Symmetry reduction for low-dimensional models of self-similar fluid flows

Clancy Rowley

22 July 2008
Oberwolfach, Germany

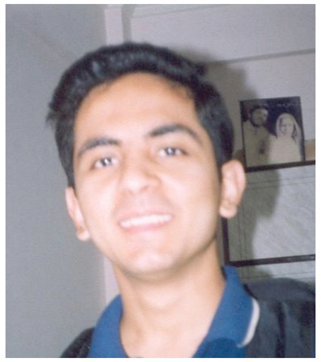
Mechanical and Aerospace Engineering
PRINCETON

Acknowledgements

- Acknowledgments
  - Mingjun Wei (free shear flow)
    now at NMSU

  - Milos Ilak (channel flow)
    graduate student

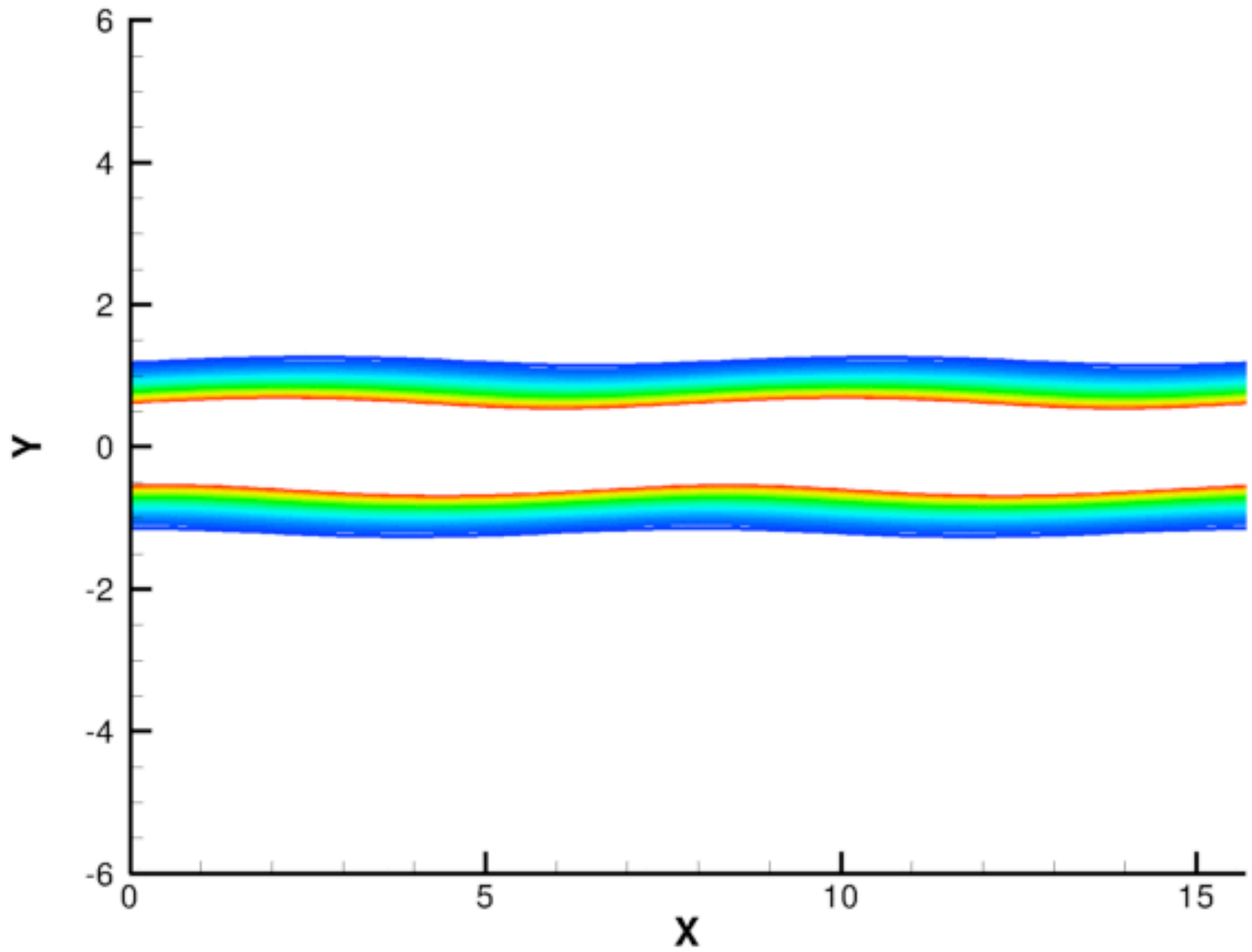
  - Sunil Ahuja (control)
    graduate student



  - Collaborators
    - Yannis Kevrekidis (Princeton)
    - Jerry Marsden

Example: free shear layer

- Challenge: find a low-dimensional model of a temporally evolving free shear layer
- Navier-Stokes equations in a domain periodic in x, infinite in y
- Phenomena: exponential growth, nonlinear saturation, pairing, decay to self-similar solution



Exact self-similar solution

- Exact solution well known

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

$$u + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{\partial p}{\partial x} + \frac{1}{Re}\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)$$

$$v + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} = -\frac{\partial p}{\partial y} + \frac{1}{Re}\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right),$$

Periodic (constant) in x
Infinite in y

Similarity variable

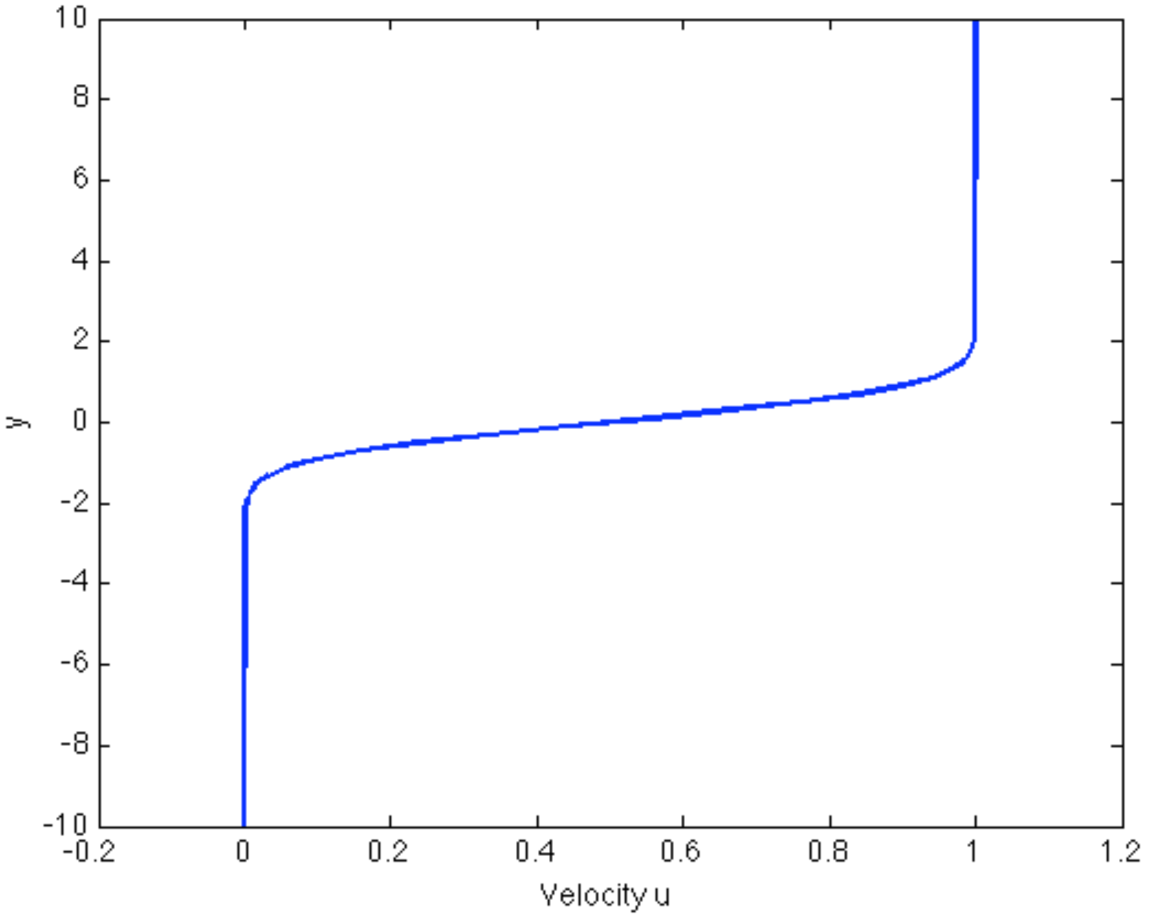
$$\eta = y\left(\frac{Re}{4(t-t_0)}\right)^{1/2}$$

Solution

$$u(\eta) = U_1 + \frac{U_2 - U_1}{2}\,\mathrm{erfc}(-\eta)$$

$$v(\eta) = 0$$

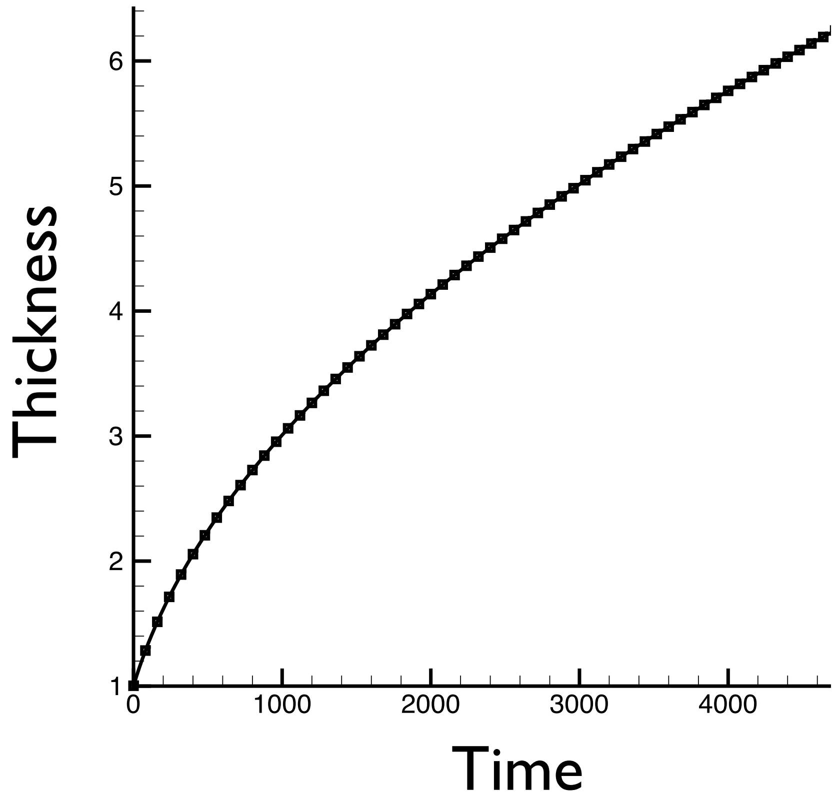
[Landau & Lifschitz]



Main idea

- Seek an analogous change of coordinates that simplifies the solution for more complex dynamics

