A statistical POD approach for feedback control in fluid dynamics

Luca Saluzzi

Scuola Normale Superiore

Joint work with S. Dolgov (Bath) and D. Kalise (Imperial College)

Nonlinear partial differential equations: theory, numerics and applications,
Sapienza, Università di Roma
24/05/2023

Ciao Maurizio!

Figure – Picture with Maurizio and nonno Angelo (Corigliano Calabro, August 2022).

HJB equation for the infinite horizon problem

Controlled Dynamics and Cost Functional

$$\begin{cases} \dot{y}(s) = f(y(s)) + B(y(s))u(s), & s \in (0, +\infty), \\ y(0) = x \in \mathbb{R}^d. \end{cases}$$

$$u(t) \in \mathcal{U} = \{u : [0, +\infty) \to U \subset \mathbb{R}^m \; measurable\},$$

$$J(u(\cdot), x) := \int_0^{+\infty} y(s)^\top Q y(s) + u^\top(s) R u(s) \, ds \, ,$$

HJB equation for the infinite horizon problem

Controlled Dynamics and Cost Functional

$$\begin{cases} \dot{y}(s) = f(y(s)) + B(y(s))u(s), & s \in (0, +\infty), \\ y(0) = x \in \mathbb{R}^d. \end{cases}$$

$$u(t) \in \mathcal{U} = \{u : [0, +\infty) \to U \subset \mathbb{R}^m \; measurable\},$$

$$J(u(\cdot), x) := \int_0^{+\infty} y(s)^\top Q y(s) + u^\top(s) R u(s) \, ds \, ,$$

Value Function

$$V(x) := \inf_{u \in \mathcal{U}} J(u(\cdot), x),$$

HJB equation for the infinite horizon problem

HJB Equations

$$\inf_{u \in U} \left\{ (f(x) + B(x)u)^\top \nabla V(x) + x^\top Q x + u^\top R u \right\} = 0.$$

HJB equation for the infinite horizon problem

HJB Equations

$$\inf_{u \in U} \left\{ (f(x) + B(x)u)^\top \nabla V(x) + x^\top Q x + u^\top R u \right\} = 0.$$

Feedback and HJB equation for $U = \mathbb{R}^m$

$$u^*(x) = -\frac{1}{2} R^{-1} B(x)^\top \nabla V(x)$$

$$\nabla V(x)^\top f(x) - \frac{1}{4} \nabla V(x)^\top B(x) R^{-1} B(x)^\top \nabla V(x) + x^\top Q x = 0 \quad x \in \Omega \subset \mathbb{R}^d$$

HJB equation for the infinite horizon problem

HJB Equations

$$\inf_{u\in U}\left\{(f(x)+B(x)u)^\top \nabla V(x)+x^\top Qx+u^\top Ru\right\}=0.$$

Feedback and HJB equation for $U=\mathbb{R}^m$

$$u^*(x)=-\frac{1}{2}R^{-1}B(x)^\top \nabla V(x)$$

$$\nabla V(x)^\top f(x)-\frac{1}{4}\nabla V(x)^\top B(x)R^{-1}B(x)^\top \nabla V(x)+x^\top Qx=0 \quad x\in\Omega\subset\mathbb{R}^d$$

Warning : Curse of dimensionality

d-dimensional HJB PDE, where d is the dimension of the dynamical system.

Taming the Curse of Dimensionality

Some literature

- $d-$dimensional state space$\Rightarrow$ HJB in $\mathbb{R}^d$ approximated with a tensorial grid : N^d degrees of freedom.
- Numerical PDEs for physical space problems $d \leq 3+1$
- For $d \leq 8$: sparse grids (Bokanowski et al. 13′, Kang and Wilcox 15′, Garcke and Kröner 17′).
- Max-plus algebra (McEneaney 06′, Akian-Gaubert-Lakhoua 08′).
- Representation formulas (Osher-Darbon 16′, Yegorov-Dower 17′).
- Deep learning for Lyapunov functions (Grüne 21′, Grüne-Sperl 23′)
- Tensor decompositions for HJB (Horowitz et al. 14′, Oster et al. 19′, Dolgov et al. 19′).

