$ can be found using Singular Value Decomposition of $X1$

*: Schmid, P.J., 2010. Dynamic mode decomposition of numerical and experimental data. J. Fluid Mech. 656, 5–28.

Dynamic Mode Decomposition with control

DMDc: Dynamic Mode Decomposition with control

- Input $Q(k) \neq 0$;

System model:

$$X(k+1) = AX(k) + BQ(k)$$

Splitting snapshot matrix X_{snap} into $X1$ and $X2$ and saving input matrix Q:

$$X2 = AX1 + BQ$$

$$X2 = (A \;\; B)\begin{pmatrix} X1 \\ Q \end{pmatrix}$$

As before, $(A\ B)$ can be estimated using the pseudo inverse of $\begin{pmatrix} X1 \\ Q \end{pmatrix}$:

$$(A \;\; B) = X2 \begin{pmatrix} X1 \\ Q \end{pmatrix}^{+}$$

Toy Example

Calendering Toy Example: (No Convection)

Heat diffusion (simple) PDE:

$$\frac{\partial T}{\partial t} = \mu \frac{\partial^2 T}{\partial y^2} + q_{source} \quad for\ y \in]0; L_y[\ and\ t \in [0; t_{end}]$$

$$T(y, t = 0) = 373.15\ K \text{ (Initial Condition)}$$

$$T(y = 0, t) = 323.15\ K \text{ (Boundary Condition)}$$

$$\frac{\partial T}{\partial y}(y = L_y, t) = 0$$

$\mu = cst$

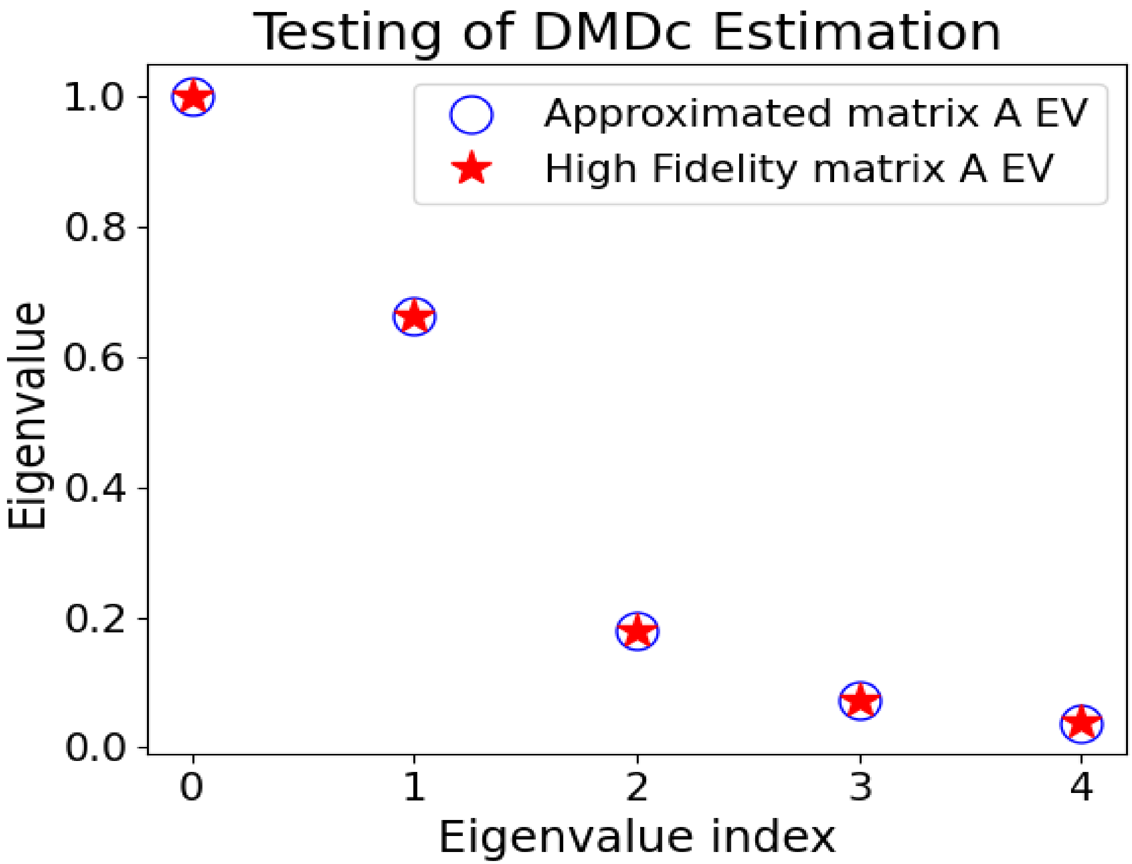
The state vector X at each instant represents the temperature values along the length L_y, N_x values. So, we will stick for the nomenclature $X(y_i, t_k)$ instead of $T(y_i, t_k)$

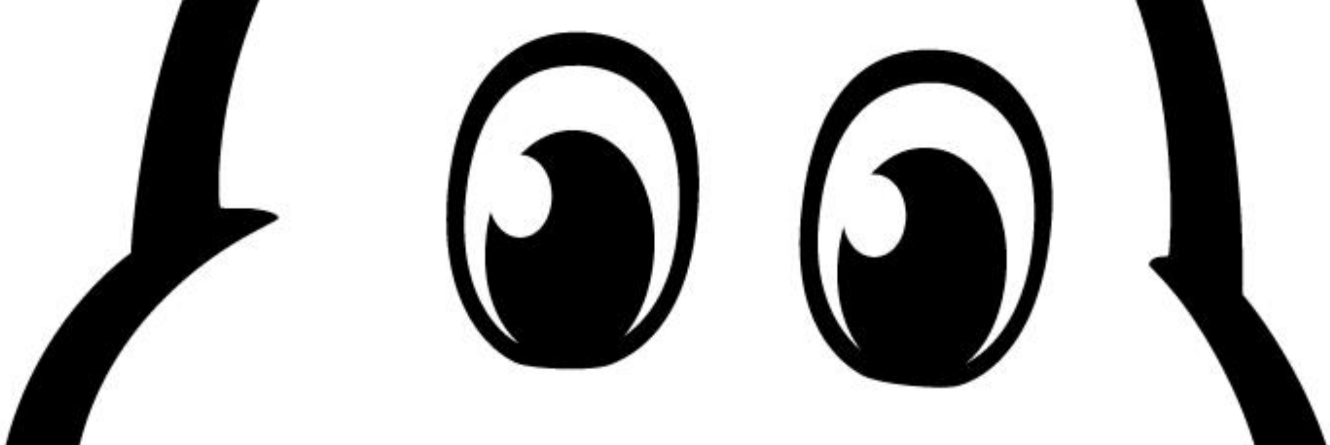
- Using 2nd order centered finite differences and 1st order implicit Euler, we get a linear High Fidelity system model of order N_x