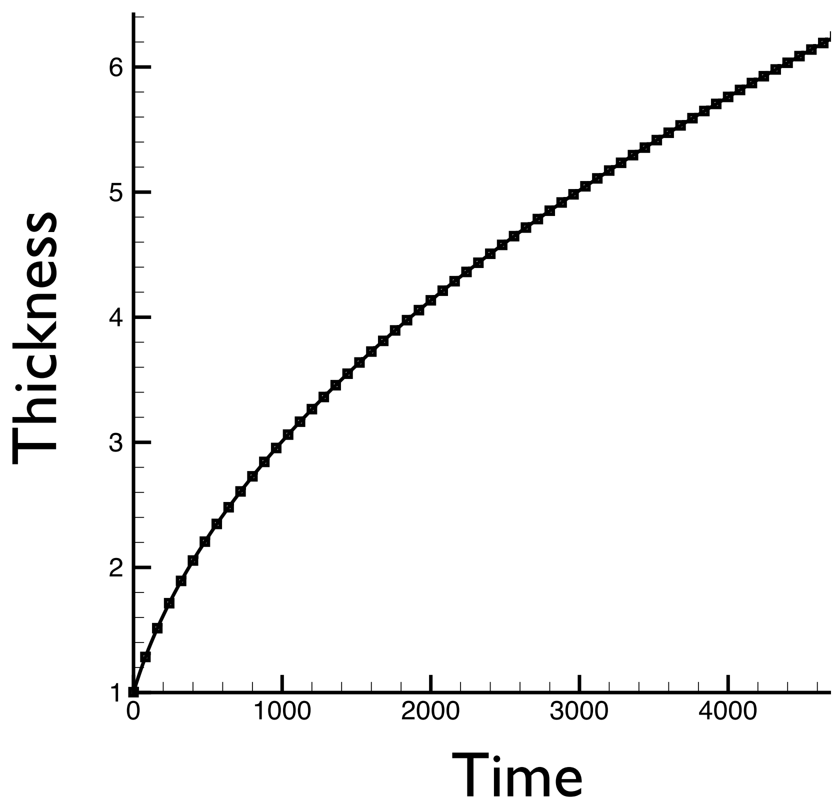
Self-similar solution



Main idea

- Seek an analogous change of coordinates that simplifies the solution for more complex dynamics

Self-similar solution

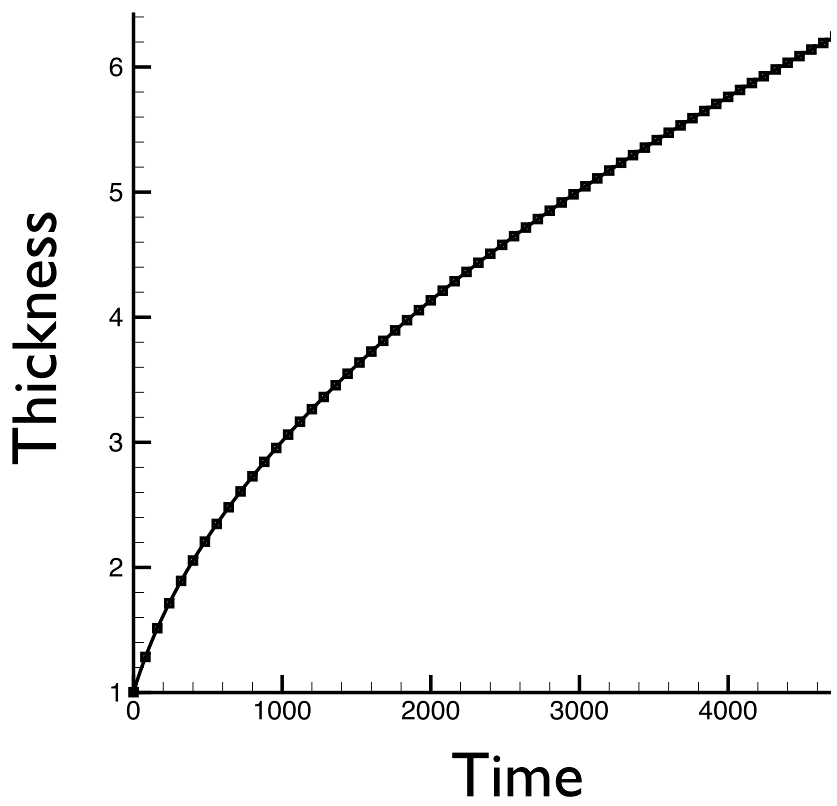


$$\eta = y\left(\frac{Re}{4(t-t_0)}\right)^{1/2}$$

Main idea

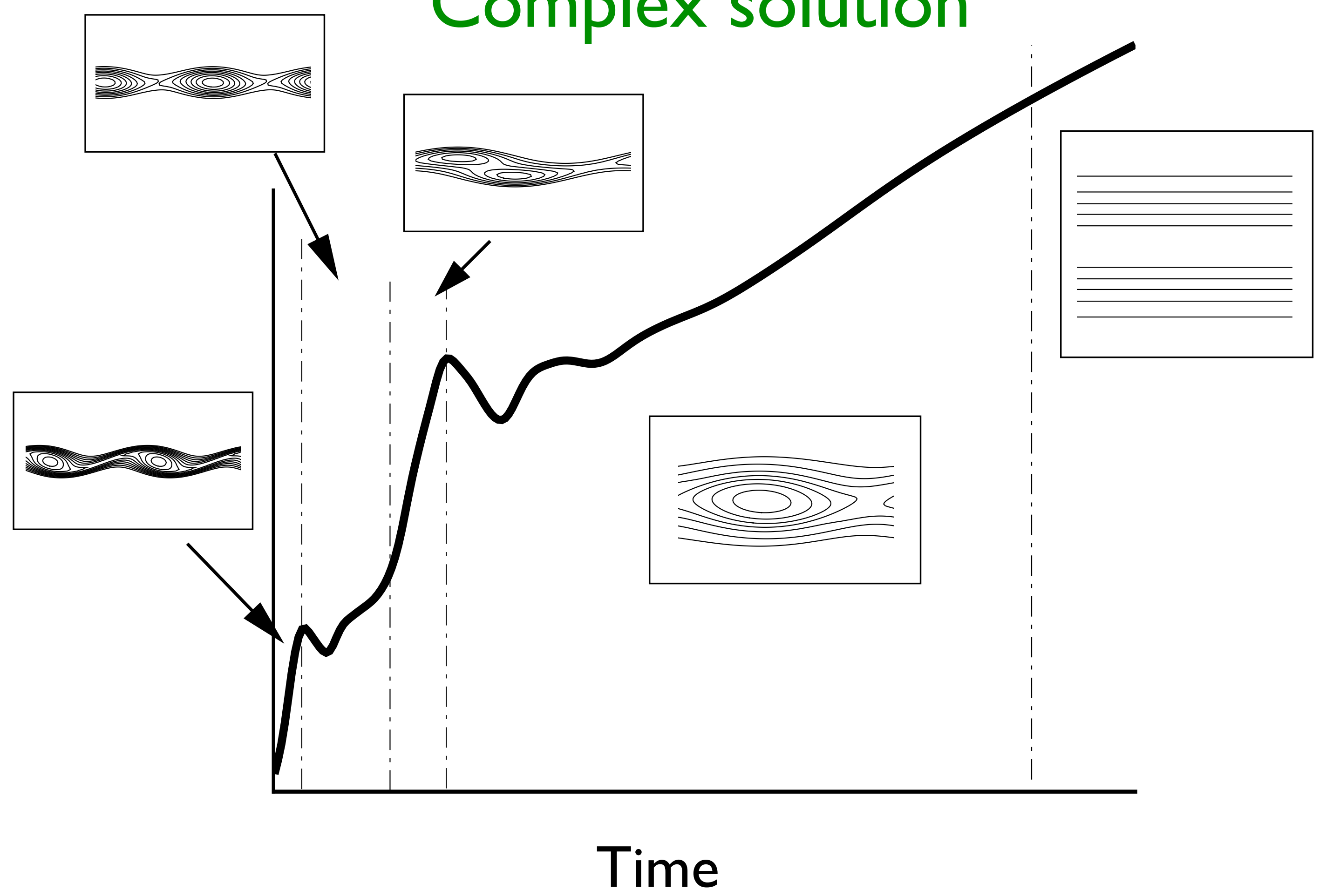
- Seek an analogous change of coordinates that simplifies the solution for more complex dynamics