Data-Driven approach

Regression problem

Given sample points $\{x_i\}_{i=1}^N$ and a dataset $\{V(x_i), \nabla V(x_i)\}_{i=1}^N$, we want to solve

$$\inf_{\tilde{V}} \sum_{i=1}^{N} |\tilde{V}(x_i) - V(x_i)|^2 + \lambda \|\nabla \tilde{V}(x_i) - \nabla V(x_i)\|^2,$$

Data-Driven approach

Regression problem

Given sample points $\{x_i\}_{i=1}^N$ and a dataset $\{V(x_i), \nabla V(x_i)\}_{i=1}^N$, we want to solve

$$\inf_{\tilde{V}} \sum_{i=1}^{N} |\tilde{V}(x_i) - V(x_i)|^2 + \lambda \|\nabla \tilde{V}(x_i) - \nabla V(x_i)\|^2,$$

Questions

- How do we compute the dataset $\{V(x_i), \nabla V(x_i)\}_{i=1}^N$?
- How we represent the approximation $\tilde{V}$?
- How we solve the minimization problem ?

Collection of the data

First option : Pontryagin Maximum Principle (PMP)

State/adjoint system :

$$\begin{cases} \frac{d}{dt} y^*(t) = f(y(t)) + B(y^*(t))u^*(t), \\ y^*(0) = x, \\ -\frac{d}{dt} p_i^*(t) = \sum_{j=1}^n p_j^*(t)\partial_{y_j}(f_j(y^*(t)) + B_j(y^*(t))u^*(t)) + 2(Qy^*)_i, \quad i = 1, \ldots, d, \\ p_i(T) = 0, \\ u^*(t) = -\frac{1}{2} R^{-1} B(y^*(t))^\top p^*(t). \end{cases}$$

$$V(x) \approx \int_0^\top y^*(s)^\top Q y^*(s) + u^*(s)^\top R u^*(s)\, ds,$$

$$\nabla V(x) \approx p(0).$$

Collection of the data

First option : Pontryagin Maximum Principle (PMP)

State/adjoint system :

$$\begin{cases} \frac{d}{dt} y^*(t) = f(y(t)) + B(y^*(t))u^*(t), \\ y^*(0) = x, \\ -\frac{d}{dt} p_i^*(t) = \sum_{j=1}^n p_j^*(t)\partial_{y_j}(f_j(y^*(t)) + B_j(y^*(t))u^*(t)) + 2(Qy^*)_i, \quad i = 1, \dots, d, \\ p_i(T) = 0, \\ u^*(t) = -\frac{1}{2} R^{-1} B(y^*(t))^\top p^*(t). \end{cases}$$

$$V(x) \approx \int_0^\top y^*(s)^\top Q y^*(s) + u^*(s)^\top R u^*(s)\, ds,$$

$$\nabla V(x) \approx p(0).$$

Warning

It is an expensive Boundary Value Problem !

Collection of the data

Second option : State-Dependent Riccati Equation

Consider the Linear Quadratic Regulator problem

$$\dot{y}(s) = Ay + Bu(s), \quad y(0) = x,$$

$$J(u(\cdot), x) := \int_0^{+\infty} y(s)^\top Q y(s) + u^\top(s) R u(s)\, ds,$$

$$V(x) = x^\top \Pi x,$$

$$A^\top \Pi + \Pi A - \Pi B R^{-1} B^\top \Pi + Q = 0,$$

$$u(x) = -R^{-1} B^\top \Pi x$$

Collection of the data

Second option : State-Dependent Riccati Equation

Consider the semilinear case

$$\dot{y}(s) = A(y)y + B(y)u(s), \quad y(0) = x,$$

$$J(u(\cdot), x) := \int_0^{+\infty} y(s)^\top Q y(s) + u^\top(s) R u(s)\, ds\,,$$

$$V(x) = x^\top \Pi(x) x,$$

$$A^\top(x)\Pi(x) + \Pi(x)A(x) - \Pi(x)B(x)R^{-1}B^\top(x)\Pi(x) + Q = 0\,,$$

$$u(x) = -\frac{1}{2}R^{-1}B^\top(x)\Pi(x)$$

Connection between HJB and SDRE

Plugging the ansatz $V(x) = x^\top \Pi(x) x$ into the HJB we obtain

$$H_{ARE}(x) + E(x) = 0,$$

where

$$H_{ARE}(x) = x^\top \left(A^\top(x)\Pi(x) + \Pi(x)A(x) - \Pi(x)B(x)R^{-1}B^\top(x)\Pi(x) + Q \right) x$$

$$E(x) = \varphi(x)\left(2\left[A(x) - W(x)\Pi(x)\right] x - W(x)\varphi^T(x) \right),$$

$$(\varphi(x))_k = \frac{1}{2} \sum_i \sum_j x_i x_j \partial_{x_k} \Pi_{i,j}(x).$$

