Towards the control of transitional flows: a machine-learning perspective

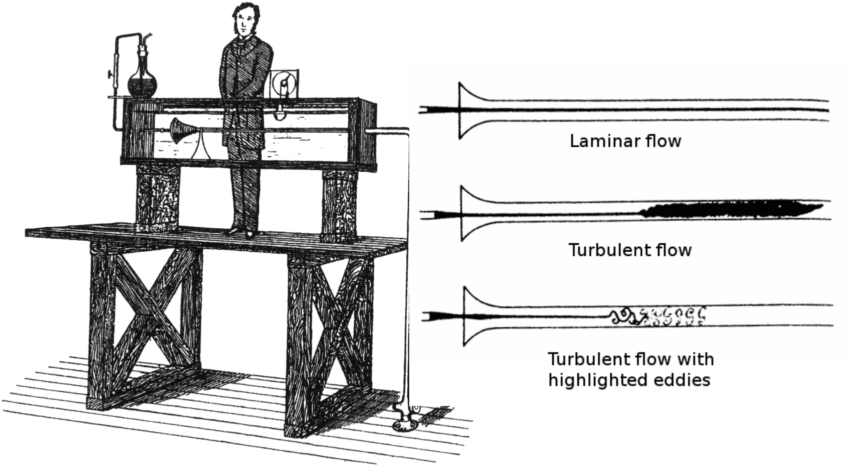
Anton Pershin[1, 2]
anton.pershin@physics.ox.ac.uk

[1]Department of Physics, University of Oxford
[2]School of Mathematics, University of Leeds

Joint work with Cedric Beaume, Kuan Li, Steven Tobias

Simons Group Meeting
October 25, 2021

Reynolds experiment



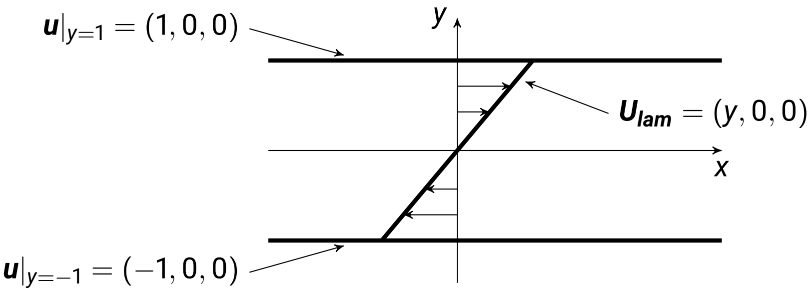
Reynolds, Phil. Trans. R. Soc. London, 174 (1884)

Plane Couette flow

Navier–Stokes equation:

$$\partial_t \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = -\nabla p + \frac{1}{Re}\nabla^2 \boldsymbol{u}$$

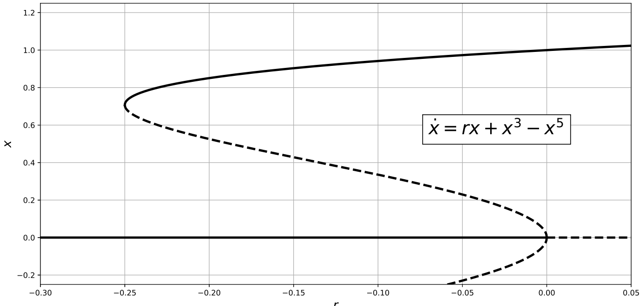
Incompressibility condition: $\nabla \cdot \boldsymbol{u} = 0$
Streamwise and spanwise directions: periodic BCs
Wall-normal direction: no-slip BCs