Self-similar solution



$$\eta = y\left(\frac{Re}{4(t-t_0)}\right)^{1/2}$$

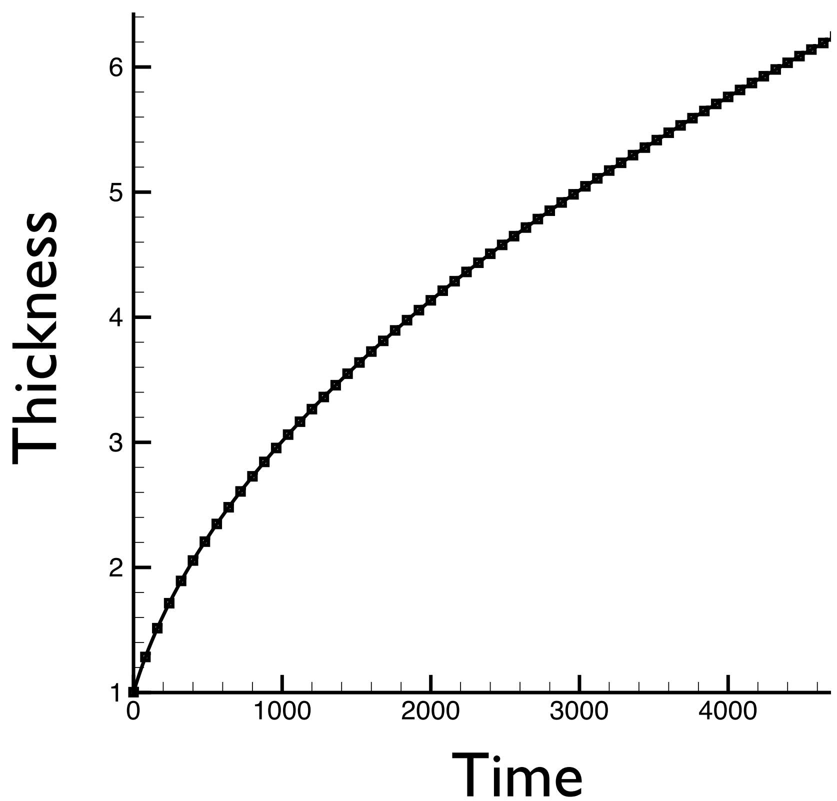
Complex solution



Main idea

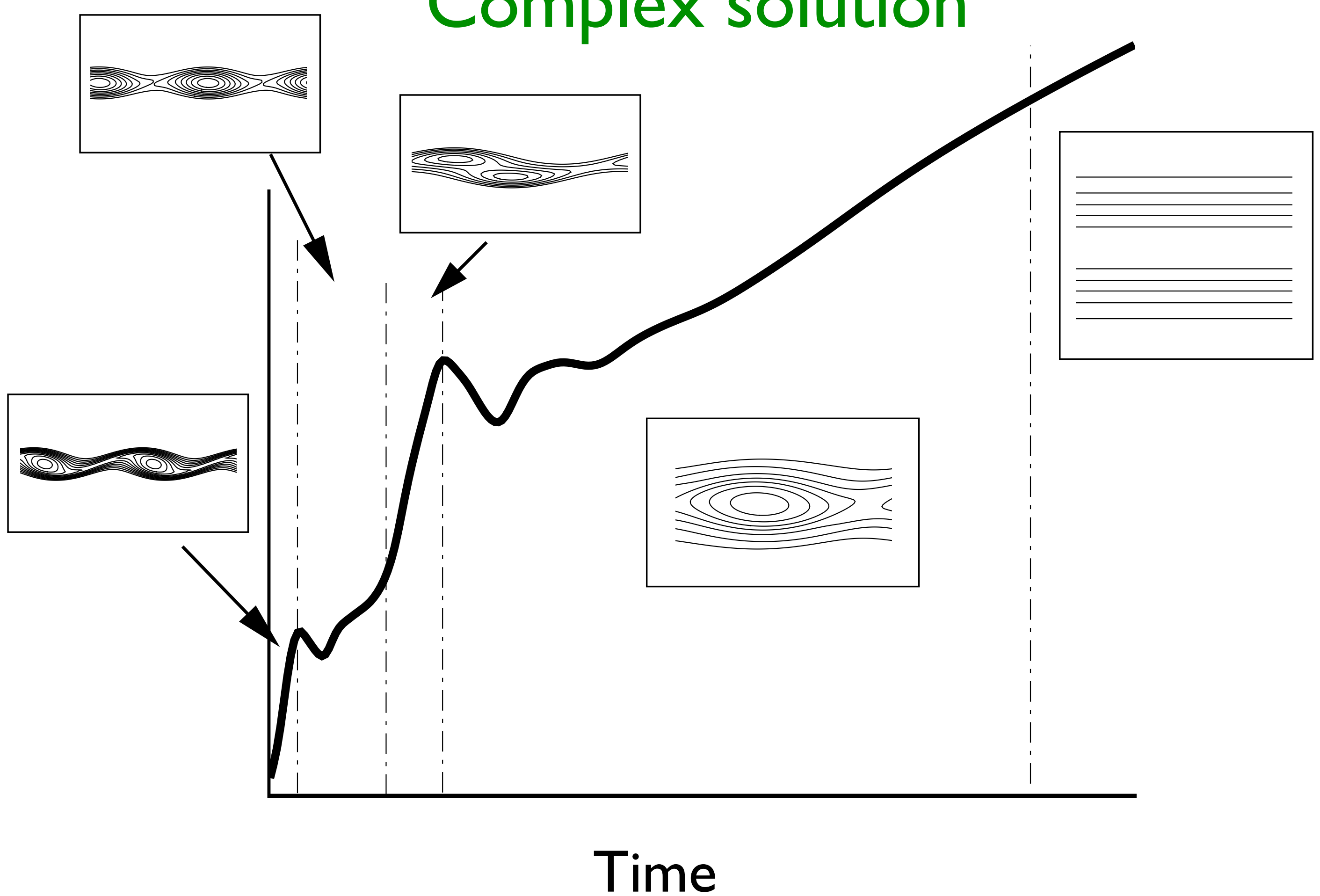
- Seek an analogous change of coordinates that simplifies the solution for more complex dynamics

Self-similar solution



$$\eta = y\left(\frac{Re}{4(t-t_0)}\right)^{1/2}$$

Complex solution

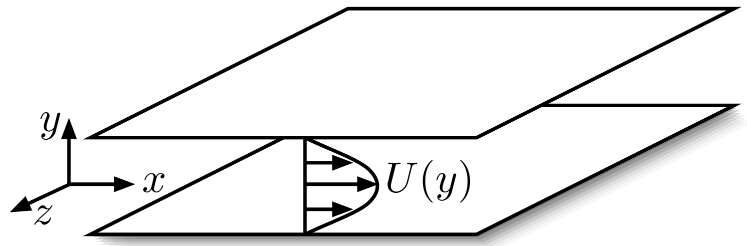


$$\eta = y \cdot g(t)$$

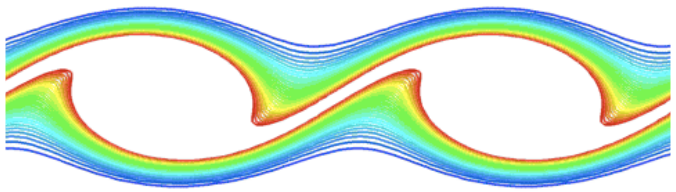
Find g(t) using symmetry reduction

Outline

- Model reduction
  - Galerkin projection
  - Proper Orthogonal Decomposition
  - Balanced truncation
  - Example: linearized channel flow



- Symmetry reduction using template fitting
  - Representing the quotient space using slices
  - Dynamics on the slice
  - Reconstruction equation
  - Example: free shear layer

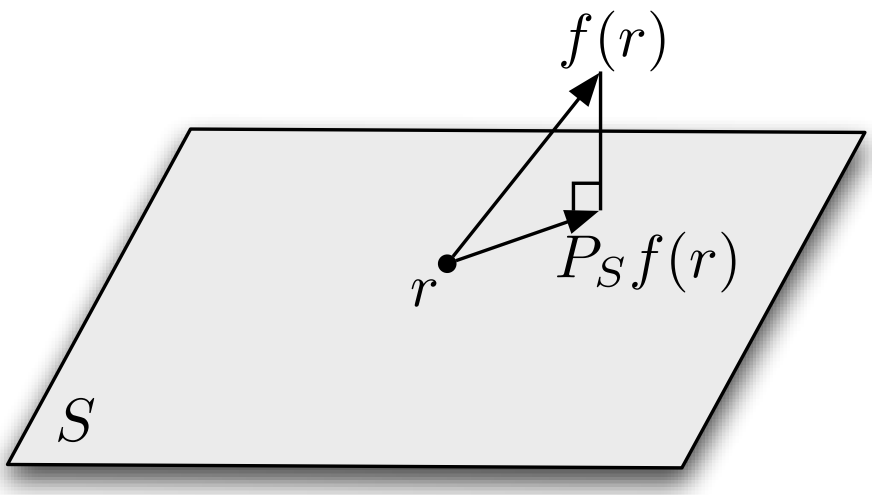


Galerkin projection

- Dynamics evolve on a high-dimensional space (or infinite-dim'l)
- Project dynamics onto a low-dimensional subspace S

$$\dot{x} = f(x) \qquad x \in V$$

$$r \in S \subset V$$



- Define dynamics on the subspace by

$$\dot{r} = P_S f(r) \qquad P_S : V \to S \ \text{ is a projection}$$

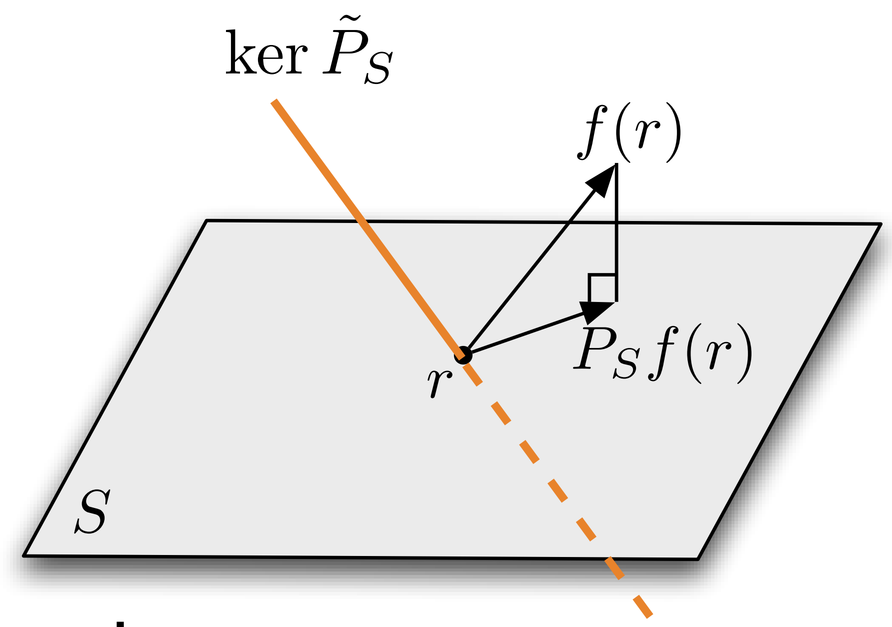
- Two choices:
  - choice of subspace
  - choice of inner product
    (equivalently, choice of the nullspace for a non-orthogonal projection)

Galerkin projection

- Dynamics evolve on a high-dimensional space (or infinite-dim'l)
- Project dynamics onto a low-dimensional subspace S

$$\dot{x} = f(x) \qquad x \in V$$

$$r \in S \subset V$$



- Define dynamics on the subspace by

$$\dot{r} = P_S f(r) \qquad P_S : V \to S \text{ is a projection}$$

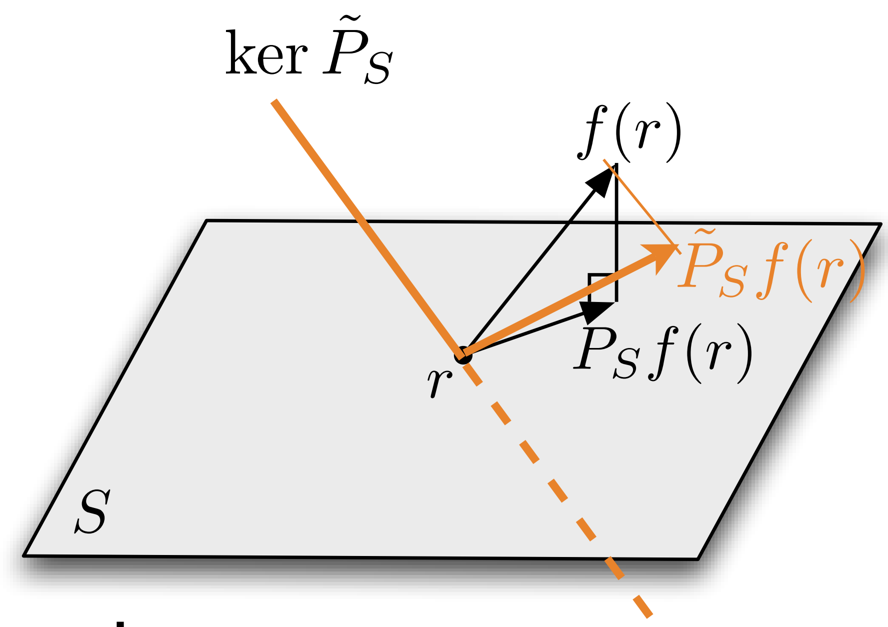
- Two choices:
  - choice of subspace
  - choice of inner product
    (equivalently, choice of the nullspace for a non-orthogonal projection)

Galerkin projection

- Dynamics evolve on a high-dimensional space (or infinite-dim'l)
- Project dynamics onto a low-dimensional subspace S

$$\dot{x} = f(x) \qquad x \in V$$

$$r \in S \subset V$$



- Define dynamics on the subspace by

$$\dot{r} = P_S f(r) \qquad P_S : V \to S$$ is a projection

- Two choices:
  - choice of subspace
  - choice of inner product
    (equivalently, choice of the nullspace for a non-orthogonal projection)

Proper Orthogonal Decomposition (POD)