Based on selected values of q_{source}, X_{snap} is collected

DMDc Validation

- Validation of the DMDc, linear approximated model





- Introduction to reduced Dynamic Mode Decomposition with control:
  - Dynamic Mode Decomposition with control
  - Proper Orthogonal Decomposition
- Recursive Proper Orthogonal Decomposition
- Recursive reduced Dynamic Mode Decomposition with control
- Conclusion and Prospects

MICHELIN

Model Reduction: Proper Orthogonal Decomposition

Identified model order: $(N_x \sim 10^2 - 10^3)$

Find matrix $\boldsymbol{R} \in \mathbb{R}^{N_x \times N_t}$ with a rank $r < K = \min(N_x, N_t)$ minimizing the Frobenius norm of the error $\boldsymbol{X_{snap}} - \boldsymbol{R}$:

$$\min_{rank(R) \leq \boldsymbol{r}} \left\| \boldsymbol{X_{snap}} - \boldsymbol{R} \right\|_F$$

<u>Using Eckart-Young theorem*</u>:

$$\min_{rank(\boldsymbol{R}) \leq \boldsymbol{r}} \left\| \boldsymbol{X_{snap}} - \boldsymbol{R} \right\|_F = \left\| \boldsymbol{X_{snap}} - \widehat{\boldsymbol{X}} \right\|_F$$

where $\hat{X}$ is the **truncated $\boldsymbol{X_{snap}}$ of order $\boldsymbol{r}$**

Using SVD of X_{snap},

$$X_{snap} = U\Sigma V^T = [U_r \; U_e] \begin{bmatrix} \Sigma_r & 0 \\ 0 & \Sigma_e \end{bmatrix} [V_r^T \; V_e^T]^T$$

$$\hat{X} = U_r \Sigma_r V_r^T$$

*: G. Berkooz and P.Holmes, The Proper Orthogonal Decomposition in the analysis of turbulent flows, Annual Review of Fluid Mechanics

MICHELIN

Reduced DMDc

Process model of high order from DMDc: $X(k+1) = AX(k) + BQ(k)$

Step 1: Use again the Singular Value Decomposition of the matrix X_{snap}

$$X_{snap} = U\Sigma \mathrm{V}^{\mathrm{T}}$$

Step 2: Keep the first r singular values such that ($\sigma_j > \epsilon\sigma_1 \; and \; j \in [1, r]$)

The first r vectors of the matrix U are the POD modes saved in the matrix $\boldsymbol{U_r}$, truncated matrix of $\boldsymbol{U}$

Step 3: The reduced order vector of $X(k)$ is $X_r(k) = U_r^T X(k)$

Step 4: The system model becomes:

$$U_r X_r(k+1) = AU_r X_r(k) + BQ(k)$$

But U_r^T is orthogonal;