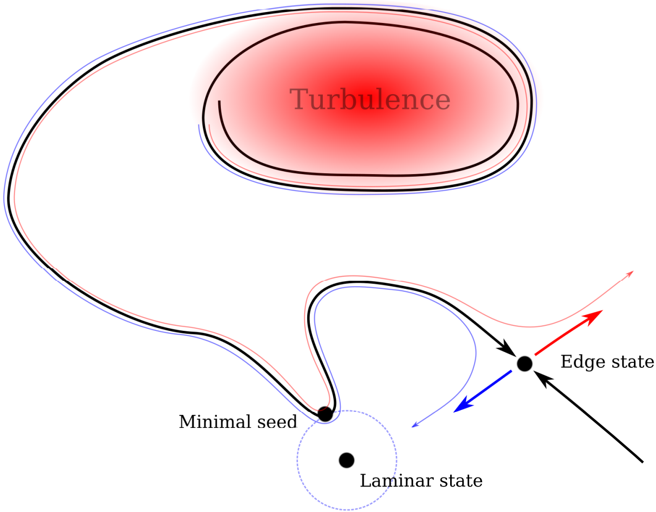
Subcritical transitional flows

	Linearly stable laminar state	Sustained turbulence
Plane Couette flow	all Re	$Re \gtrsim 325$
Pipe flow	all Re	$Re \gtrsim 2040$
Plane Poiseuille flow	$Re \lesssim 5772$	$Re \gtrsim 840$

Transition is complicated by the coexistence of two attractive states:



Edge of chaos is wrapped up around the turbulent saddle[1]

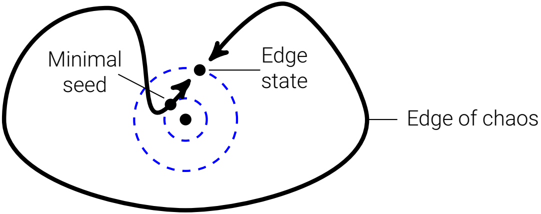


[1]Chantry *et al.*, J. Fluid Mech. **747** (2014)

How robust is the laminar state to perturbations?

Indicators of stability:

- Infinitesimal perturbations $\Longrightarrow$ linear growth rate
- Finite-amplitude perturbations $\Longrightarrow$ the size of the basin of attraction



Laminarisation probability $P_{lam}(E)$ is the probability that a random finite perturbation of energy E laminarises

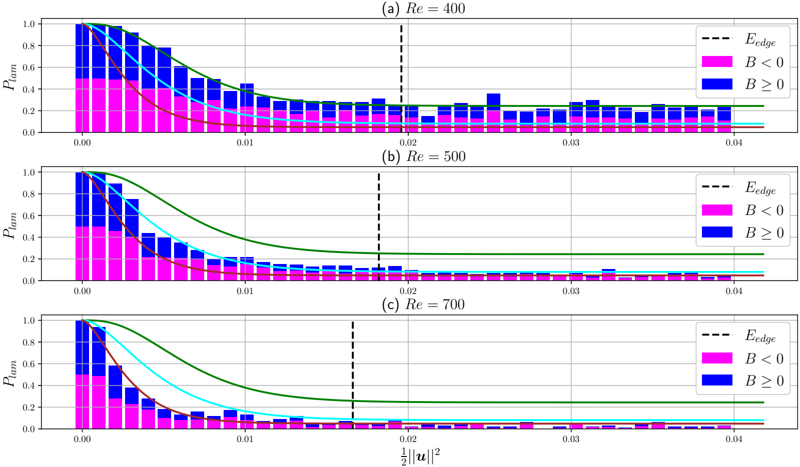
Random perturbation:

$$\boldsymbol{u} = A\boldsymbol{u}_{\perp} + B\boldsymbol{U}_{lam},$$

where $A, B, \boldsymbol{u}_{\perp}$ are generated randomly and $\langle \boldsymbol{u}_{\perp}, \boldsymbol{U}_{lam} \rangle = 0$

Laminarisation probability

- $P_{lam}(E)$ approximates the size of the basin of attraction
- Laminarisation probability fitting: $p(E) = 1 - (1-a)\gamma(\alpha, \beta E)$
- Control strategies can be assessed by comparing $P_{lam}(E)$