- Obtain "optimal" basis for the subspace, from data
  - Gather data, as "snapshots" u(x,t) from simulations or experiments
  - Determine orthonormal basis functions that optimally span the data:

$$P_n u(x,t) = \sum_{j=1}^{n} a_j(t)\varphi_j(x)$$

  - Minimize $$\int_0^T \|u(t) - P_n u(t)\|^2 \, dt$$

$$\varphi_j \in V$$

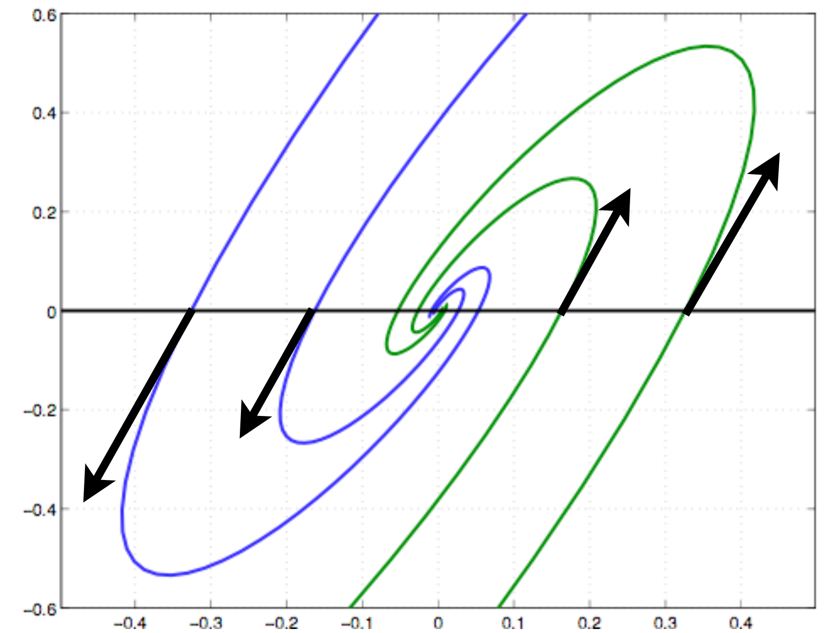
POD modes

$$S = \mathrm{span}\{\varphi_j\}$$

  - Solution: SVD of the matrix of snapshots
- Limitations
  - Optimal for capturing a given dataset, not necessarily dynamics
  - Low-energy modes may be important to the dynamics

Energy-based inner products

- Reduced-order models can behave unpredictably
  - Can even change stability type of equilibria
    [Rempfer, Thoret. CFD 2000]
  - Simple example: consider the system:
    $$\frac{d}{dt}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 3 & -2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$
  - Sink at the origin
  - Projection onto x_1 axis is
    $$\dot{x}_1 = x_1 \quad \text{unstable}$$



  - Can at least fix this simple problem by changing the inner product used for the projection
    - Cute result: If an orthogonal projection is used with an "energy-based" inner product, this will ensure stability of the origin
    - Note: does not guarantee stability preserved for other equilibrium points, periodic orbits, etc.

[Rowley, T Colonius, RM Murray, Phys D 2004]

Energy-based inner products

- Consider a system with a stable equilibrium point at the origin:

$$\dot{x} = f(x) \qquad f(0) = 0 \qquad x \in \mathbb{R}^n$$

- Consider an inner product whose induced norm is a Liapunov function ("energy-based"):

$$\langle x, y \rangle = x^T Q y, \qquad Q > 0 \qquad \begin{array}{l} V(x) = x^T Q x \text{ is a Liapunov function} \\ \dot{V}(x) = 2x^T f(x) \leq 0, \qquad \forall x \in U \end{array}$$

- Reduced-order dynamics given by orthogonal projection

$$r = Px \qquad P^2 = P$$

$$\dot{r} = Pf(r) \qquad \langle x, Py \rangle = \langle Px, y \rangle \qquad QP = P^T Q$$

- Then V is a Liapunov function for the reduced-order system:

$$\dot{V}(r) = 2r^T QPf(r) = 2r^T P^T Qf(r) = 2(Pr)^T Qf(r)$$
$$= 2r^T Qf(r) \leq 0$$

- So: if an energy-based inner product is used, the origin is stable for the reduced-order system, regardless of the subspace used for the projection

Balanced truncation

Moore (1981)

- Linear time-invariant input-output system

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

  - Compute controllability and observability Gramians

$$W_c = \int_0^\infty e^{At} B B^T e^{A^T t}\, dt \qquad W_o = \int_0^\infty e^{A^T t} C^T C e^{At}\, dt$$

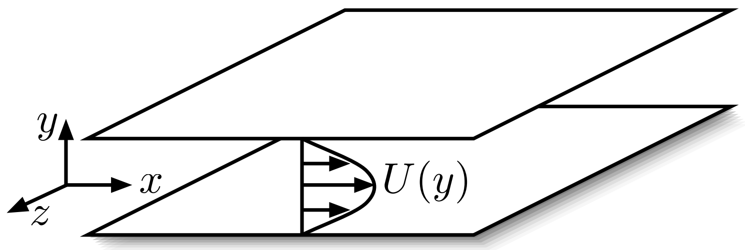
effect of input u on the state x — effect of state x on future outputs y

  - Find coordinates in which W_c and W_o are equal and diagonal
  - Truncate states that are least controllable/observable
  - Get a priori error bounds on the reduced order model, close to the best possible from any model reduction method
  - Can show that balanced truncation is equivalent to POD of impulse-state response data, using the observability Gramian as an inner product

CWR, *Int J Bif Chaos* (2005)

Example: linearized channel flow

- Linearized channel flow



  - Periodic boundary conditions in x and z
  - Linearize Navier-Stokes about a steady laminar solution $U(y) = 1 - y^2$

Orr-Sommerfeld/Squire system

$$\frac{\partial}{\partial t}\begin{bmatrix} -\Delta & 0 \\ 0 & I \end{bmatrix}\begin{bmatrix} v \\ \eta \end{bmatrix} = \begin{bmatrix} L_{OS} & 0 \\ -U'\partial_z & L_{SQ} \end{bmatrix}\begin{bmatrix} v \\ \eta \end{bmatrix}$$

$$v(\pm 1) = v_y(\pm 1) = 0$$
$$\eta(\pm 1) = 0$$

$$L_{OS} = U\partial_x\Delta - U''\partial_x - \frac{1}{Re}\Delta^2$$

$$L_{SQ} = -U\partial_x + \frac{1}{Re}\Delta$$

$v =$ wall-normal velocity

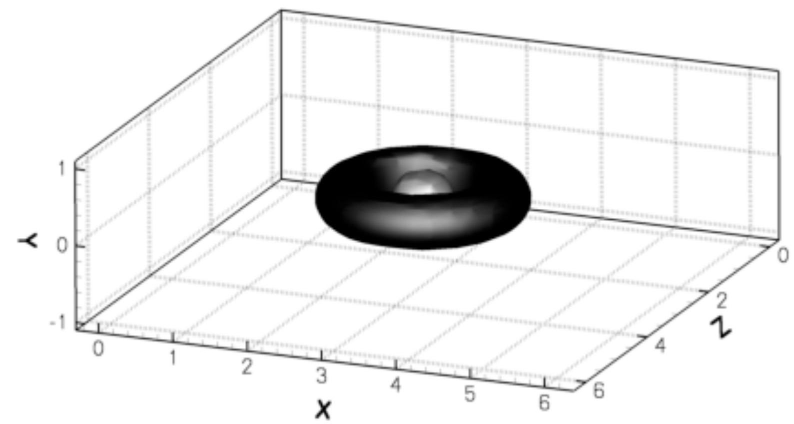
$\eta =$ wall-normal vorticity

- Very well-studied system:
  - Non-normality, large transient growth

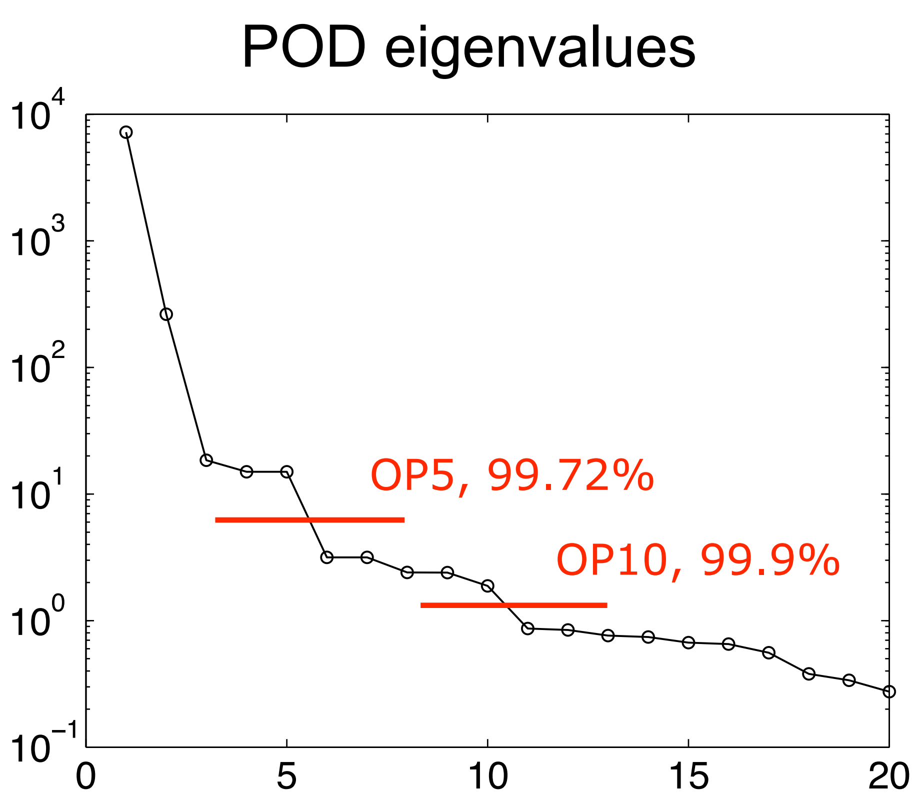
Trefethen et al [Science, 1993]
Farrell & Ioannou [96,96,01]
Schmid & Henningson [01]
Bamieh & Jovanovic [01,03]

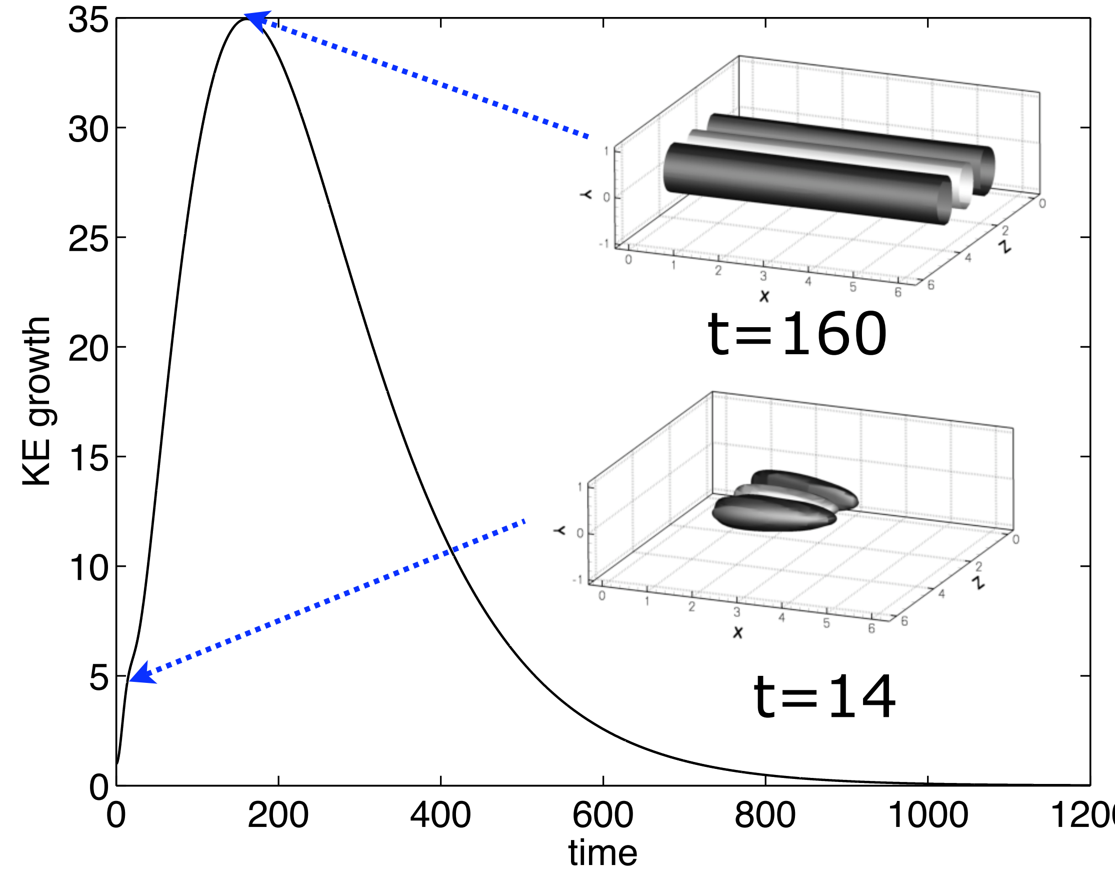
POD models of channel flow

- First 5 modes contain over 99.7% of energy
- First 10 modes contain over 99.9% of energy