State-Dependent Riccati Equation

Connection between HJB and SDRE

Plugging the ansatz $V(x) = x^\top \Pi(x) x$ into the HJB we obtain

$$H_{ARE}(x) + E(x) = 0,$$

where

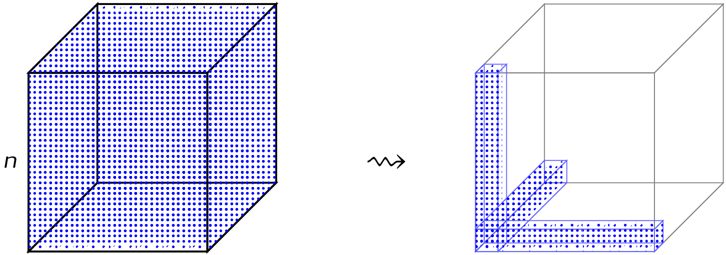
$$H_{ARE}(x) = x^\top \left(A^\top(x)\Pi(x) + \Pi(x)A(x) - \Pi(x)B(x)R^{-1}B^\top(x)\Pi(x) + Q \right) x$$

$$E(x) = \varphi(x)\Big(2\,[A(x) - W(x)\Pi(x)]\,x - W(x)\varphi^T(x)\Big),$$

$$(\varphi(x))_k = \frac{1}{2}\sum_i \sum_j x_i x_j \partial_{x_k} \Pi_{i,j}(x).$$

It is possible to minimize $E(x)$ playing with the infinite semilinear forms (Dolgov-Kalise-S. 22′)

Low-rank tensor decomposition ⇔ separation of variables :



$\mathcal{O}(n^d)$ $\mathcal{O}(dn)$

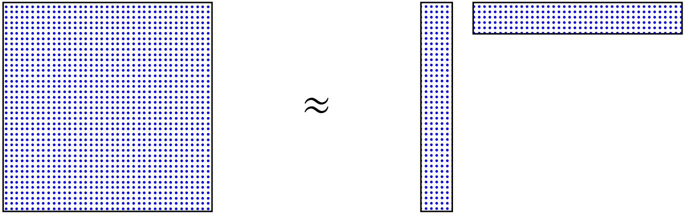
- Approximate : $\underbrace{V(x_1,\dots,x_d)}_{\text{tensor}} \approx \underbrace{\sum_{\alpha} V_\alpha^1(x_1)V_\alpha^2(x_2)\cdots V_\alpha^d(x_d)}_{\text{tensor product decomposition}}$.

- Derivative : 1D derivatives of each V^k

2 variables : low-rank matrices

- Collect discretised values of $V(x_1, x_2)$ into a matrix :

$$V(i,j) = \sum_{\alpha=1}^{r} U_\alpha(i) W_\alpha(j) + \mathcal{O}(\varepsilon)$$

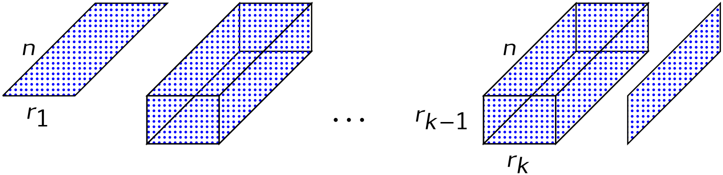


- Rank $r \ll n$.
- $\mathtt{mem}(U) + \mathtt{mem}(W) = 2nr \ll n^2 = \mathtt{mem}(U)$.
- Singular Value Decomposition : optimal $\varepsilon(r)$ dependence.

$$\|V - UW^*\|_F^2 \to \min_{V,W}$$

Tensor Train (TT) decomposition (Oseledets 2011)

$$V(i_1 \dots i_d) = \sum_{\substack{\alpha_k=1 \\ 0<k<d}}^{r_k} G^1_{\alpha_1}(i_1) \cdot G^2_{\alpha_1,\alpha_2}(i_2) \cdot G^3_{\alpha_2,\alpha_3}(i_3) \cdots G^d_{\alpha_{d-1}}(i_d)$$



- TT blocks G^k are **three-dimensional** tensors with
- **TT ranks** $r_1, \dots, r_{d-1} \leq r$.
- Storage : $\mathcal{O}(dnr^2)$, instead of $\mathcal{O}(n^d) \rightarrow$ depends on the growth of the rank r
- We solved problems up to dimension $d = 100$ with $n = 5$ Legendre basis.