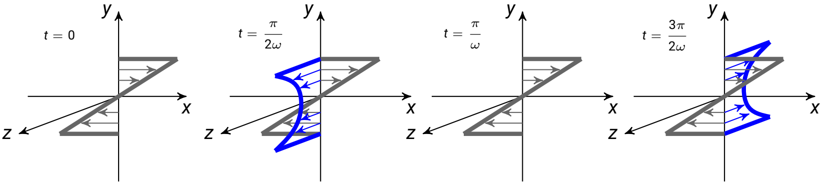


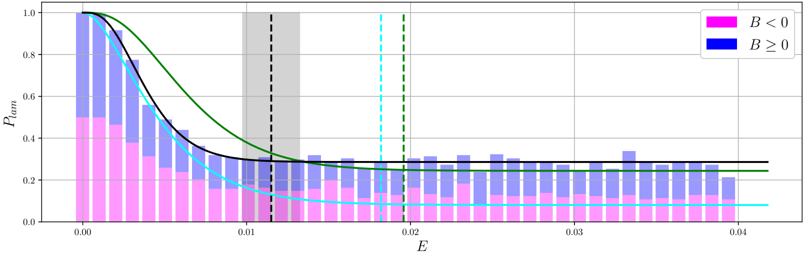
Control strategy: wall oscillations

We impose in-phase oscillations on the walls[2]:

$$\boldsymbol{u}(x, \pm 1, z, t) = \pm \boldsymbol{e_x} + W_{osc} sin(\omega t) \boldsymbol{e_z}$$

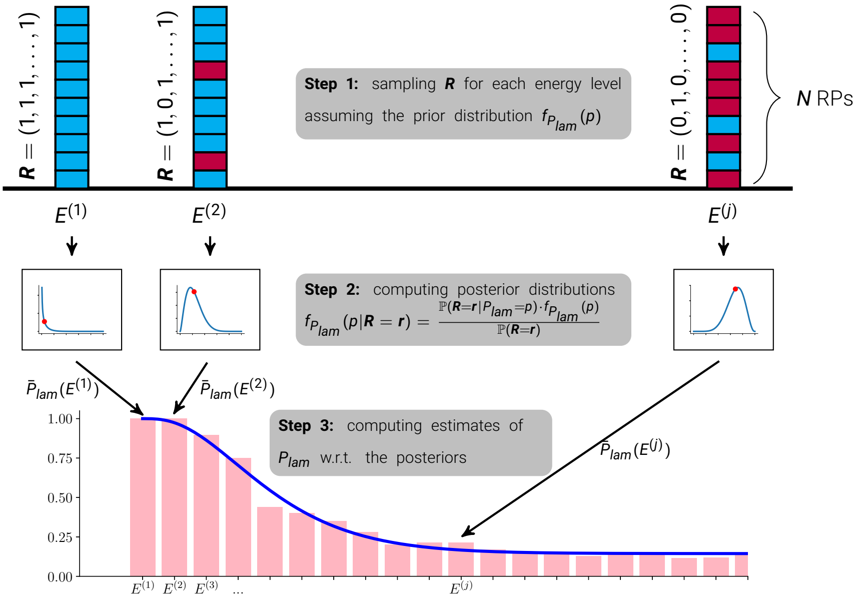
$$\Longrightarrow \quad \boldsymbol{U_{lam}} = y\boldsymbol{e_x} + W(y,t)\boldsymbol{e_z}.$$





[2] Motivated by Rabin *et al.*, J. Fluid Mech. **738** (2014)

Bayesian inference of laminarization probability

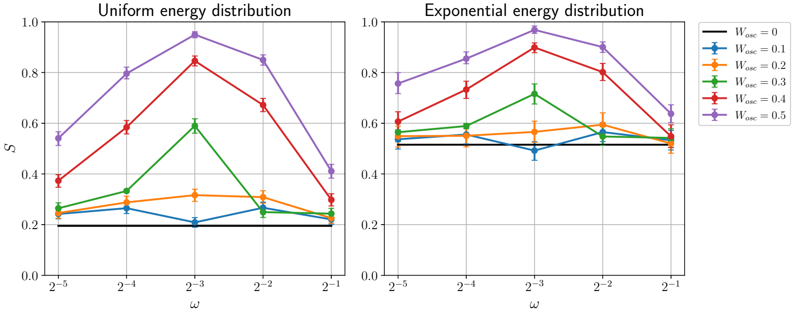


Laminarization score

- Now we can estimate the laminarization score S:

$$S = \int_0^{E_{max}} p(E) f_E(E) dE,$$

- It is assumed that the perturbation energy is distributed as $f_E(E)$
- This is an efficient method for the assessment of laminar flow robustness for a wide range of control parameter values[3]



[3] Pershin, Beaume and Tobias, *submitted, arXiv:2108.07629* (2021)