$$X_r(k+1) = U_r^T A U_r X_r(k) + U_r^T BQ(k)$$

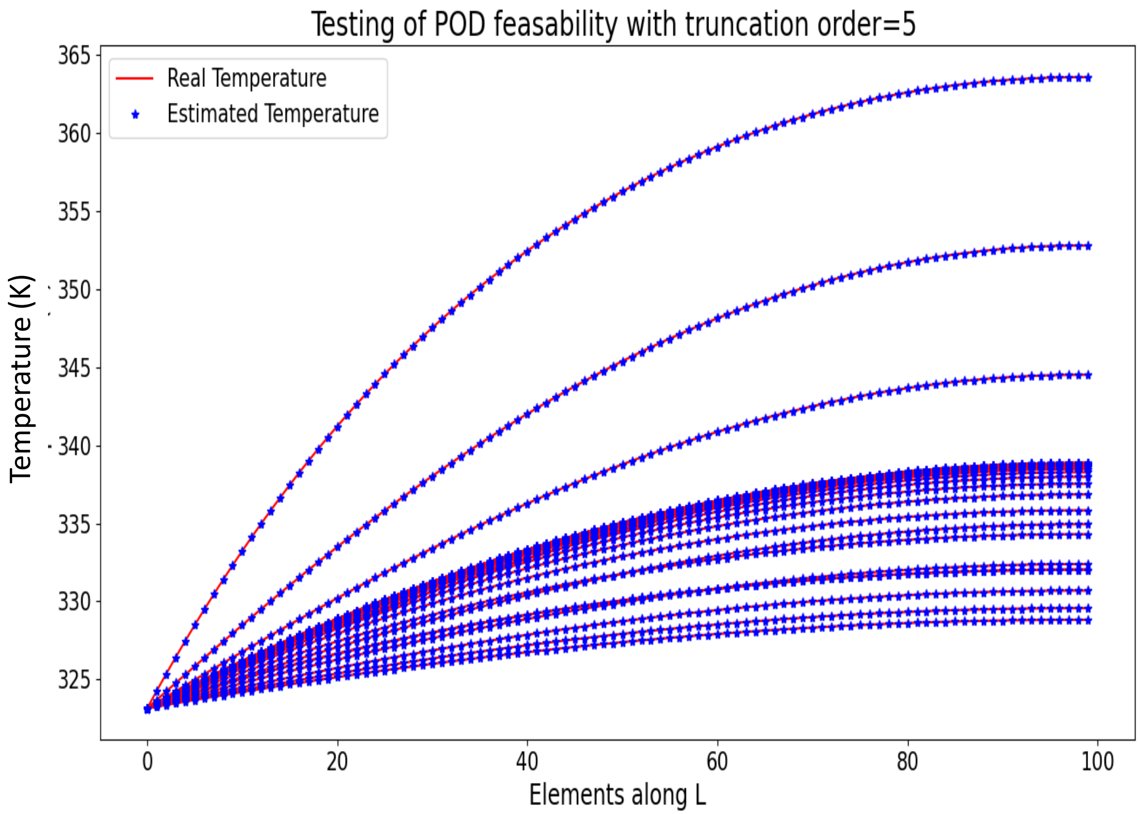
$$X_r(k+1) = A_r X_r(k) + B_r Q(k)$$

Then reduced order A is $A_r = U_r^T A U_r$ of order $(r * r)$

And the reduced order B is $B_r = U_r^T B$ of order $(r * n_Q)$

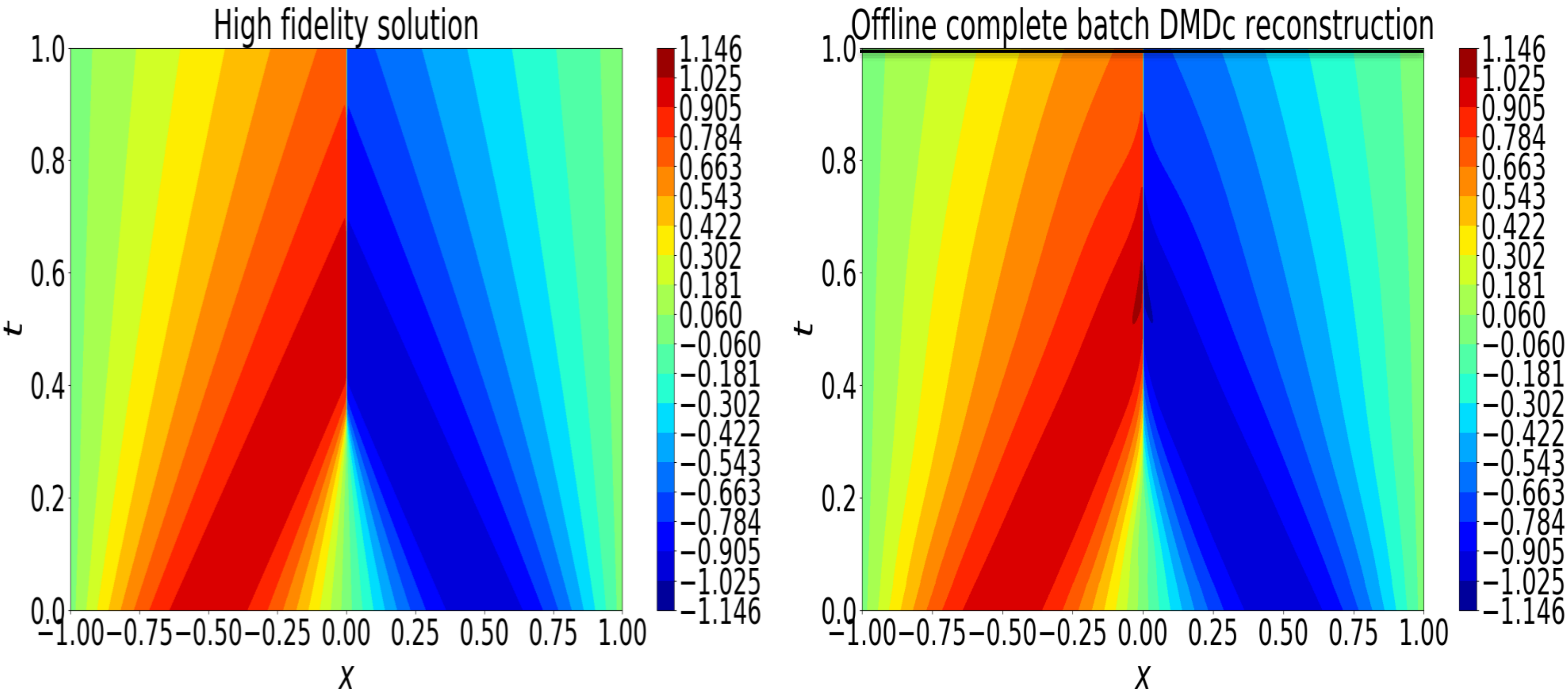
POD Reduction and Expansion Results

The resulting linear model of reduced order from N_x=100 to $r = 5$



Nonlinear example

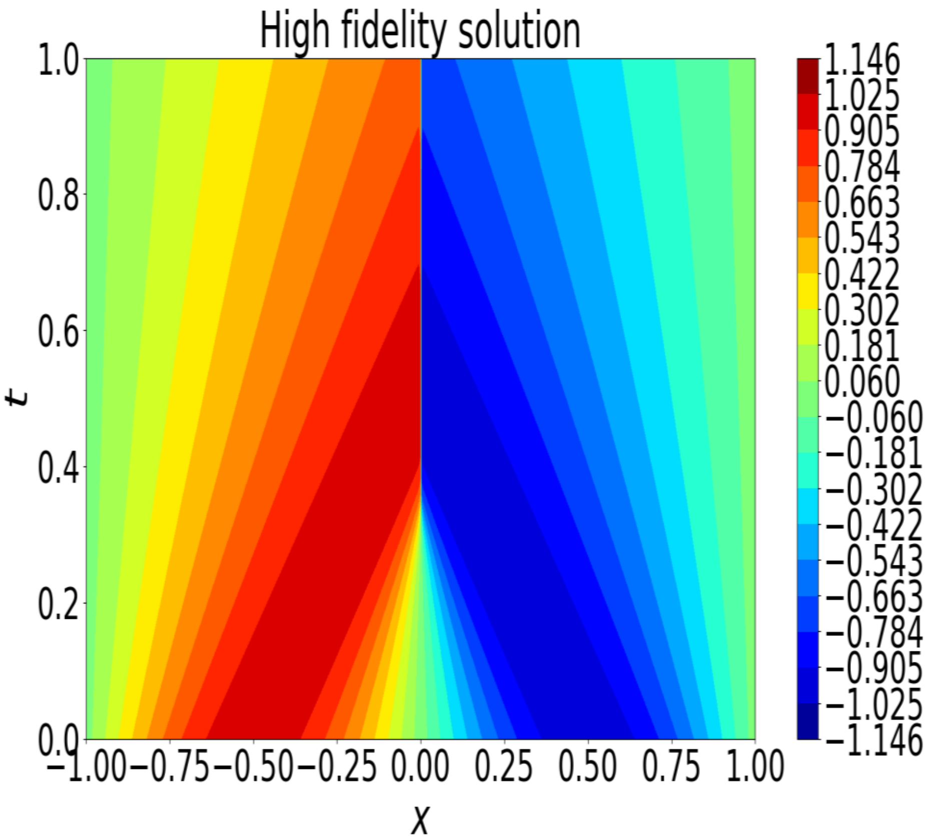
Example based on the Burgers' equation*, a nonlinear hyperbolic equation

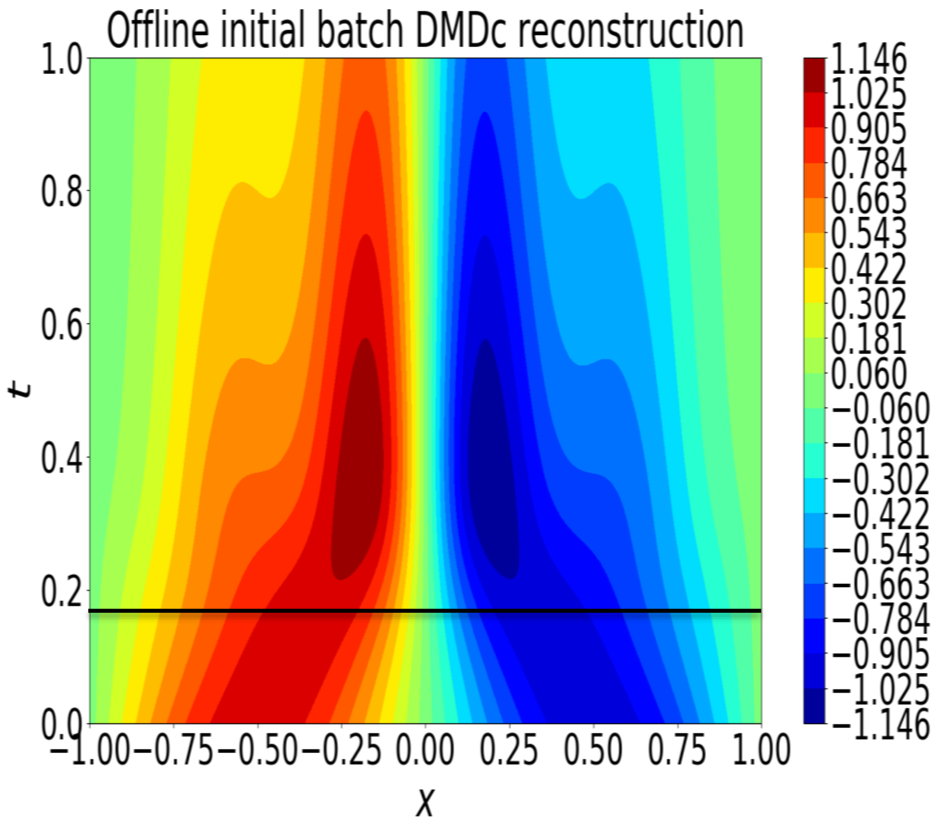


*: J. Burgers, A mathematical model illustrating the theory of turbulence, Advances in Applied Mechanics, (1948), pp. 171–199.