Cucker-Smale model

Dynamics

Given N_a agents, we introduce $(y, v) \in \mathbb{R}^{N_a} \times \mathbb{R}^{N_a}$:

$$\begin{cases} \dot{y}_i = v_i, & i = 1, \ldots N_a \\ \dot{v}_i = \frac{1}{N_a} \sum_{j=1}^{N_a} \frac{v_j - v_i}{1 + \|y_i - y_j\|^2} + u_i & i = 1, \ldots N_a \end{cases}$$

Cucker-Smale model

Dynamics

Given N_a agents, we introduce $(y, v) \in \mathbb{R}^{N_a} \times \mathbb{R}^{N_a}$:

$$\begin{cases} \dot{y}_i = v_i, & i = 1, \dots N_a \\ \dot{v}_i = \frac{1}{N_a} \sum_{j=1}^{N_a} \frac{v_j - v_i}{1 + \|y_i - y_j\|^2} + u_i & i = 1, \dots N_a \end{cases}$$

Cost functional

$$J(y(\cdot), v(\cdot), u(\cdot)) = \frac{1}{N_a} \int_0^\infty \|y(s)\|^2 + \|v(s)\|^2 + \|u(s)\|^2 \, ds.$$

We fix $\Omega = [-0.5, 0.5]^{2N_a}$ and 5 Legendre basis in each variable.

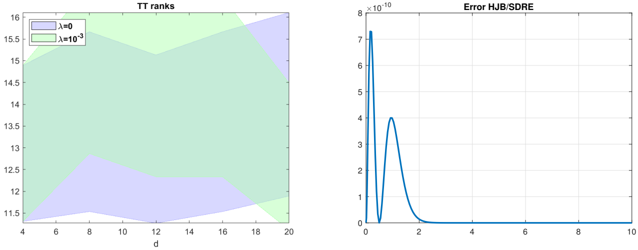


Figure – Shaded area denotes mean ± 1 standard deviation over 10 runs.

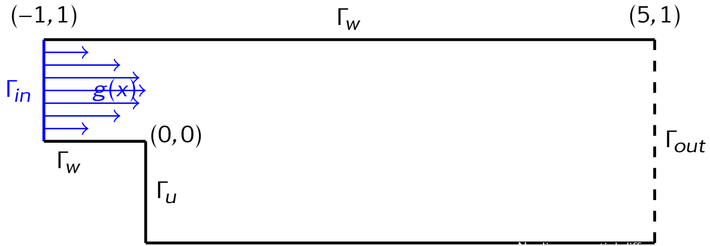
d	SDRE	TT
10	$1.4e-3s$	$1.8e-5s$
20	$5.4e-3s$	$6.9e-5s$
30	$1.0e-2s$	$1.6e-4s$
40	$2.2e-2s$	$3.3e-4s$
100	$1.3e-1s$	$5.3e-3s$

Table – Averaged CPU time for a single computation of the suboptimal control for the different methods.

Optimal control of Navier-Stokes equation

$$
\begin{cases}
\partial_t y - \nu \Delta y + y \cdot \nabla y + \nabla p = 0 & (x,t) \in \Omega \times [0,T], \\
\nabla \cdot y = 0 & (x,t) \in \Omega \times [0,T], \\
y = g(x) & (x,t) \in \Gamma_{in} \times [0,T], \\
y = 0 & (x,t) \in \Gamma_{w} \times [0,T], \\
y = u(t) & (x,t) \in \Gamma_{u} \times [0,T], \\
\nu \partial_n y - p\vec{n} = 0 & (x,t) \in \Gamma_{out} \times [0,T], \\
y(x,0) = y_0(x) & x \in \Omega,
\end{cases}
$$

$$
J_T(y,u) = \int_0^T \int_\Omega |\nabla \times y(s,x)|^2 \, dx \, ds + \int_0^T \delta |u(s)|^2 \, ds
$$