Learning transition to turbulence via reservoir computing

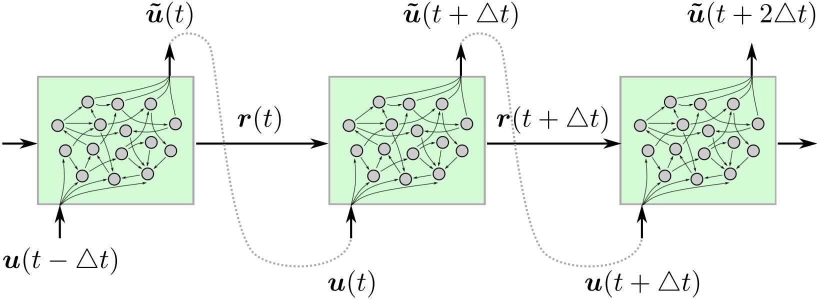
Echo State Network (ESN)

Echo State Network is a reservoir-computing architecture:

$$\boldsymbol{r}(t+\Delta t) = \tanh(\boldsymbol{b} + \boldsymbol{W}_{in}\boldsymbol{u}(t) + \boldsymbol{W}\boldsymbol{r}(t)) + \xi Z,$$
$$\tilde{\boldsymbol{u}}(t+\triangle t) = \boldsymbol{W}_{out}\boldsymbol{r}(t+\Delta t).$$

where

- $\boldsymbol{W}_{in}$ and $\boldsymbol{W}$ are random sparse matrices
- $\boldsymbol{b}$ is a random bias
- Z is a random variable and ξ is the noise strength

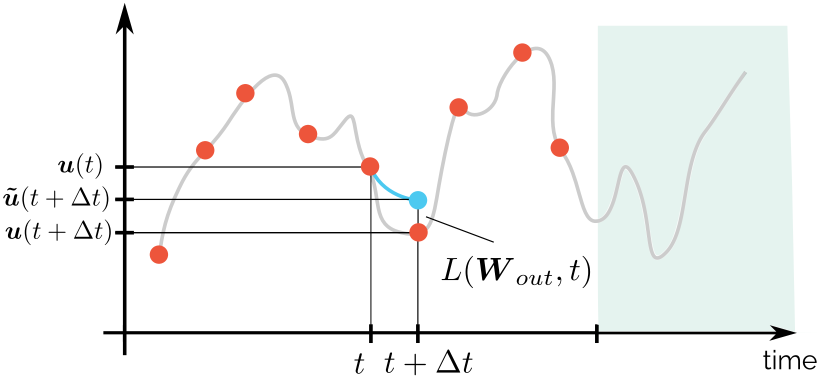


Training

Minimization of the residual sum of squares (RSS):

$$\min_{\boldsymbol{W}_{out}} \sum_{k=1}^{N_t} ||\boldsymbol{W}_{out}\boldsymbol{r}(k\Delta t) - \boldsymbol{u}(k\Delta t)||_2^2.$$

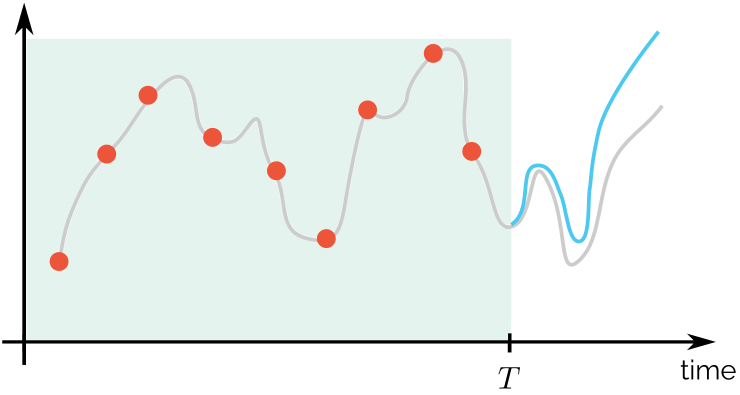
Solution for $\boldsymbol{W}_{out}$ is found via the normal equation.