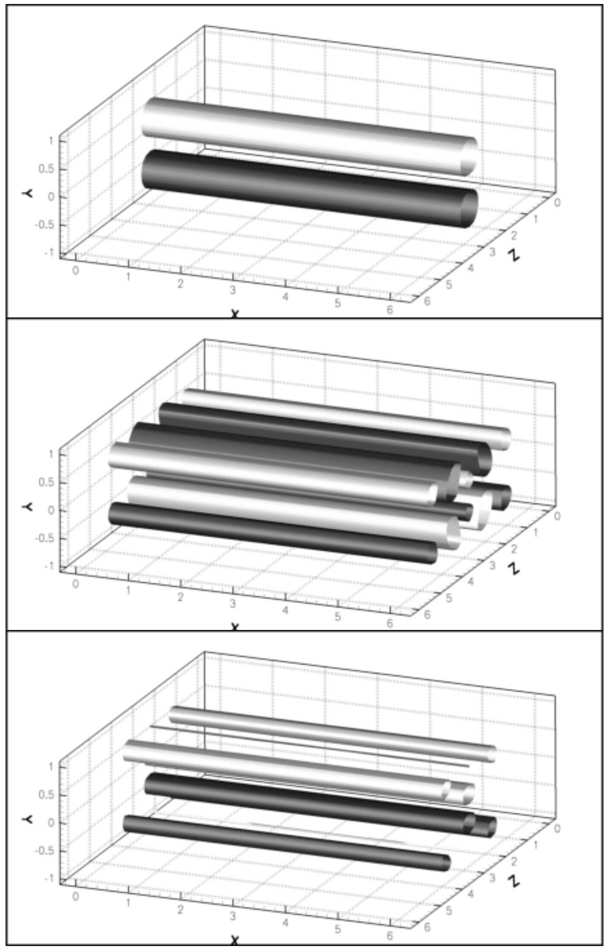
initial condition



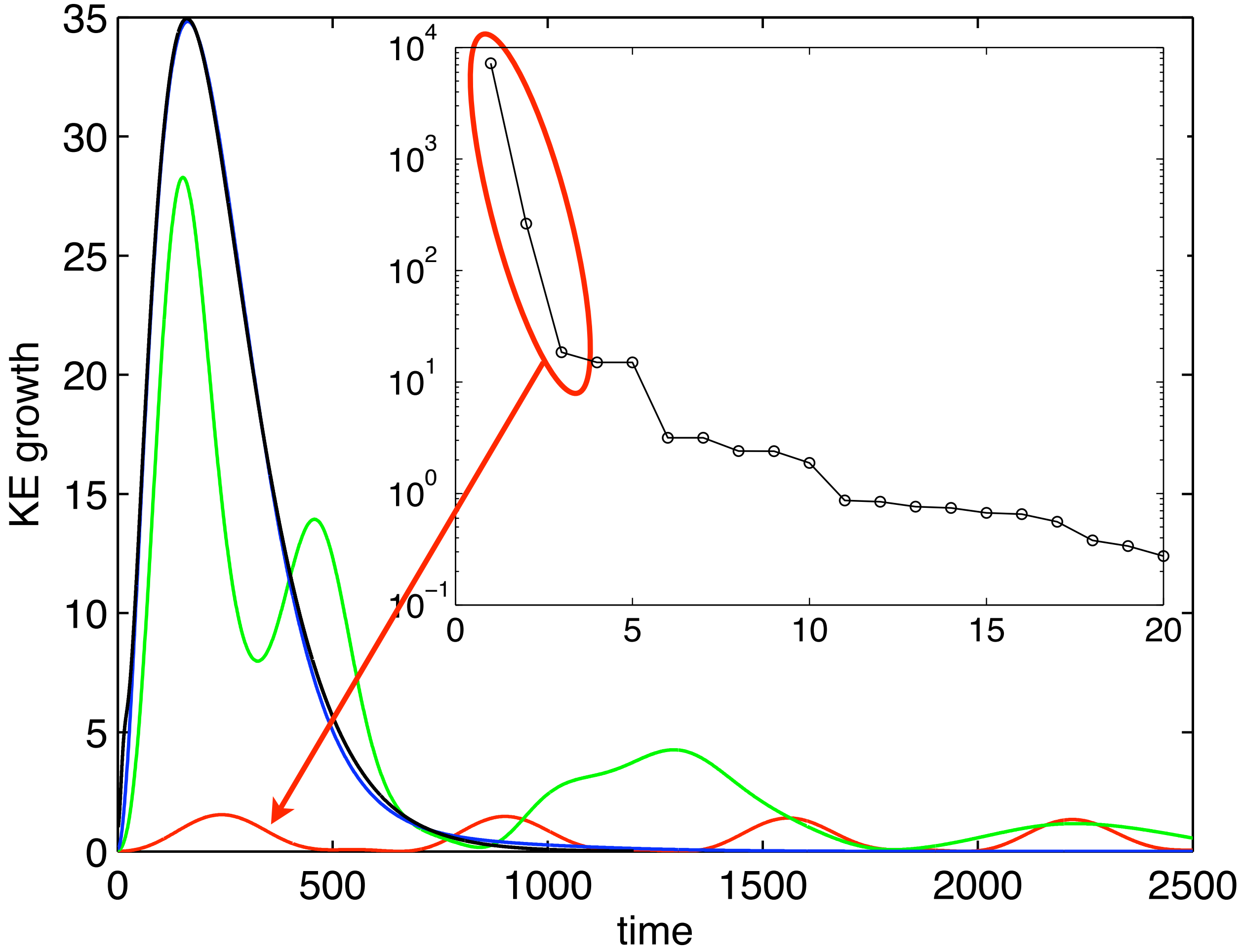


POD model performance

POD modes 1-3

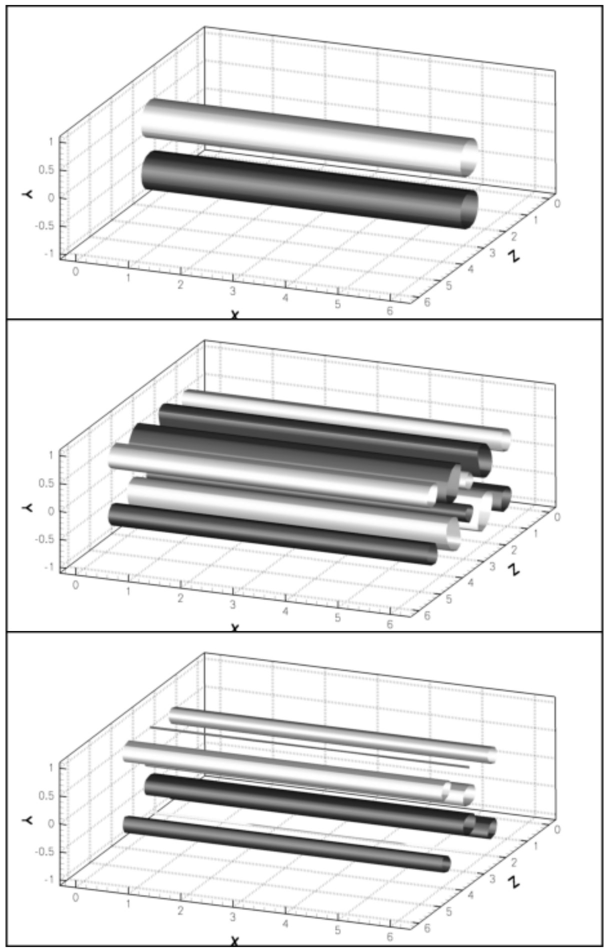


Energy growth

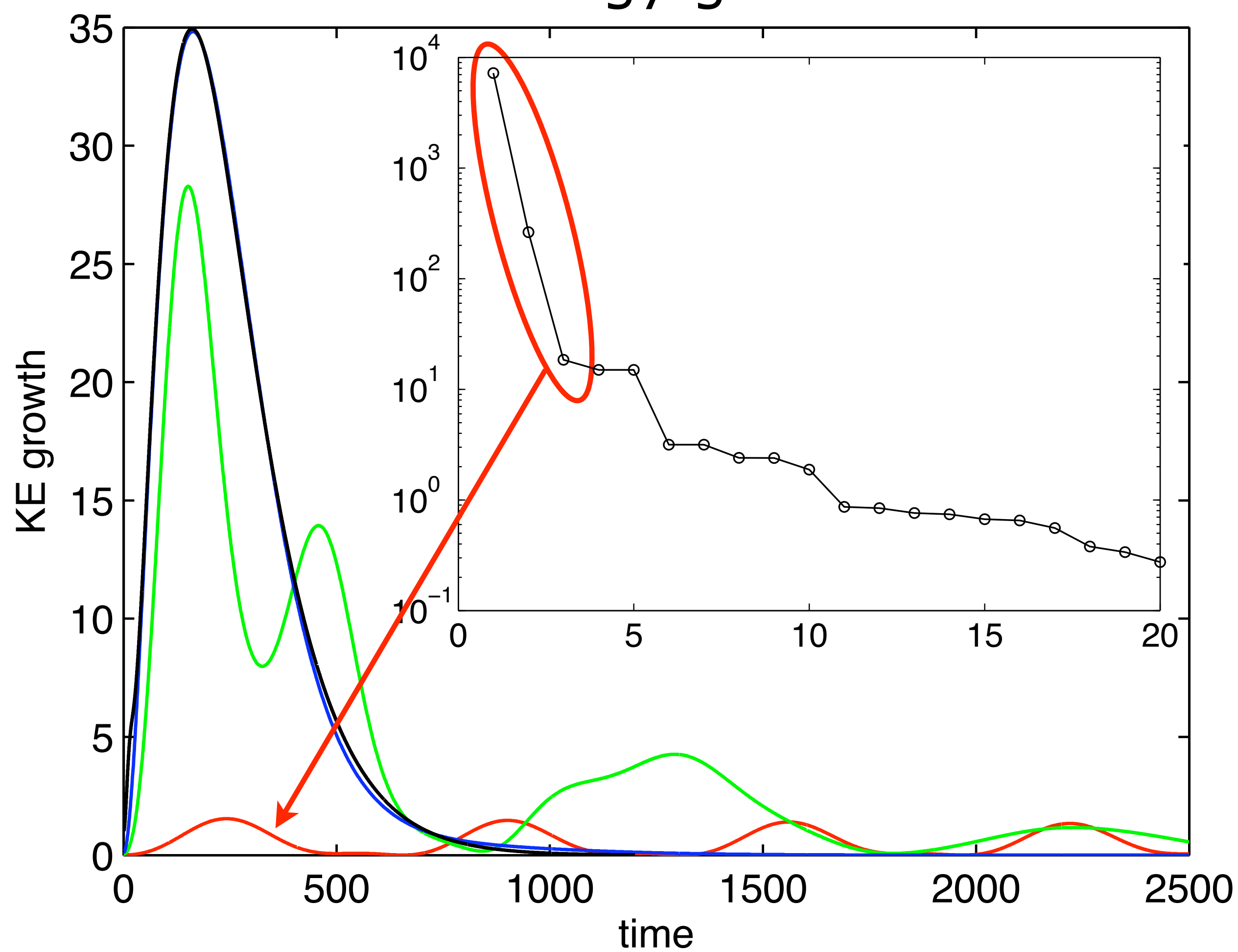


POD model performance

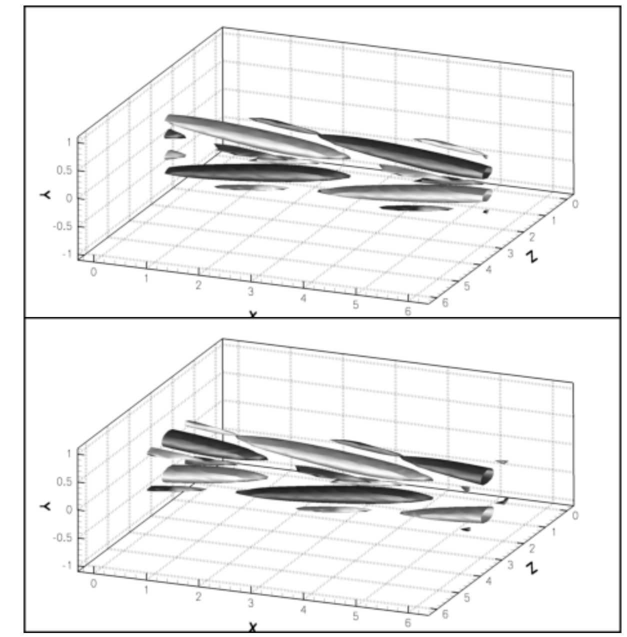
POD modes 1-3



Energy growth

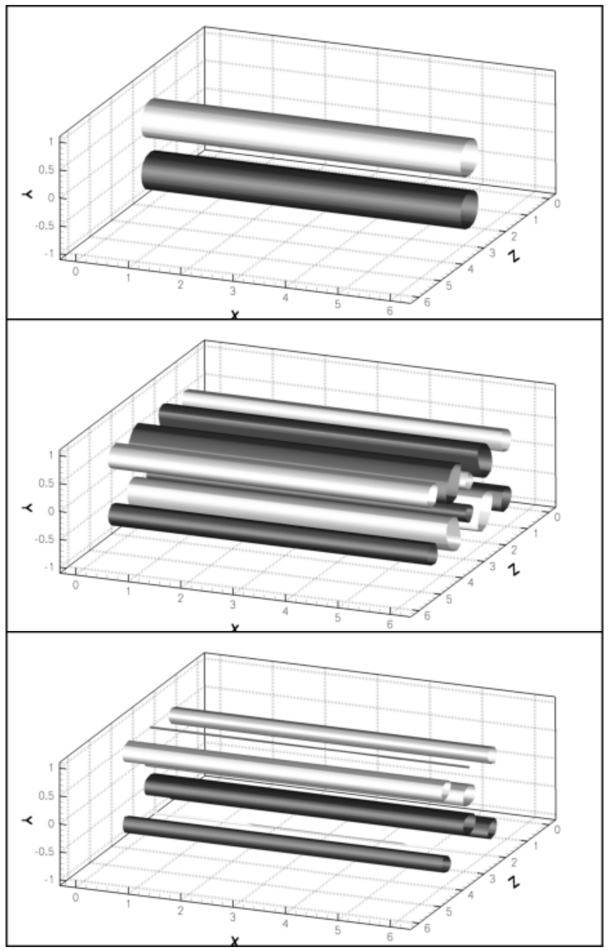


POD modes 4-5

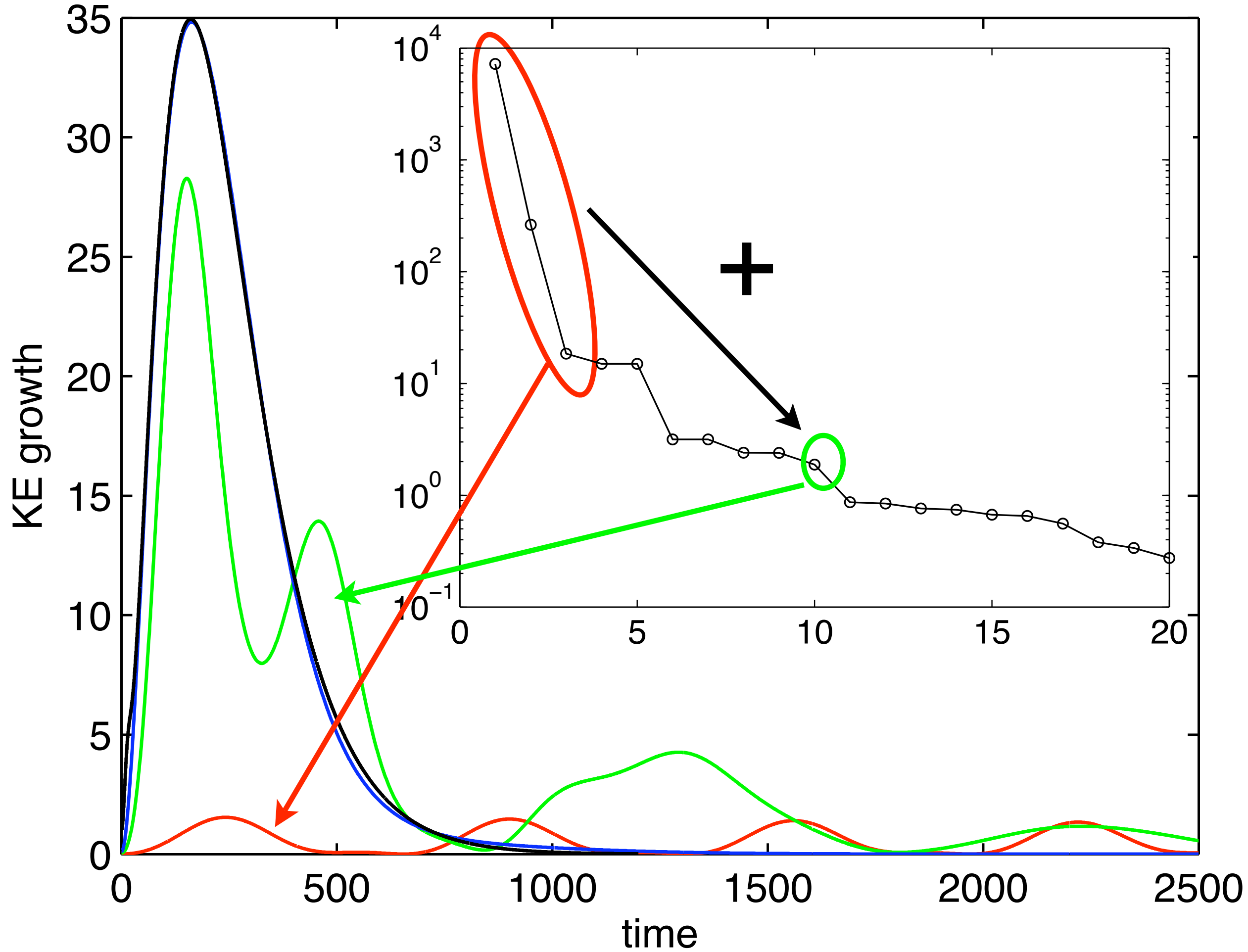


POD model performance