Nonlinear example

Example based on the Burgers' equation*, a nonlinear hyperbolic equation





*: J. Burgers, A mathematical model illustrating the theory of turbulence, Advances in Applied Mechanics, (1948), pp. 171–199.

- Introduction to reduced Dynamic Mode Decomposition with control:
  - Dynamic Mode Decomposition with control
  - Proper Orthogonal Decomposition
- Recursive Proper Orthogonal Decomposition
- Recursive reduced Dynamic Mode Decomposition with control
- Conclusion and Prospects

BUT!

Everything used is for a time invariant system based on **complete $\boldsymbol{X_{snap}}$**

- System is identified using DMDc based on the available snapshot matrix

$$X(k+1) = AX(k) + BQ(k)$$

- POD is applied based on available snapshot matrix

$$X_r(k) = U_r^T X(k)$$

! ***Purely Offline methods***

Recursive update of the system model and the POD modes matrix

- Changes in the system along with online processing
- Time varying systems

$$X_r(k) = U_{r,k}^T X(k)$$
$$X_r(k+1) = A_r(k) X_r(k) + B_r(k) Q(k)$$

Recursive POD

Aim: Update the POD modes matrix recursively:

- SVD of the new snapshot matrix after receiving a new snapshot
  - ! High computation requirements (slow processing of order $(N_x^2 N_t)$)

- Recursive Least Squares approach (specifically Projection Approximation Subspace Tracking (PAST) method**)

*: B. Yang, Projection approximation subspace tracking, IEEE Transactions on Signal Processing

Least Squares Approach

- Update the POD recursively using a Recursive Least Squares approach (specifically Projection Approximation Subspace Tracking (PAST) method*)

$$\min_{rank(X)\leq K} \left\| \boldsymbol{X_{snap}} - \boldsymbol{R} \right\|_F = \left\| \boldsymbol{X_{snap}} - \widehat{\boldsymbol{X}} \right\|_F$$

$$\widehat{\boldsymbol{X}} = U_r \Sigma_\mathrm{r} \mathrm{V}_\mathrm{r}^\mathrm{T} = U_r U_r^T X_{snap}$$

- Simplified cost function

$$\left\| \boldsymbol{X_{snap}} - \widehat{\boldsymbol{X}} \right\|_F^2 = \sum_{k=1}^{i} \lambda^{i-k} \left\| X(k) - U_{r,i} U_{r,i}^T X(k) \right\|_2^2 ; \lambda \in]0;1[$$

- PAST algorithm **approximation**

$$U_{r,i}^T X(k) \approx U_{r,i-1}^T X(k)$$

$$\sum_{k=1}^{i} \lambda^{i-k} \left\| X(k) - U_{r,i} U_{r,i}^T X(k) \right\|_2^2 \approx \sum_{k=1}^{i} \lambda^{i-k} \left\| X(k) - U_{r,i} U_{r,i-1}^T X(k) \right\|_2^2$$

✓ ✓ ✓

- Solvable with Recursive Least Squares (RLS) algorithm

- Introduction to reduced Dynamic Mode Decomposition with control:
  - Dynamic Mode Decomposition with control
  - Proper Orthogonal Decomposition
- Recursive Proper Orthogonal Decomposition
- Recursive reduced Dynamic Mode Decomposition with control
- Conclusion and Prospects

Recursive Reduced DMDc

Aim: Update the reduced identified system recursively:

$$X_r(k+1) = U_{r,k+1}^T X(k+1) = A_{r,k} X_r(k) + B_{r,k} Q(k)$$

- DMDc on the available snaphsot matrix initiating A_r and B_r

$$X_r(k+1) = \begin{pmatrix} A_{r,k} & B_{r,k} \end{pmatrix} \begin{pmatrix} X_r(k) \\ Q(k) \end{pmatrix}$$