Prediction mode:

$$\boldsymbol{r}(t+\Delta t) = \tanh(\boldsymbol{b} + \boldsymbol{W}_{in}\tilde{\boldsymbol{u}}(t) + \boldsymbol{W}\boldsymbol{r}(t)) + \xi Z,$$
$$\tilde{\boldsymbol{u}}(t+\triangle t) = \boldsymbol{W}_{out}\boldsymbol{r}(t+\Delta t).$$

We still need to specify the initial condition $\boldsymbol{r}(T)$.

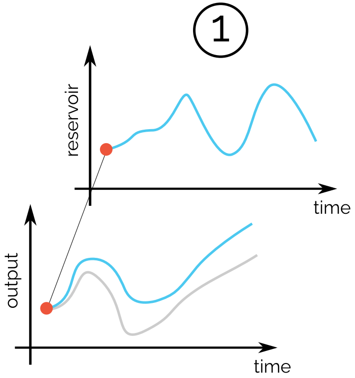


Prediction

Prediction mode:

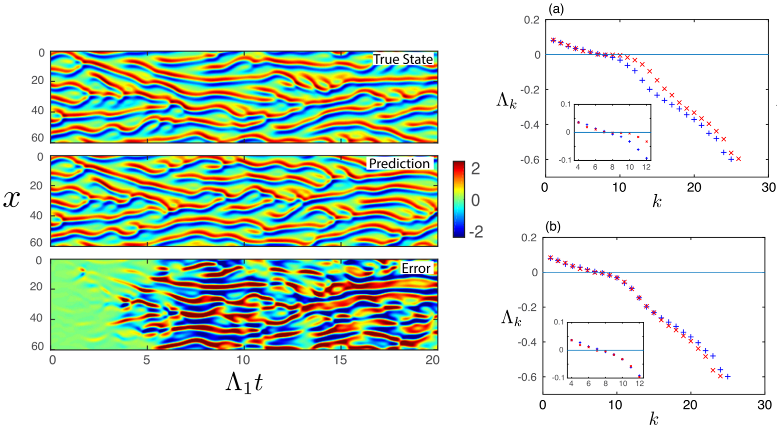
$$\boldsymbol{r}(t+\Delta t) = \tanh(\boldsymbol{b} + \boldsymbol{W}_{in}\boldsymbol{u}(t) + \boldsymbol{W}\boldsymbol{r}(t)) + \xi Z,$$
$$\tilde{\boldsymbol{u}}(t+\triangle t) = \boldsymbol{W}_{out}\boldsymbol{r}(t+\Delta t).$$

We still need to specify the initial condition $\boldsymbol{r}(T)$.



Successful applications

- Low-order dynamical models, Lorenz 63, Lorenz 96
- Kuramoto–Sivashinsky equation
- 2D turbulent Rayleigh–Bénard convection



Pathak *et al.*, Chaos **27**, 121102 (2017)

Moehlis–Faisst–Eckhardt model

The model is obtained by Galerkin projection[4]:

$$\boldsymbol{u}(\boldsymbol{x},t) = \sum_{j=1}^{9} a_j(t)\boldsymbol{u}_j(\boldsymbol{x}).$$

9-dimensional system of ODEs:

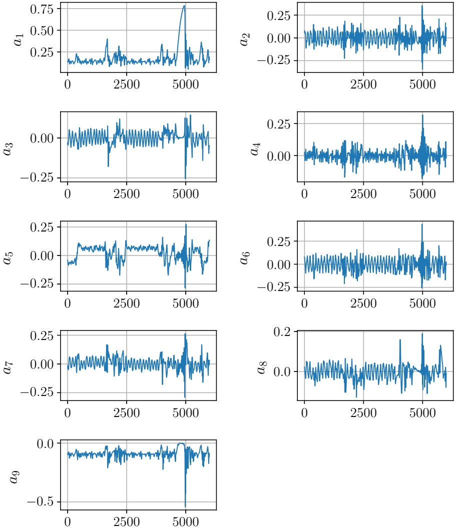
$$\frac{d}{dt}\boldsymbol{a} = \boldsymbol{f}(\boldsymbol{a}; Re, \Gamma_x, \Gamma_z),$$

where $\boldsymbol{a}(t) = [a_1(t), \ldots, a_9(t)]^T$.