Semidiscretization of NS equation

Finite elements

By using the stable $P_2 - P_1$ Taylor-Hood finite elements pair, involving bilinear elements $\{\varphi_k\}_{k=1}^{N_p}$ for the pressure and biquadratic elements $\{\phi_k\}_{k=1}^{N_v}$ for the velocity :

$$\begin{cases} M\dot{y}(t) + Ay(t) + N(y(t))y(t) + B^\top p(t) = 0, \\ By(t) = 0, \end{cases}$$

Boundary conditions

Since we are dealing with non-homogeneous Dirichlet boundary condition, we consider the solution y as the sum of two functions

$$y = \widetilde{y} + \underline{y},$$

where $\widetilde{y}$ takes into account the boundary conditions, while $\underline{y}$ has homogeneous boundary conditions. In particular, $\widetilde{y}$ is chosen as the solution of the following Stokes equation

$$\begin{cases} -\nu\Delta\widetilde{y} + \nabla p = 0 & (x,t) \in \Omega \times [0,T], \\ \nabla \cdot \widetilde{y} = 0 & (x,t) \in \Omega \times [0,T], \end{cases}$$

The discrete system admits a solution in the form

$$\widetilde{y} = (B_1 + \widetilde{B}[u_1(t), u_2(t)]^\top) = \mathbf{Bu}(t)$$

Discretization of NS equation

Time discretization

Applying an implicit Euler scheme for the time :

$$\begin{bmatrix} G^n & B^\top \\ B & 0 \end{bmatrix} \begin{bmatrix} \underline{y}^n \\ \underline{p}^n \end{bmatrix} = \begin{bmatrix} M(\underline{y}^{n-1} + \widetilde{y}^{n-1}) - G^n \widetilde{y}^n \\ 0 \end{bmatrix},$$

where $G^n = M + \Delta t(A + N(\underline{y}^n + \widetilde{y}^n))$.
It is a nonlinear system of equations approximated via a Newton's method.

Discretization of NS equation

Time discretization

Applying an implicit Euler scheme for the time :

$$\begin{bmatrix} G^n & B^\top \\ B & 0 \end{bmatrix} \begin{bmatrix} \underline{y}^n \\ \underline{p}^n \end{bmatrix} = \begin{bmatrix} M(\underline{y}^{n-1} + \widetilde{y}^{n-1}) - G^n \widetilde{y}^n \\ 0 \end{bmatrix},$$

where $G^n = M + \Delta t(A + N(\underline{y}^n + \widetilde{y}^n))$.
It is a nonlinear system of equations approximated via a Newton's method.

Parameters

We fix $\nu = 2 \cdot 10^{-3}$, $T = 20$, $n_t = 80$, $y_0 \equiv 0$, $N_v = 5191$, $N_p = 1340 \rightsquigarrow$ d=11722

Discretization of NS equation

Time discretization

Applying an implicit Euler scheme for the time :

$$\begin{bmatrix} G^n & B^\top \\ B & 0 \end{bmatrix} \begin{bmatrix} \underline{y}^n \\ \underline{p}^n \end{bmatrix} = \begin{bmatrix} M(\underline{y}^{n-1} + \widetilde{y}^{n-1}) - G^n \widetilde{y}^n \\ 0 \end{bmatrix},$$

where $G^n = M + \Delta t(A + N(\underline{y}^n + \widetilde{y}^n))$.
It is a nonlinear system of equations approximated via a Newton's method.

Parameters

We fix $\nu = 2 \cdot 10^{-3}$, $T = 20$, $n_t = 80$, $y_0 \equiv 0$, $N_v = 5191$, $N_p = 1340 \rightsquigarrow$ d=11722

POD

We need a tool to reduce first the dimension of the system $\rightsquigarrow$ Proper Orthogonal Decomposition

Proper Orthogonal Decomposition and SVD

Given snapshots $[y(t_0),\ldots,y(t_n)] \in \mathbb{R}^{d\times(n+1)}$

We look for an orthonormal basis $\{\psi_i\}_{i=1}^{\ell}$ in $\mathbb{R}^d$ with $\ell \ll \min\{n,m\}$ s.t.

$$J(\psi_1,\ldots,\psi_\ell) = \sum_{j=0}^{n} \left\| y_j - \sum_{i=1}^{\ell} \langle y_j, \psi_i \rangle \psi_i \right\|^2 = \sum_{i=\ell+1}^{d} \sigma_i^2$$

reaches a minimum :

$$\min J(\psi_1,\ldots,\psi_\ell) \quad s.t. \langle \psi_i, \psi_j \rangle = \delta_{ij}$$

Singular Value Decomposition : $Y = \Psi \Sigma V^T$.