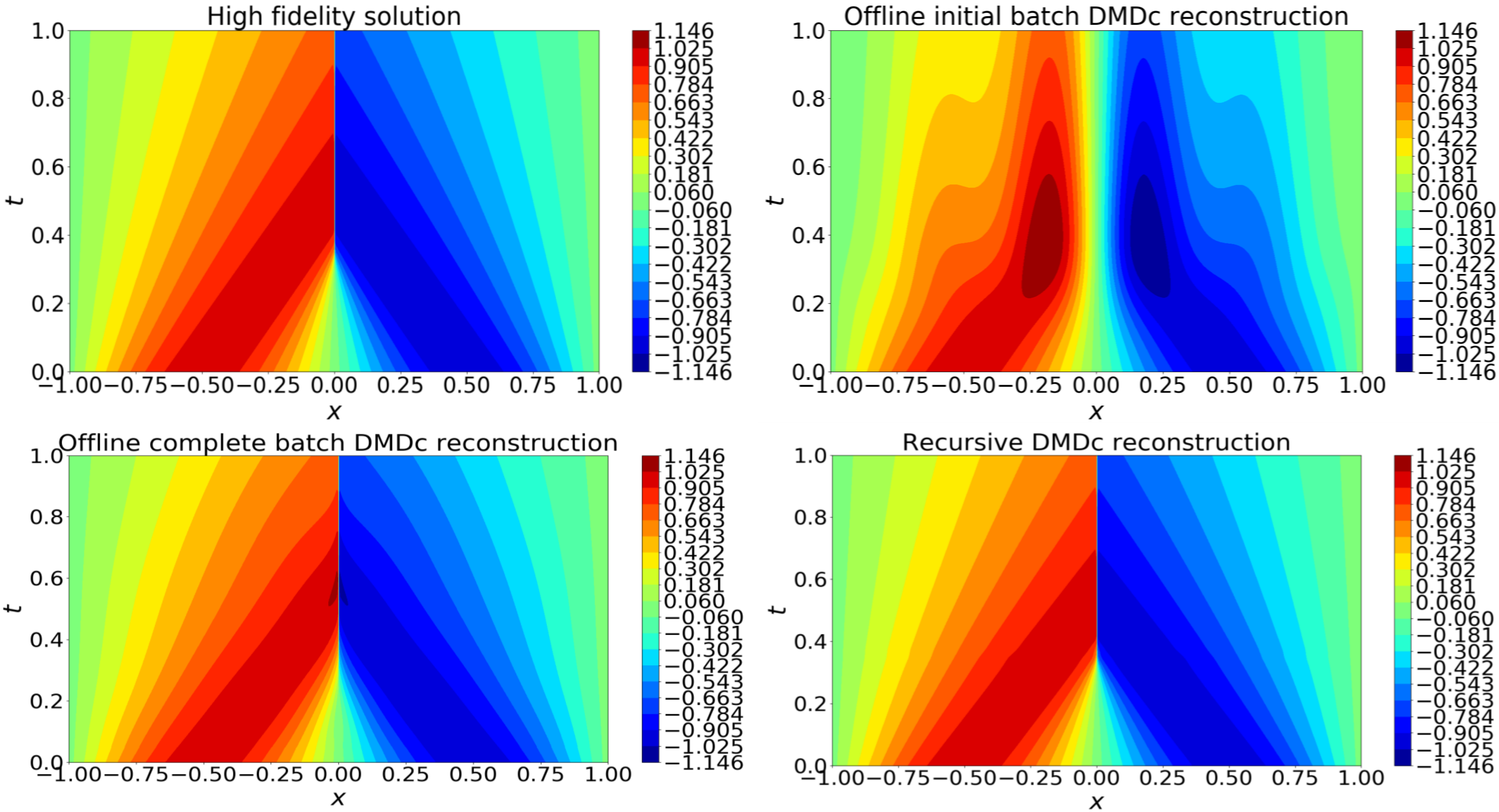
- Cost function

$$min \left\| X_r(k+1) - \begin{pmatrix} A_{r,k} & B_{r,k} \end{pmatrix} \begin{pmatrix} X_r(k) \\ Q(k) \end{pmatrix} \right\|_2^2$$

- Find $A_{r,k}$ and $B_{r,k}$ **recursively** using Recursive Least Square Algorithm

Nonlinear example

Example based on the Burgers' equation*, a nonlinear hyperbolic equation

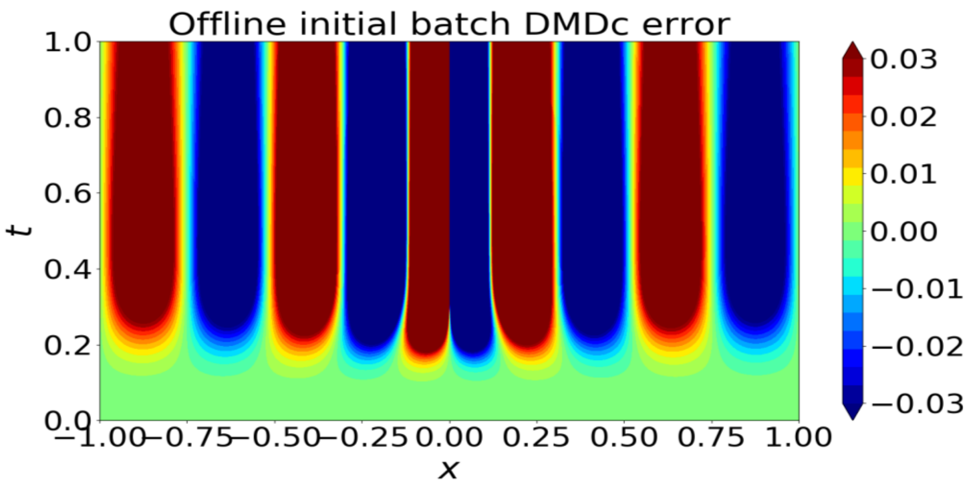


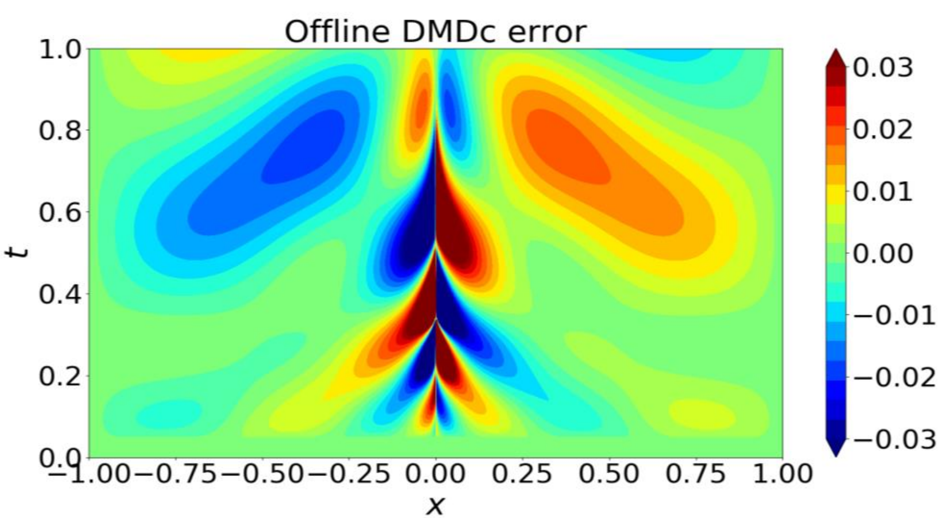
*: J. Burgers, A mathematical model illustrating the theory of turbulence, Advances in Applied Mechanics, (1948), pp. 171–199.

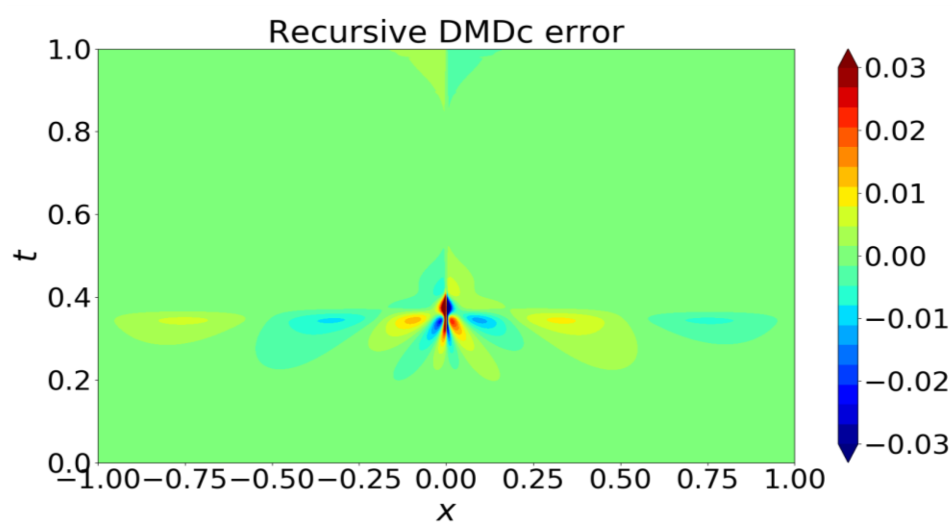
Recursive rDMDc Residuals

$$BFT = 100\left(\frac{||X-\hat{X}||_2}{||X-mean(X)||_2}\right)$$

Method	BFT (%)
Offline initial batch DMDc	62,38
Offline complete batch DMDc	96,8
Recursive DMDc	99,42