Parameters:

- Domain wavelengths: $\Gamma_x = 1.75\pi, \Gamma_z = 1.2\pi$
- Reynolds number: $Re \in [200; 500]$

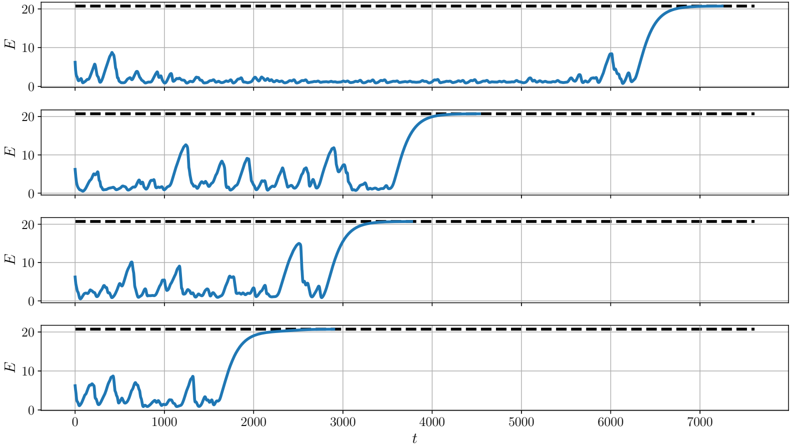
Sustained turbulence: $Re \gtrsim 320$



[4] Moehlis *et al.*, New J. Phys., **6** 56 (2004)

Laminarization ($Re = 300$)

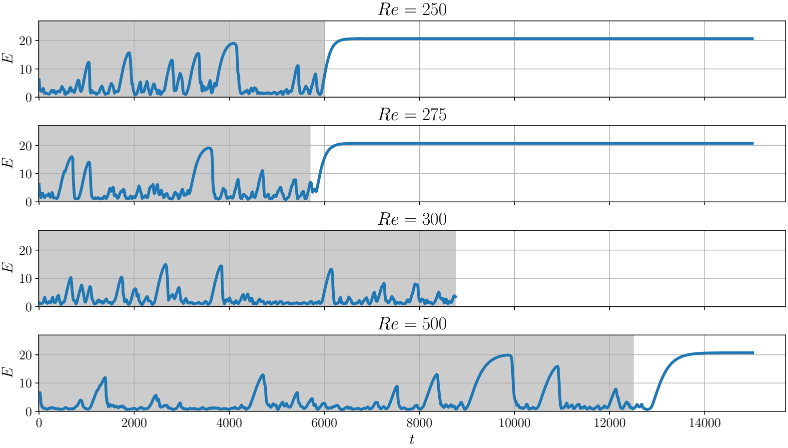
- Turbulence in shear flows is a “leaky” attractor[5]
- As a result, all trajectories eventually end up with laminarization



[5] Avila *et al.*, Science **333**, 6039 (2011)

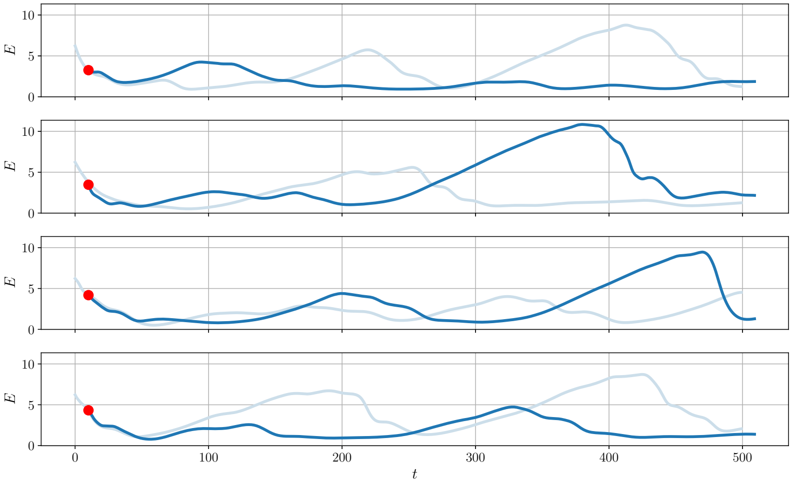
Trajectories used for training

For training, we consider only turbulent trajectories without laminarization events