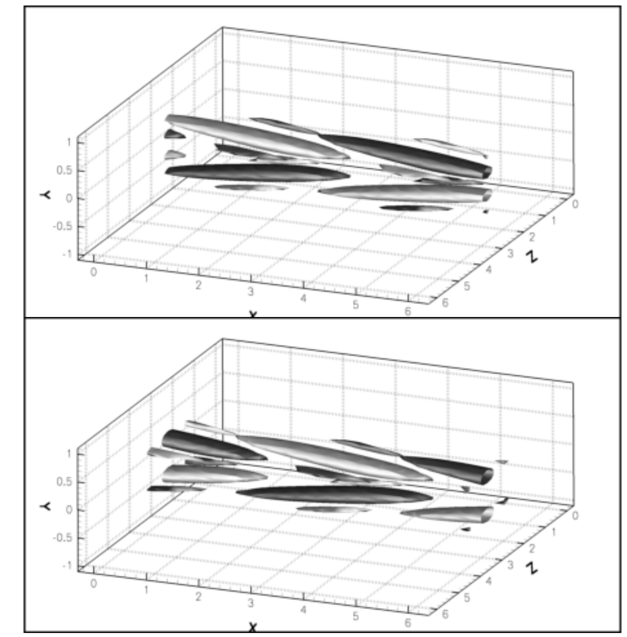
POD modes 1-3



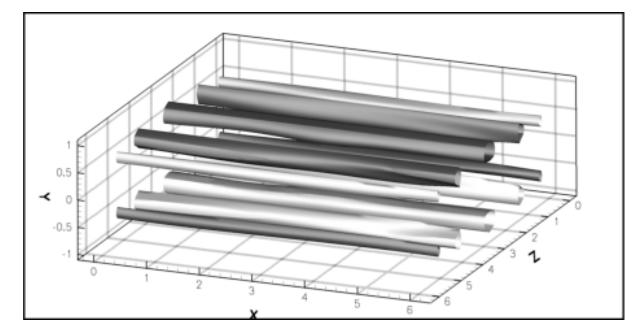
Energy growth



POD modes 4-5

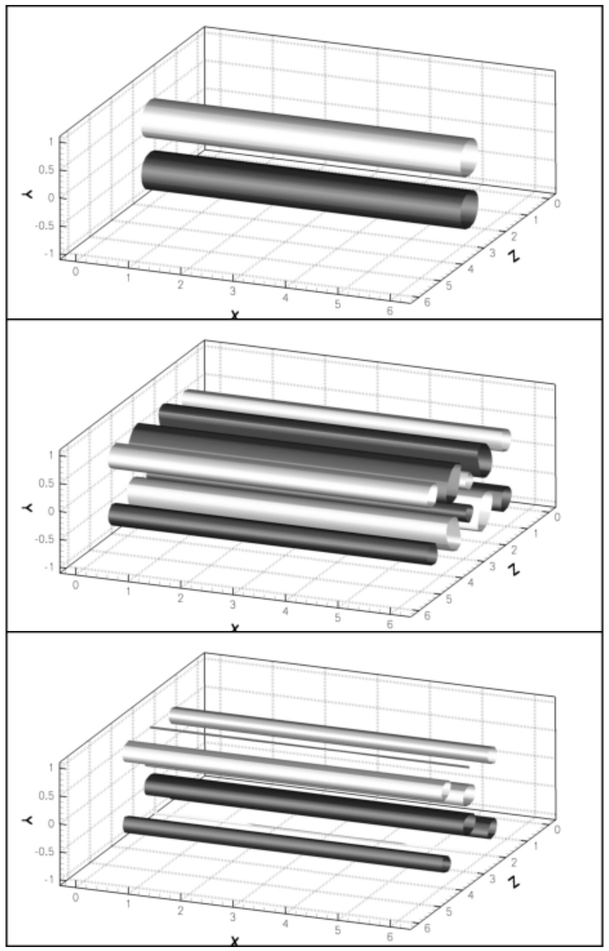


POD mode 10

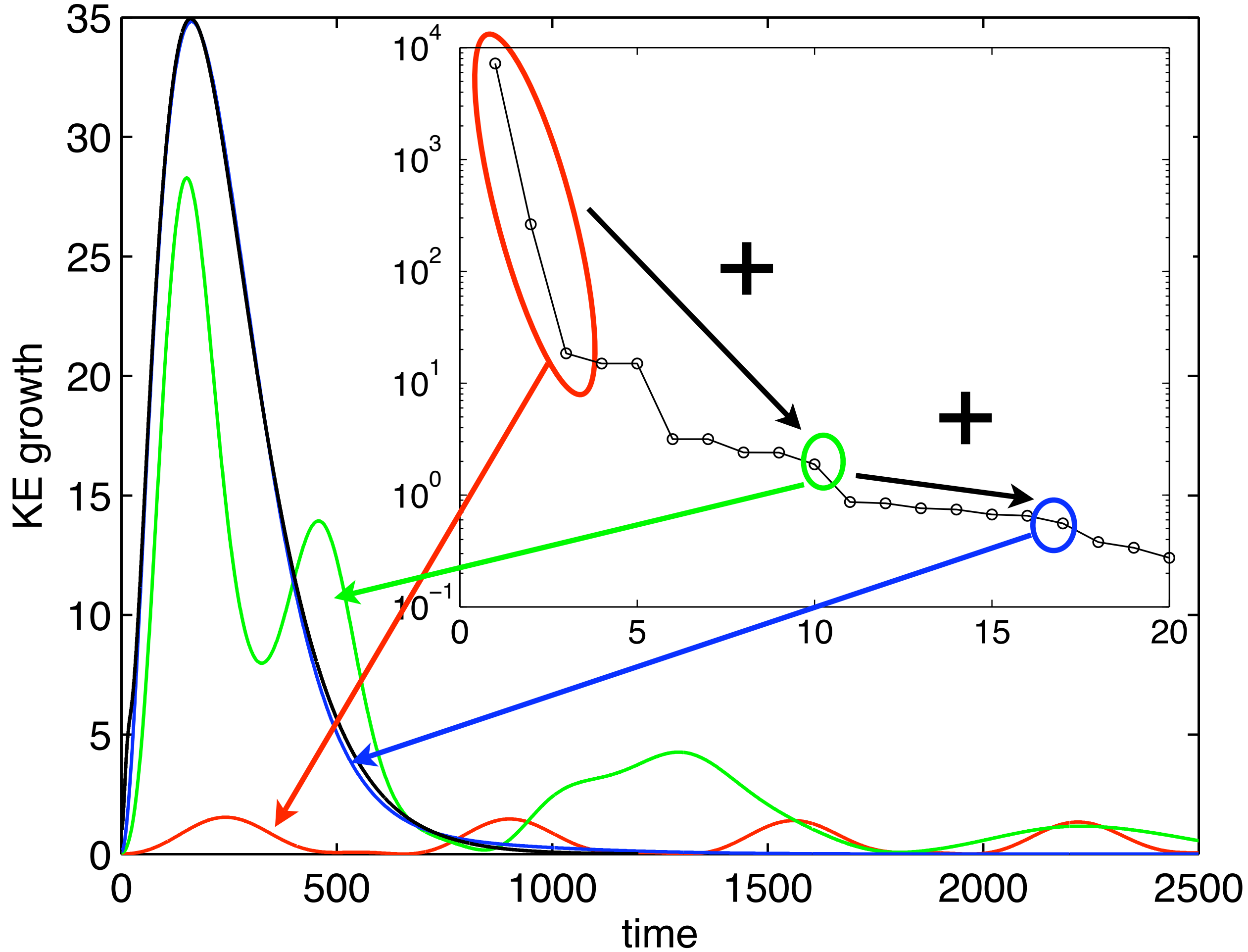


POD model performance

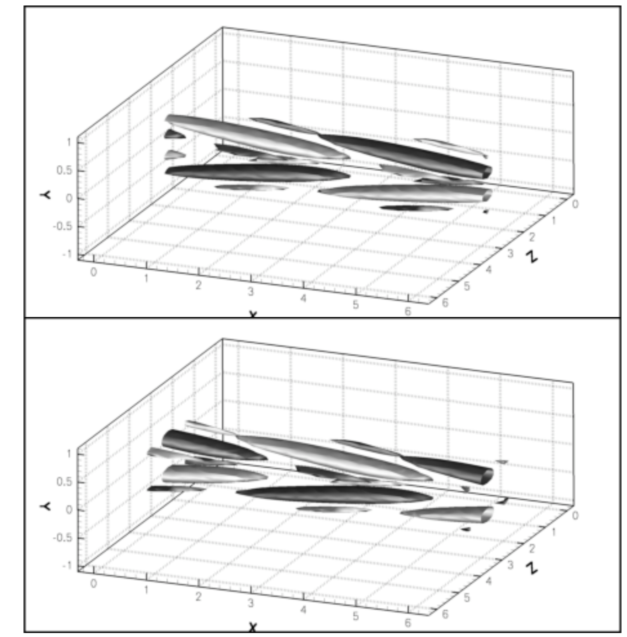
POD modes 1-3



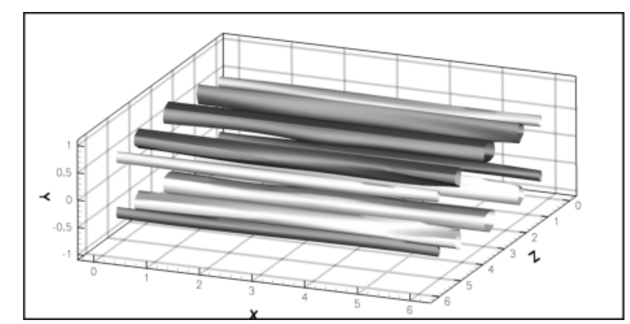
Energy growth



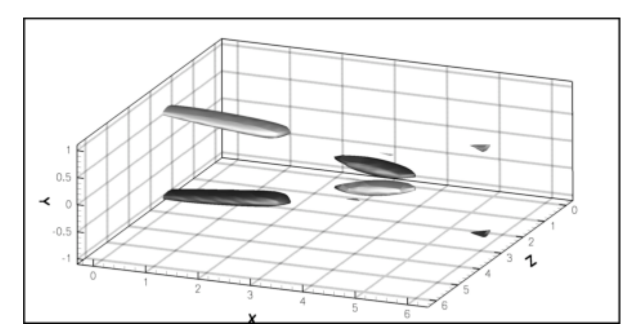
POD modes 4-5



POD mode 10

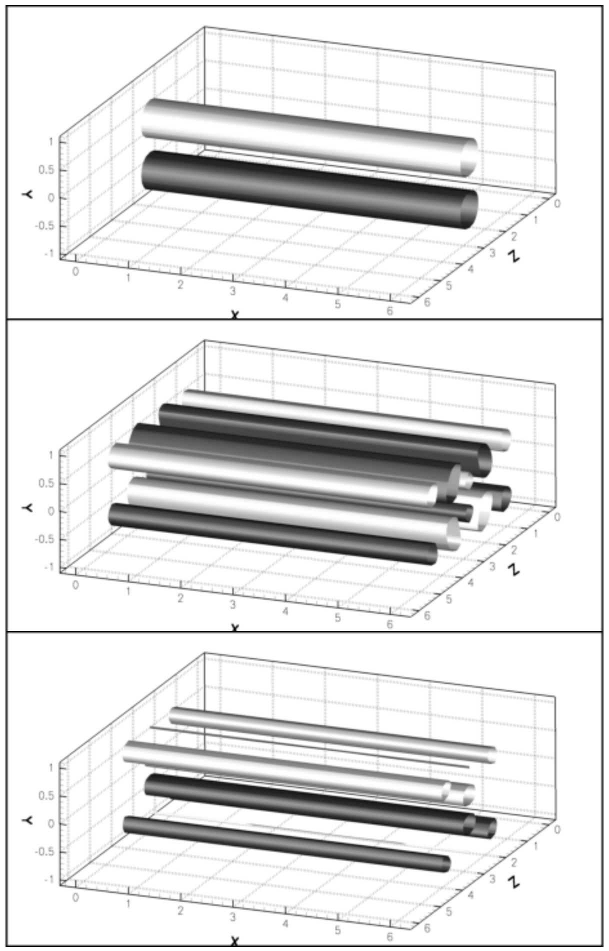


POD mode 17