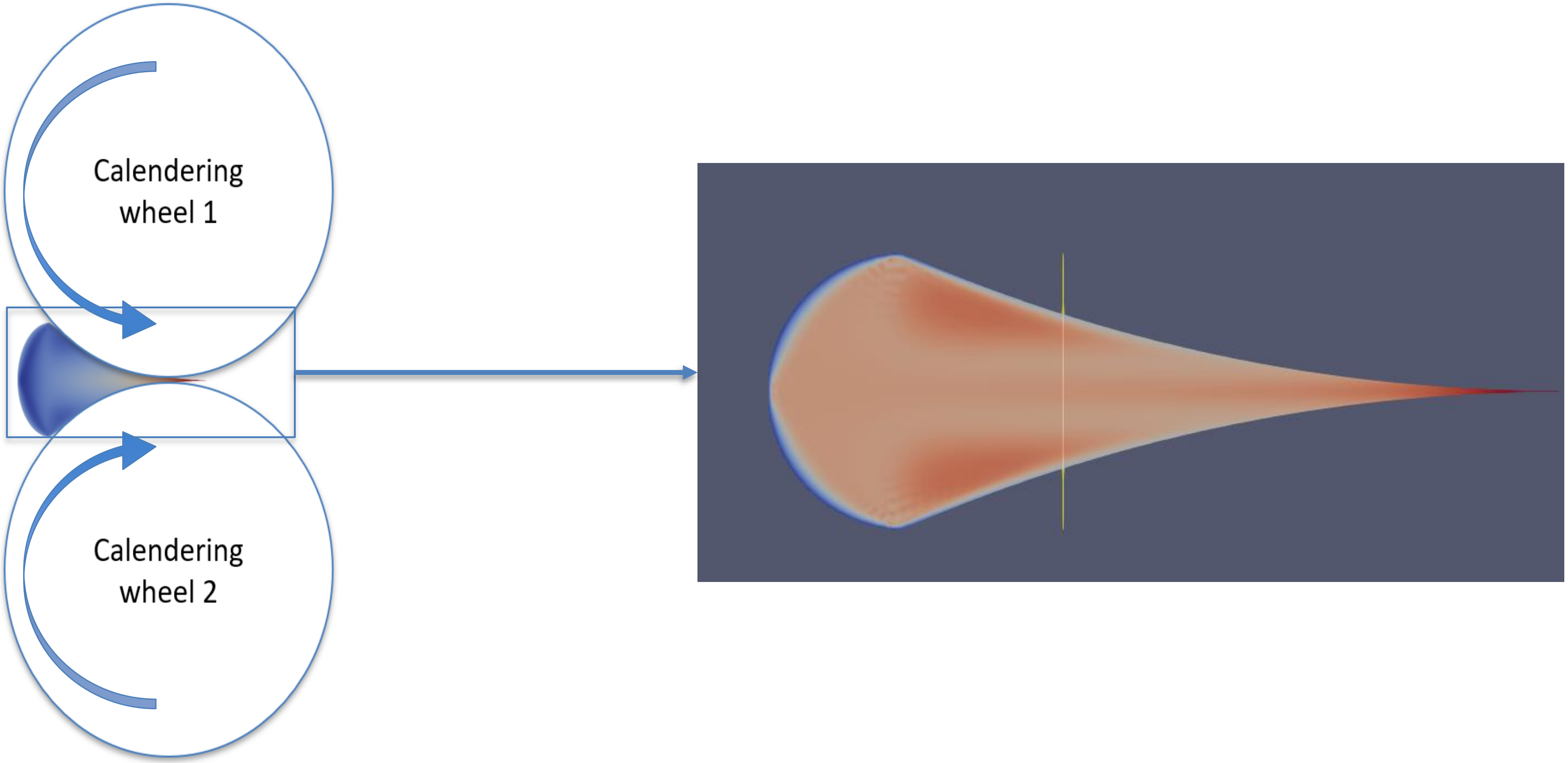




MICHELIN

Calendering Process

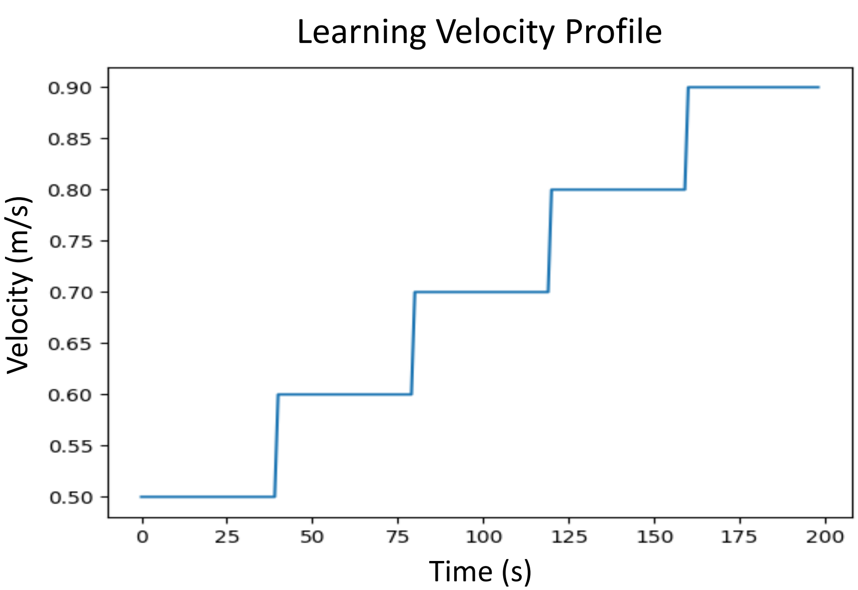
Calendering process revisited: Finite Elements solver (MEF++)*