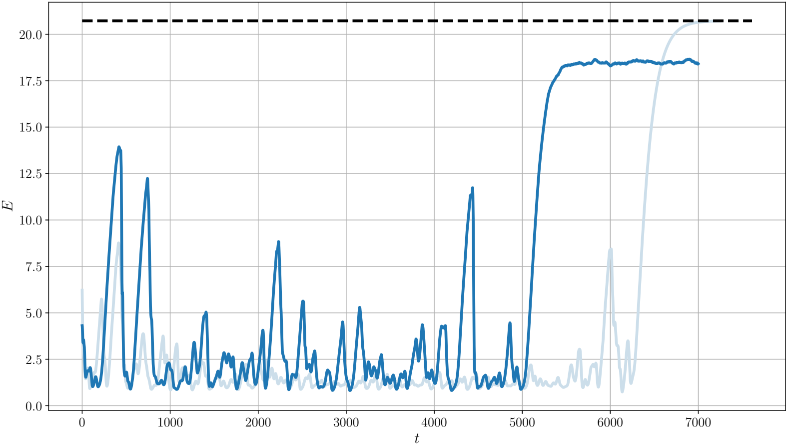
Short-term prediction ($Re = 300$)

Due to the chaotic nature of the original model, the ESN skill for short-term prediction is limited



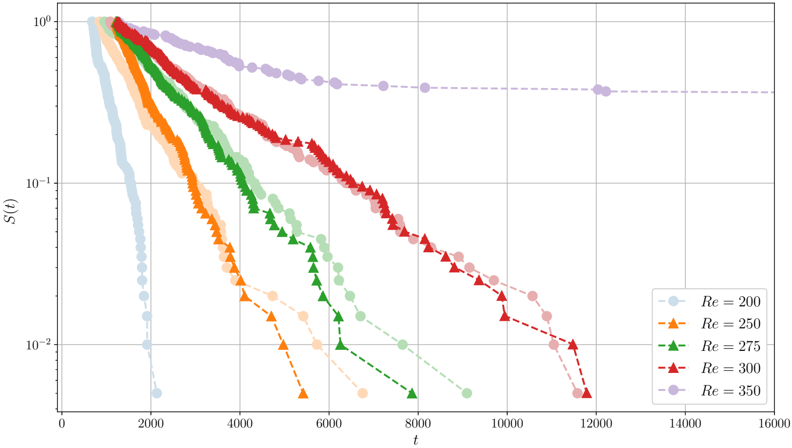
Long-term prediction ($Re = 300$)

- ESN is able to “learn” the laminarization dynamics without experiencing laminarization during the training
- Moreover, ESN is able to replicate the laminar solution



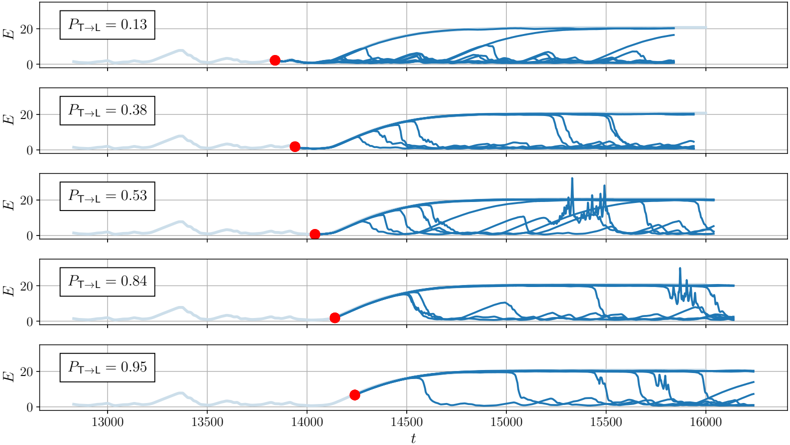
Lifetime distribution

- ESN can successfully replicate the lifetime statistics
- Its skill may degrade depending on the time series used for training