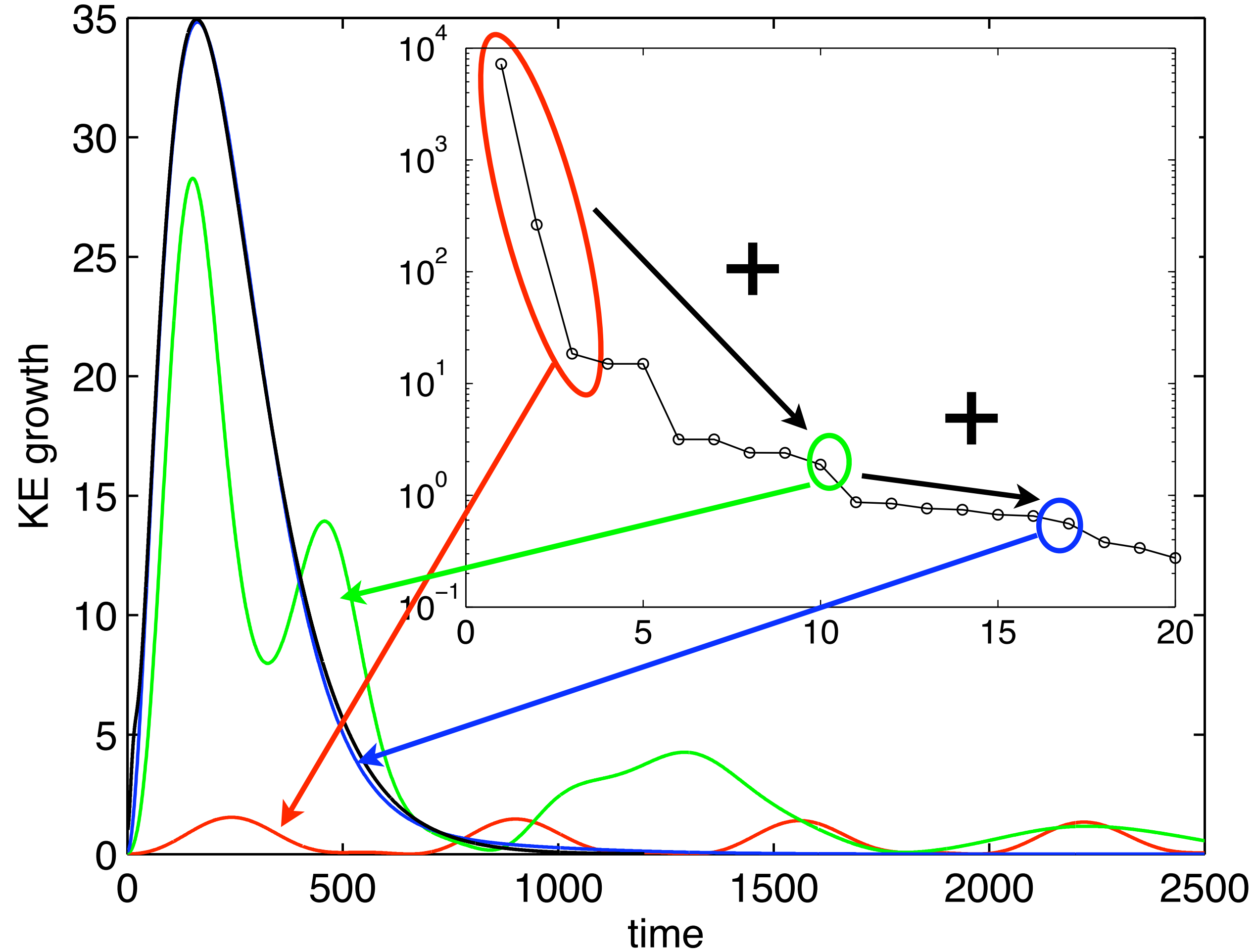


POD model performance

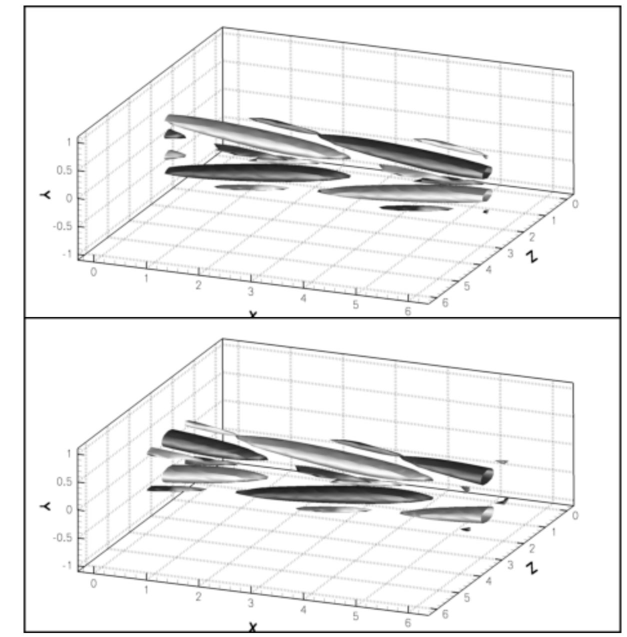
POD modes 1-3



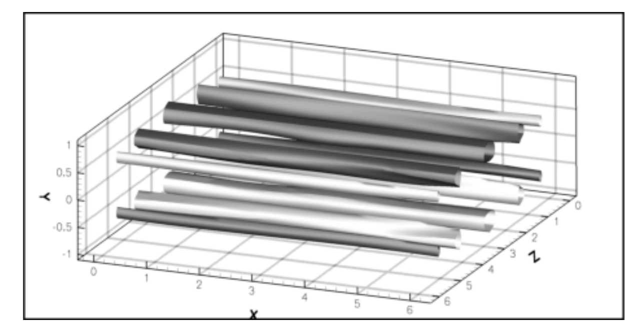
Energy growth



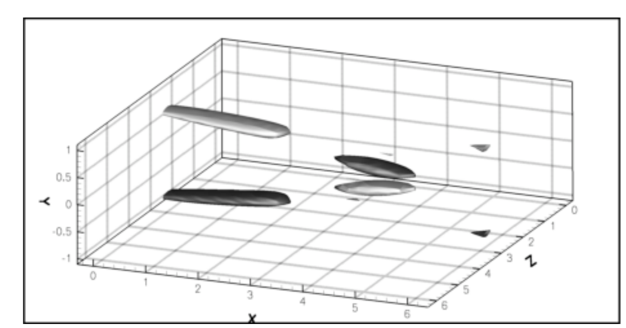
POD modes 4-5



POD mode 10

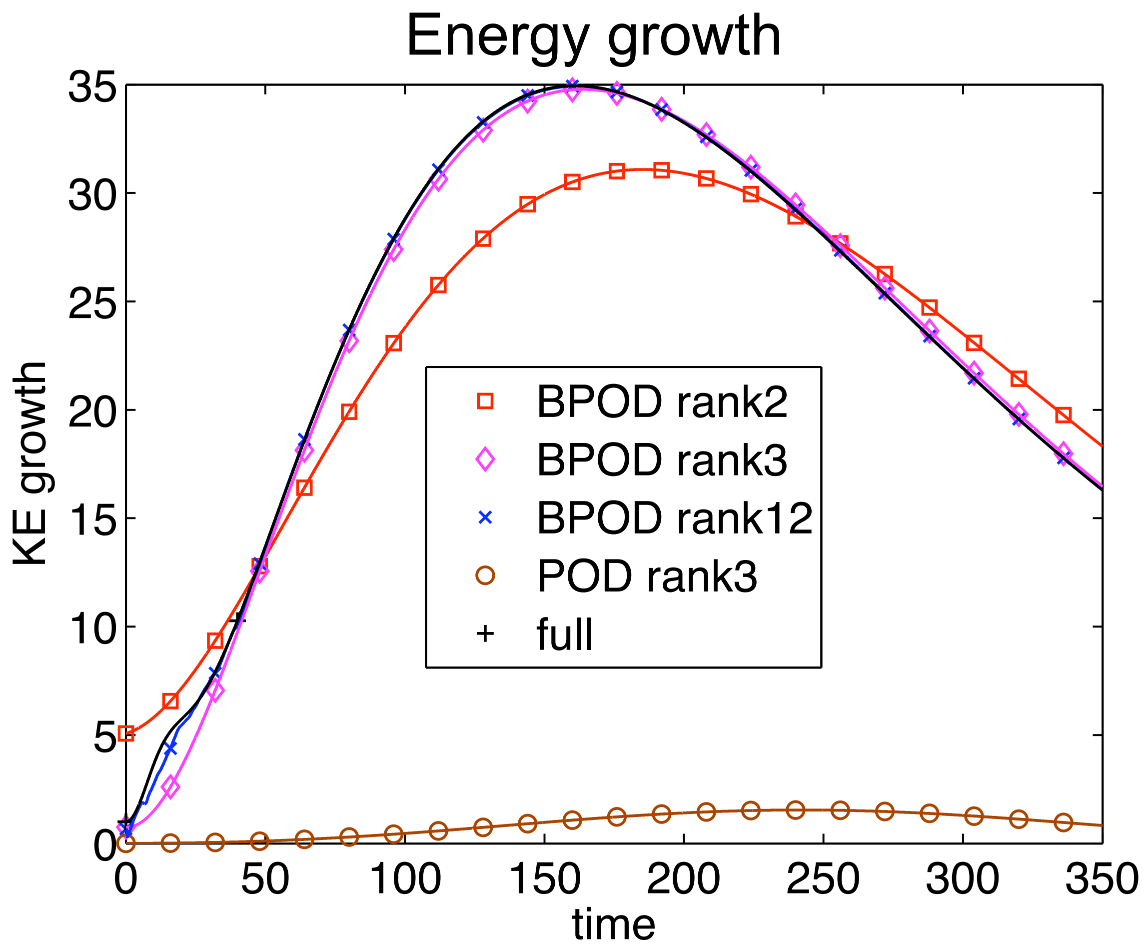


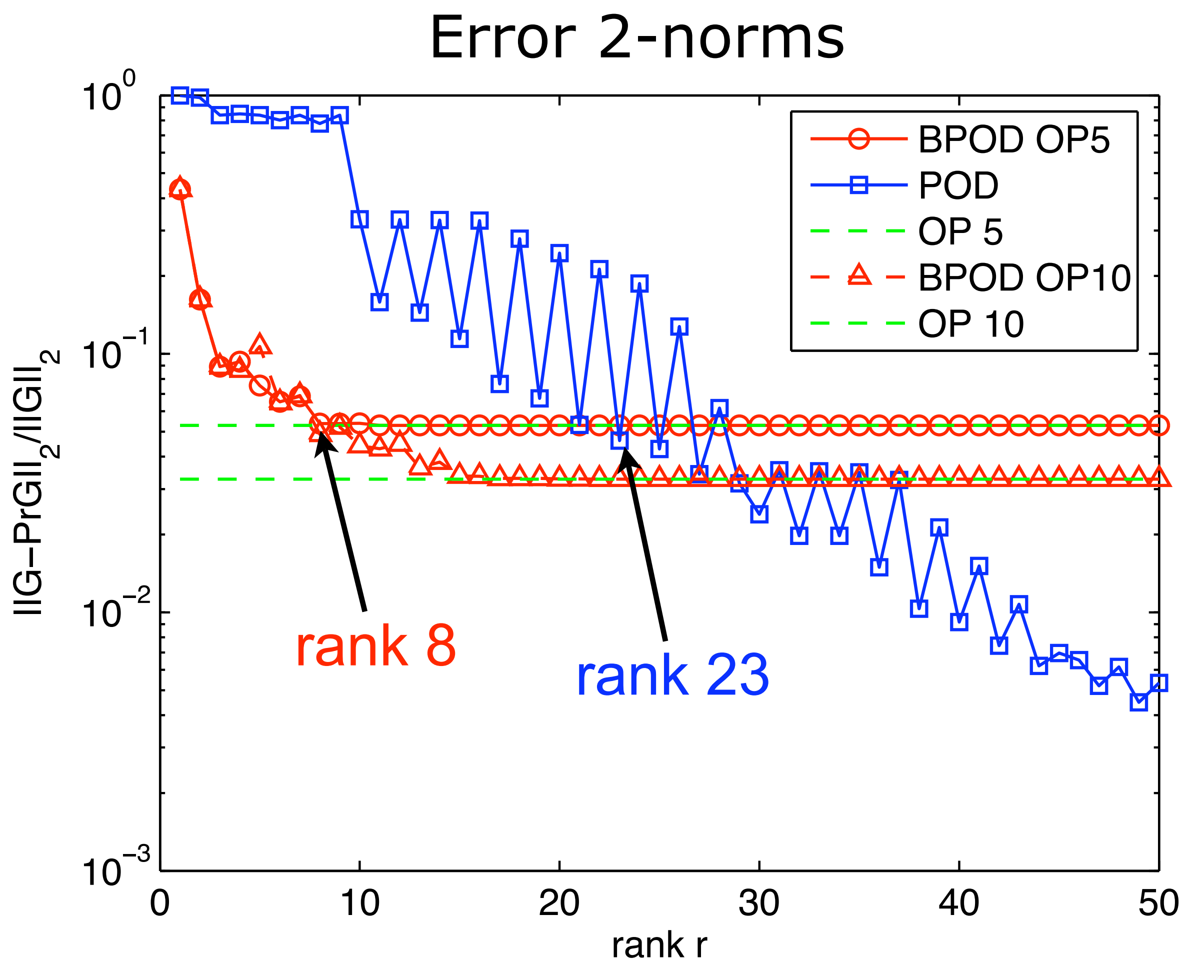
POD mode 17



Some low-energy POD modes are very important for the system dynamics.
Can't naively use just the most energetic ones