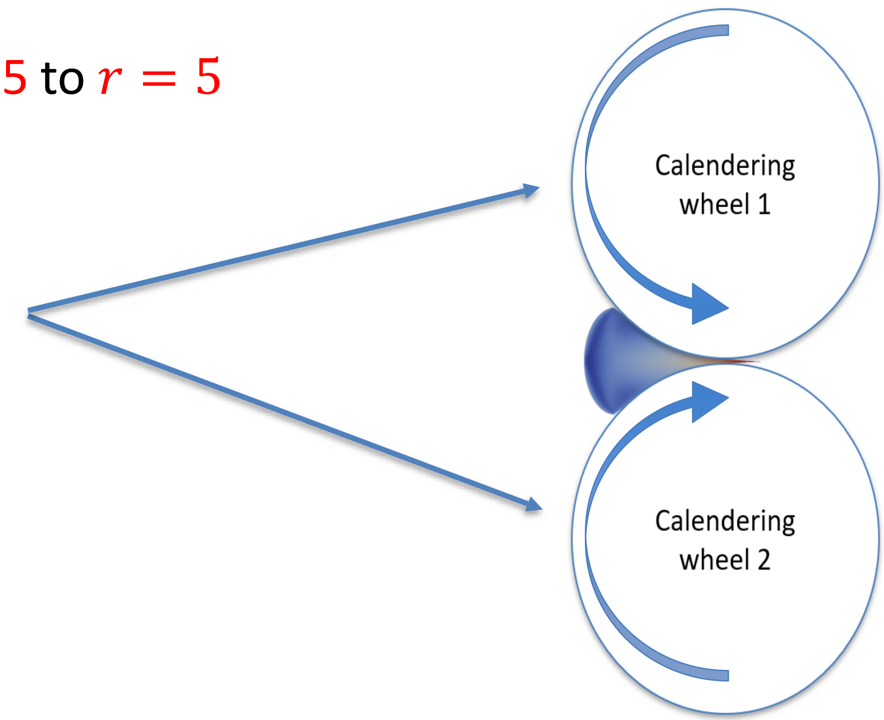


*: https://fr.wikipedia.org/wiki/MEF%2B%2B

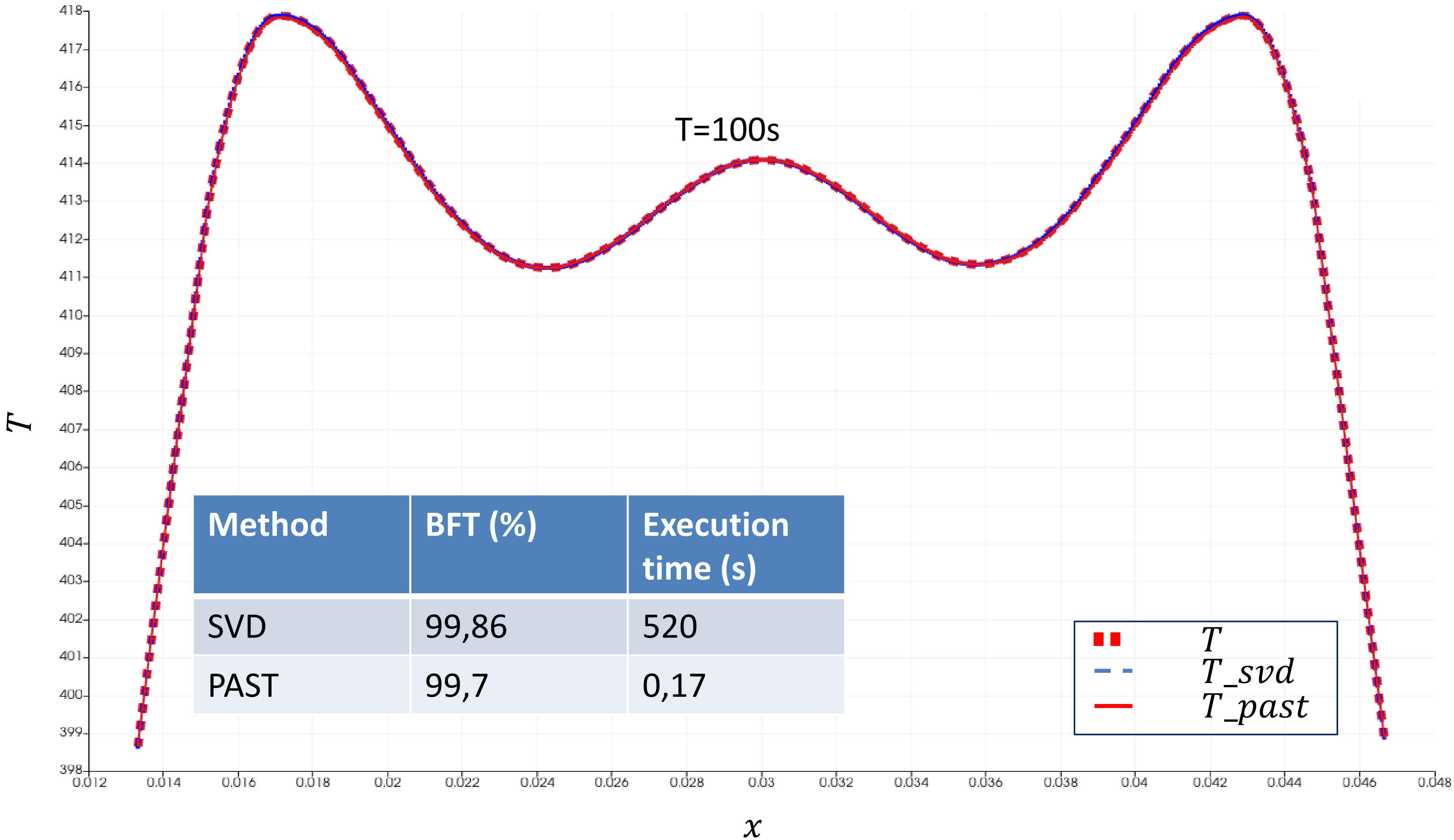
Industrial Application

- Data collection (X_{snap}) from the inhouse Finite Elements (FEM) solver using step input
- Model reduction based on updating the POD matrix using SVD with sliding window or PAST algorithm
- System Identification based on DMDc with reduction or recursive DMDc method with reduction
- Model order reduction from N_x=8385 to $r = 5$