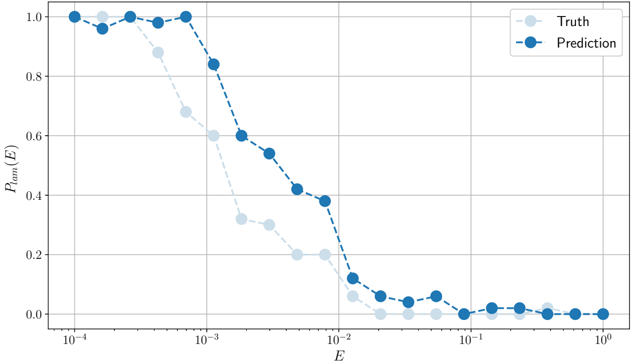
Turbulent-to-laminar transition ($Re = 500$)

- Ensemble approach can be used to estimate the probability of turbulent-to-laminar transition
- The probability grows as the initial condition gets closer to the laminarization event



Laminar-to-turbulent transition ($Re = 500$)

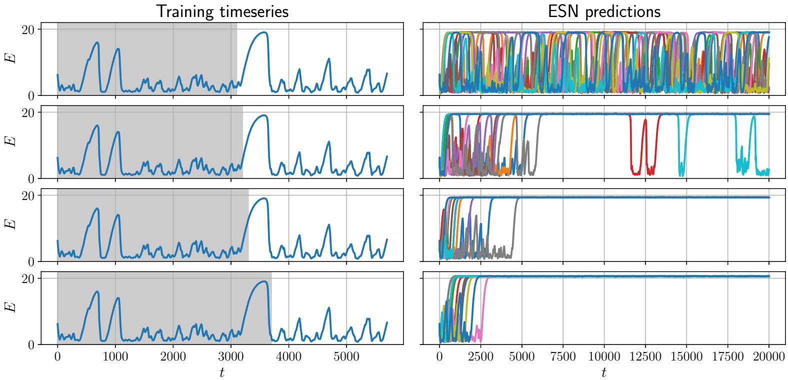
- Robustness of the laminar state to finite-amplitude perturbations is important for assessing laminar-to-turbulent transition
- **Laminarization probability** is the probability that a random perturbation decays as a function of its kinetic energy[6]
- ESN can successfully replicate the laminarization probability



[6] Pershin, Beaume, Tobias, J. Fluid Mech. **895**, A16 (2020)

How can ESN learn laminar dynamics?

- ESN can be expected to embed attractors of the true system by Echo State Network Approximation Theorem [7]
- In practice, it is important to guarantee that the training timeseries includes excursion close to the laminar state



[7] Hart *et al.*, Neural Networks **128**, 234–247 (2020)

Conclusion

Laminarization probability:

- Helps analyse finite-amplitude instabilities
- Approximates the size of the basin of attraction
- Allows to quantify and compare the efficiency of control strategies
- Bayesian inference provides an efficient framework for the laminarization probability estimation
- Minimal seeds and edge states may be misleading for control design

Pershin, Beaume and Tobias, J. Fluid Mech. **895**, A16 (2020)
Pershin, Beaume and Tobias, *submitted, arXiv:2108.07629* (2021)

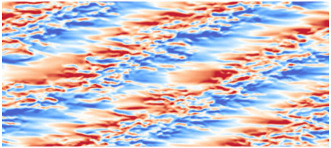
Echo State Networks for transition to turbulence:

- Build a surrogate model of a shear flow
- Able to learn laminar dynamics without seeing it during the training
- Able to approximate key statistics of a transitional flow based only on a single turbulent trajectory
- Perspective: they can be used for designing optimal control

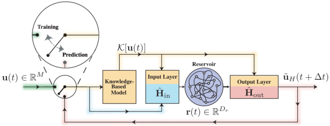
Pershin, Beaume, Li and Tobias, *in preparation* (2021)

Future work

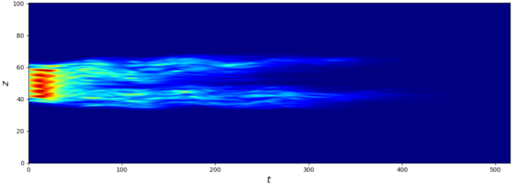
(a) **Spatially extended flows?**[8]



(b) **Physics-informed ESN?**[9]



(c) **Optimal control using reservoir computing?**



[8] Chantry *et al.*, J. Fluid Mech. **791**, R8 (2016)
[9] Pathak *et al.*, Chaos **28**, 041101 (2018)