For $\ell \in \{1,\ldots,d = rank(Y)\}$, $\{\psi_i\}_{i=1}^{\ell}$ are called **POD basis** of rank ℓ.

ERROR INDICATOR : $\mathcal{E}(\ell) = \dfrac{\sum_{i=1}^{\ell} \sigma_i^2}{\sum_{i=1}^{d} \sigma_i^2}$ with σ_i singular values of the SVD.

Statistical Proper Orthogonal Decomposition

Some literature

The coupling of the POD and HJB was introduced in (Kunisch et al, $04'$) and then applied in a series of works. The reduction usually relies on the basis coming from the **uncontrolled dynamics**.

Statistical Proper Orthogonal Decomposition

Some literature

The coupling of the POD and HJB was introduced in (Kunisch et al, 04′) and then applied in a series of works. The reduction usually relies on the basis coming from the **uncontrolled dynamics**.

Drawbacks and novelties of our work

- In general the uncontrolled dynamics does not give information on the optimum and it may **blow up**
- We are interested in approximating the (sub)optimal trajectories, not all the possible solutions of the dynamical system
- Our aim is to build reduced basis for different initial conditions and boundary conditions

Statistical Proper Orthogonal Decomposition

Some literature

The coupling of the POD and HJB was introduced in (Kunisch et al, 04′) and then applied in a series of works. The reduction usually relies on the basis coming from the **uncontrolled dynamics**.

Drawbacks and novelties of our work

- In general the uncontrolled dynamics does not give information on the optimum and it may **blow up**
- We are interested in approximating the (sub)optimal trajectories, not all the possible solutions of the dynamical system
- Our aim is to build reduced basis for different initial conditions and boundary conditions

Stochastic dynamics

Given a complete probability space $(\Omega, \mathcal{F}, \mathbb{P})$, we consider a stochastic controlled dynamics

$$\begin{cases} \dot{y}(s,\omega) = f(y(s,\omega), u(s,\omega), \omega), \quad s \in (0,+\infty), \omega \in \Omega, \\ y(0,\omega) = x \in \mathbb{R}^d, \end{cases}$$

Offline and Online Stages

Algorithm Offline Stage for the Statistical POD method

Fix the final time T, a time discretization $\{t_i\}_{i=1}^{n_t}$, N realizations $\{\omega_i\}_{i=1}^{N}$ and $Y_{\underline{\omega}} = [\,]$
for $i = 1, \dots, N$ **do**
    Solve the PMP system
    Compute the optimal trajectory $Y_{\omega_i} = [y^*(t_1, \omega_i), \dots, y^*(t_{n_t}, \omega_i)]$
    $Y_{\underline{\omega}} = [Y_{\underline{\omega}} \; Y_{\omega_i}]$
end for
Perform the SVD $Y_{\underline{\omega}} = U_{\underline{\omega}} \Sigma_{\underline{\omega}} V_{\underline{\omega}}^\top$
Select ℓ according to the error indicator and build U^ℓ with the first ℓ columns of $U_{\underline{\omega}}$
Construct the reduced dynamics $(U^\ell)^\top f(U^\ell y^\ell(s, \omega), u(s, \omega), \omega)$

Algorithm Offline Stage for the Statistical POD method

1: Fix the final time T, a time discretization $\{t_i\}_{i=1}^{n_t}$, N realizations $\{\omega_i\}_{i=1}^{N}$ and $Y_{\underline{\omega}} = [\,]$
2: **for** $i = 1, \dots, N$ **do**
3: Solve the PMP system
4: Compute the optimal trajectory $Y_{\omega_i} = [y^*(t_1, \omega_i), \dots, y^*(t_{n_t}, \omega_i)]$
5: $Y_{\underline{\omega}} = [Y_{\underline{\omega}}\ Y_{\omega_i}]$
6: **end for**
7: Perform the SVD $Y_{\underline{\omega}} = U_{\underline{\omega}} \Sigma_{\underline{\omega}} V_{\underline{\omega}}^\top$
8: Select ℓ according to the error indicator and build U^ℓ with the first ℓ columns of $U_{\underline{\omega}}$
9: Construct the reduced dynamics $(U^\ell)^\top f(U^\ell y^\ell(s,\omega), u(s,\omega), \omega)$