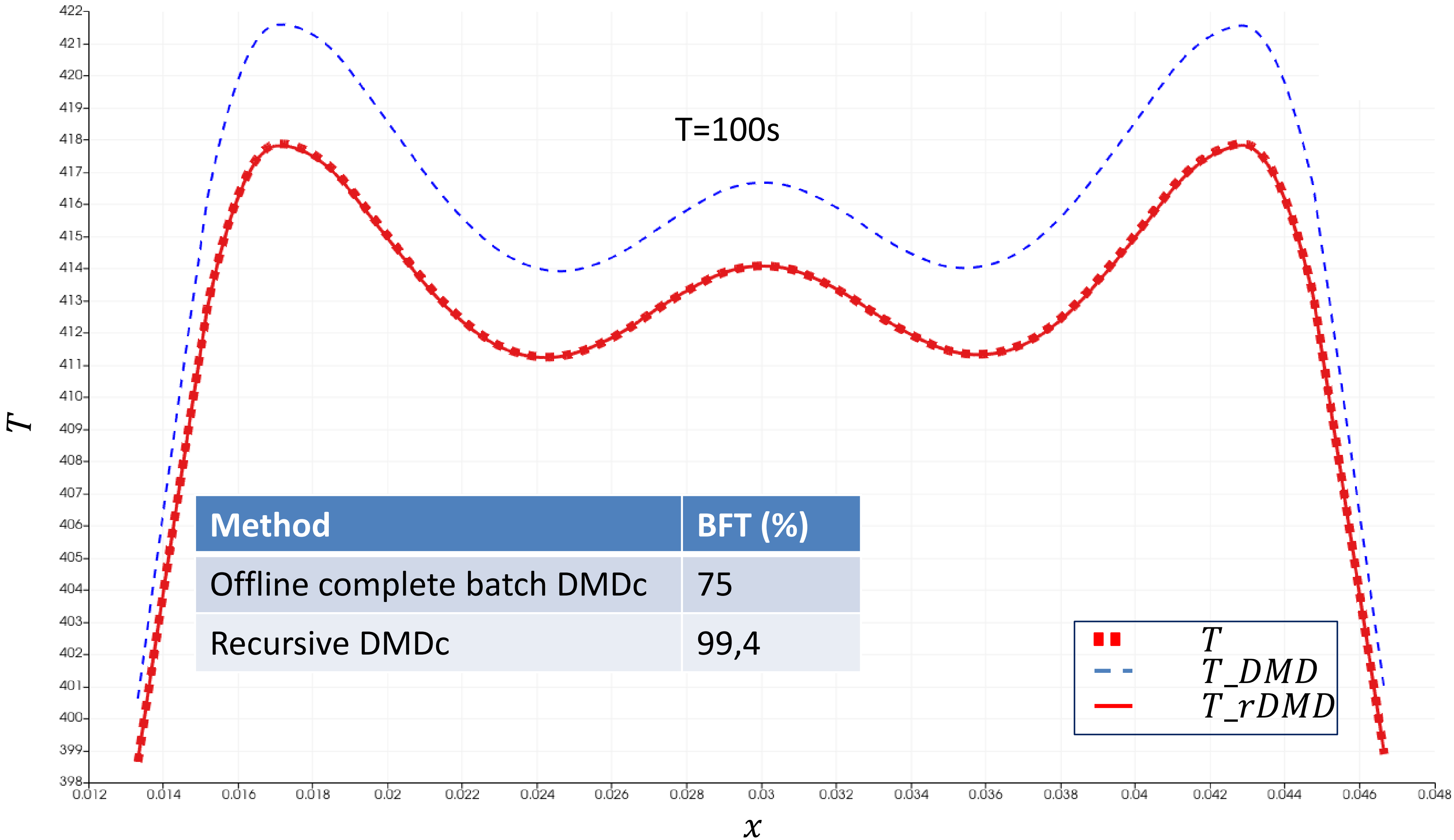


POD Update Preliminary Results



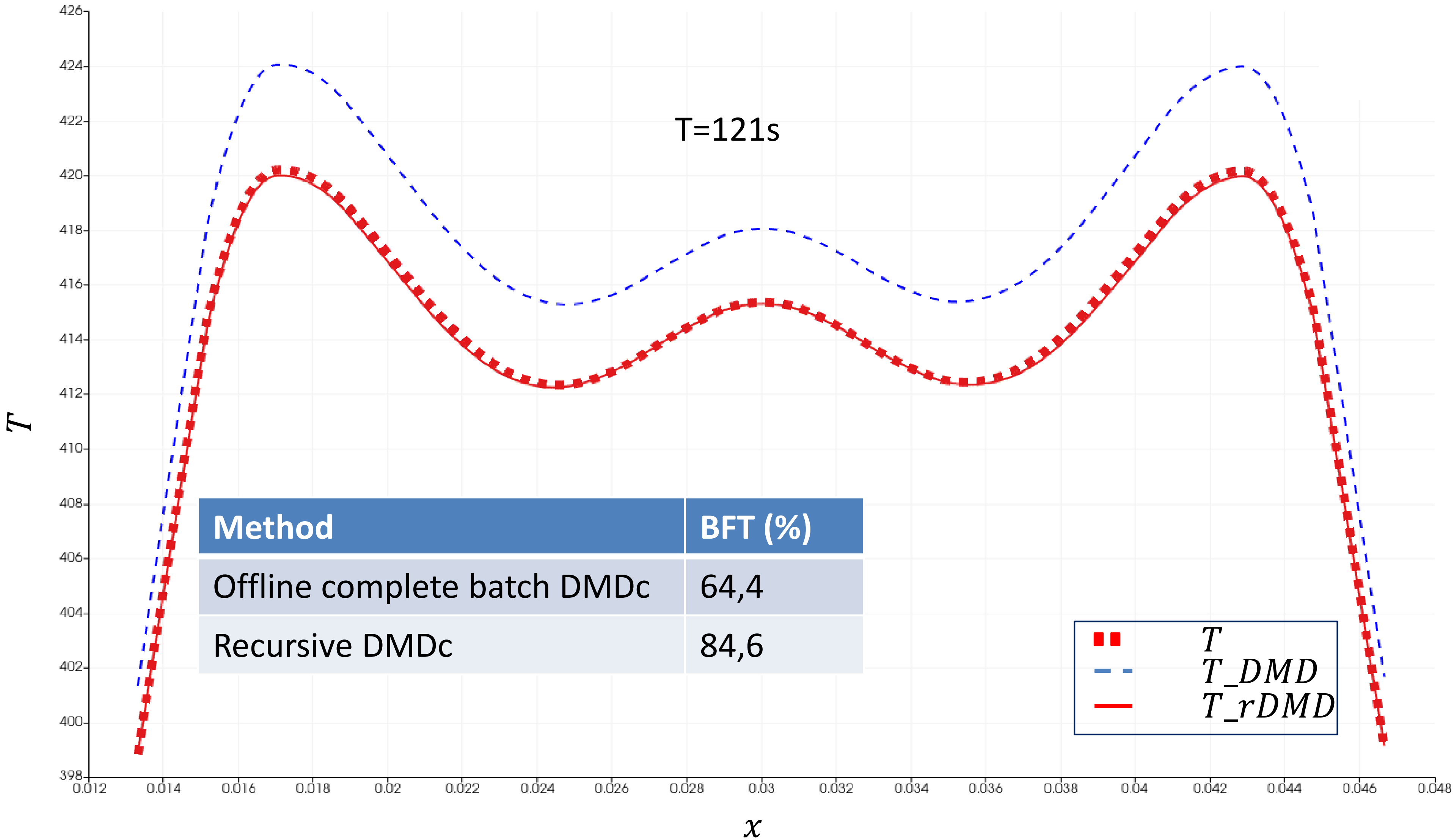
Method	BFT (%)	Execution time (s)
SVD	99,86	520
PAST	99,7	0,17

System Identification Preliminary Results (1/2)



Method	BFT (%)
Offline complete batch DMDc	75
Recursive DMDc	99,4

System Identification Preliminary Results (2/2)



Method	BFT (%)
Offline complete batch DMDc	64,4
Recursive DMDc	84,6

- Introduction to reduced Dynamic Mode Decomposition with control:
  - Dynamic Mode Decomposition with control
  - Proper Orthogonal Decomposition
- Recursive Proper Orthogonal Decomposition
- Recursive reduced Dynamic Mode Decomposition with control
- Conclusion and Prospects

MICHELIN

Conclusion and Prospects

- **Data-driven system identification** using the Dynamic Mode Decomposition with control
- **Reduction** of the system model order using Proper Orthogonal Decomposition
- **Recursive** Proper orthogonal Decomposition and Dynamic Mode Decomposition
- Validation of methods on a linear time varying example, nonlinear example, industrial application simulator

Prospects:

- Output measurement study
- Observer requirements and structure
- Model Predictive Control implementation

MICHELIN

THANK YOU

Discussions?