Online Stage

The reduced dynamics is beneficial for the computation of fast online controls and approximation the value function.
The reduced feedback map $u^\ell : \mathbb{R}^\ell \to U$ will then be applied to Full dynamical system :

$$\dot{y}(s,\omega) = f(y(s,\omega), u^\ell((U^\ell)^\top y(s,\omega)), \omega),$$

obtaining a control problem where the computation of the optimal feedback is independent from the original dimension of the dynamical system.

Application to NS equation

Offline Stage

For the application of the statistical POD technique we consider the following stochastic inflow

$$g(x) = 4x_2(1-x_2) + \frac{1}{2}\sum_{k=1}^{N_\omega} k^{-\gamma} sin(2k\pi x_2)\zeta_k,$$

with $N_\omega = 8$, $\gamma = 3$ and $\{\zeta_k\}_k$ are uniform random variables in $[-c, c]$.
The performances will be studied according to the following error indicators

$$err_{J,1} = |J(u) - J_{red}(u)|, \quad err_{J,2} = |J(u_{red}) - J_{red}(u_{red})|, \quad err_{J,3} = |J(u) - J(u_{red})|$$

Online Stage

The dynamics is written in semilinear form

$$\dot{y}^\ell(t) = \mathcal{A}^\ell(y^\ell(t))y^\ell(t) + \mathcal{B}^\ell(y^\ell(t))\mathbf{u}(t).$$

and the optimal control is solved via TT Gradient Cross with data given by the SDRE in the reduced domain of dimension ℓ. We will investigate the error in the computation of the cost functional by the TT method with the error indicator

$$err_{TT} = |J(u_{red,TT}) - J(u_{red})|$$

Results

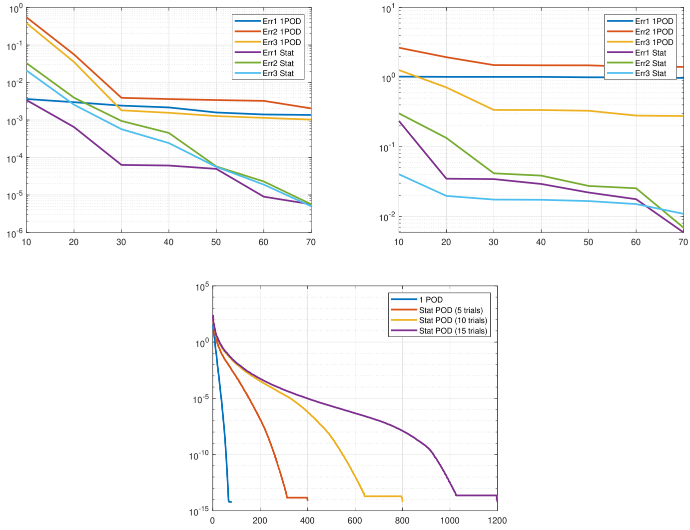


Figure – Top : mean errors study on 10 iid random inflows with $c = 1$ (right) and $c = 2$ (left) for statistical POD built upon $N = 15$ realisations. Bottom : Singular values of $Y_{\underline{\omega}}$ for different numbers of offline realisations N.

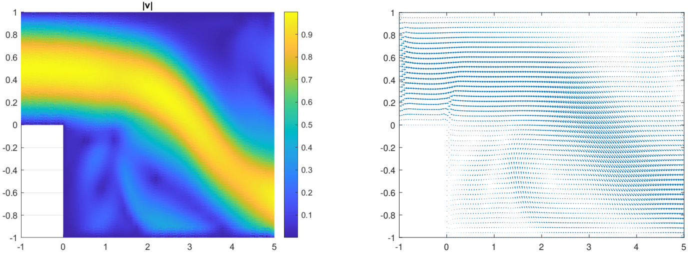
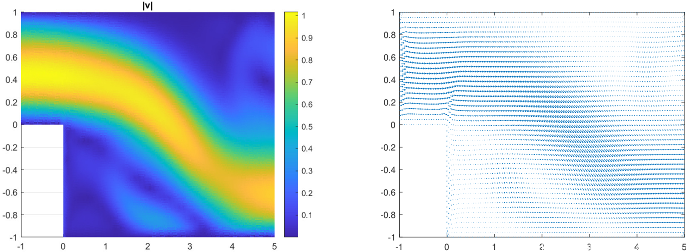


Figure – Top : Uncontrolled case, Bottom : LQR controller. Mean velocity (left) and velocity vector field (right) at final time $T = 20$.



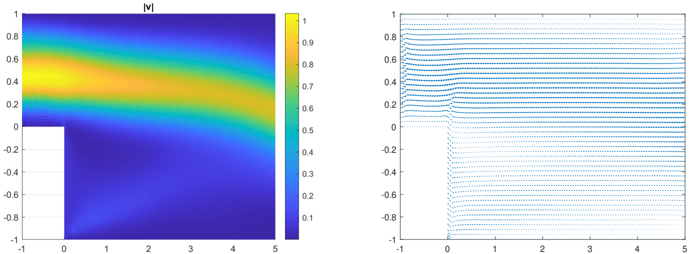


Figure – (Controlled case) Mean velocity (left) and velocity vector field (right) at final time $T = 20$. The optimal control is computed via Tensor Train Cross and Statistical POD with $\ell = 20$ basis.

ℓ	SDRE POD	SDRE SPOD	LQR POD	LQR SPOD	err_{TT} POD	err_{TT} SPOD
5	3.0234	**2.9927**	3.2341	3.2169	$2.49e-1$	**3.60e-2**
10	2.9941	**2.9674**	3.1952	3.1952	$3.50e-1$	**6.21e-3**
20	3.0880	**2.9527**	3.2034	3.1642	$1.19e-1$	**1.62e-2**

Table – Total cost and error in the TT approximation for different techniques.

Conclusions and perspectives

- We have developed a data-driven method for the approximation of high-dimensional infinite horizon optimal control laws
- We also showed that the TT rank in the first example is independent of the dimension, yielding a effective mitigation of the curse of dimensionality
- We introduced a Statistical POD approach to reduce further the dimension and to take into accounts different initial conditions/boundary conditions
- We applied the procedure the control of Navier-Stokes, showing the reduction of the turbulence via the reduced control

- We aim at considering more challenging problems and studying theoretically the algorithm

References

- S. Dolgov, D. Kalise, and K. K. Kunisch. Tensor Decomposition Methods for High-dimensional Hamilton–Jacobi–Bellman Equations. SIAM Journal on Scientific Computing, 43(3) :A1625–A1650, 2021
- S. Dolgov, D. Kalise and L. Saluzzi, Data-driven Tensor Train Gradient Cross Approximation for Hamilton-Jacobi-Bellman Equations, to appear on SISC.
- S. Dolgov, D. Kalise and L. Saluzzi, Optimizing semilinear representations for State-dependent Riccati Equation-based feedback control, IFAC, 2022.
- S. Dolgov, D. Kalise and L. Saluzzi, A statistical POD approach for boundary optimal control in fluid dynamics, in preparation

- TT Gradient cross solver publicly available at `https://github.com/saluzzi/TT-Gradient-Cross`

References

- S. Dolgov, D. Kalise, and K. K. Kunisch. Tensor Decomposition Methods for High-dimensional Hamilton–Jacobi–Bellman Equations. SIAM Journal on Scientific Computing, 43(3) :A1625–A1650, 2021
- S. Dolgov, D. Kalise and L. Saluzzi, Data-driven Tensor Train Gradient Cross Approximation for Hamilton-Jacobi-Bellman Equations, to appear on SISC.
- S. Dolgov, D. Kalise and L. Saluzzi, Optimizing semilinear representations for State-dependent Riccati Equation-based feedback control, IFAC, 2022.
- S. Dolgov, D. Kalise and L. Saluzzi, A statistical POD approach for boundary optimal control in fluid dynamics, in preparation

- TT Gradient cross solver publicly available at `https://github.com/saluzzi/TT-Gradient-Cross`

Thank you for the attention! :)