Balanced truncation model performance

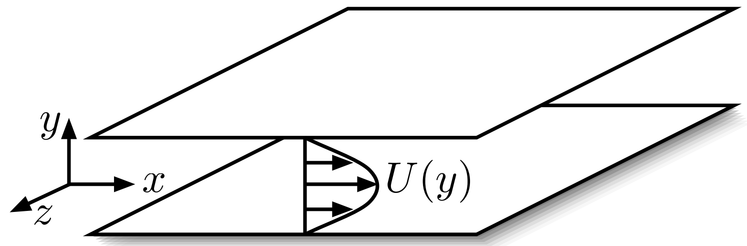




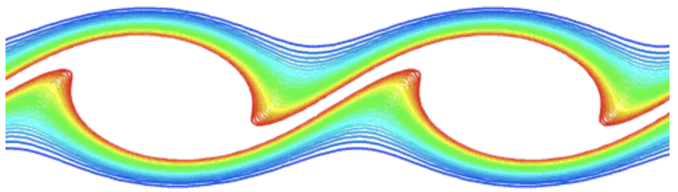
- Three-mode balanced truncation model excellent at capturing the energy growth
- Rank 8 balanced truncation sufficient to correctly capture the dynamics of the first five POD modes, compared to at least 23 POD modes

Outline

- Model reduction
  - Galerkin projection
  - Proper Orthogonal Decomposition
  - Balanced truncation
  - Example: linearized channel flow



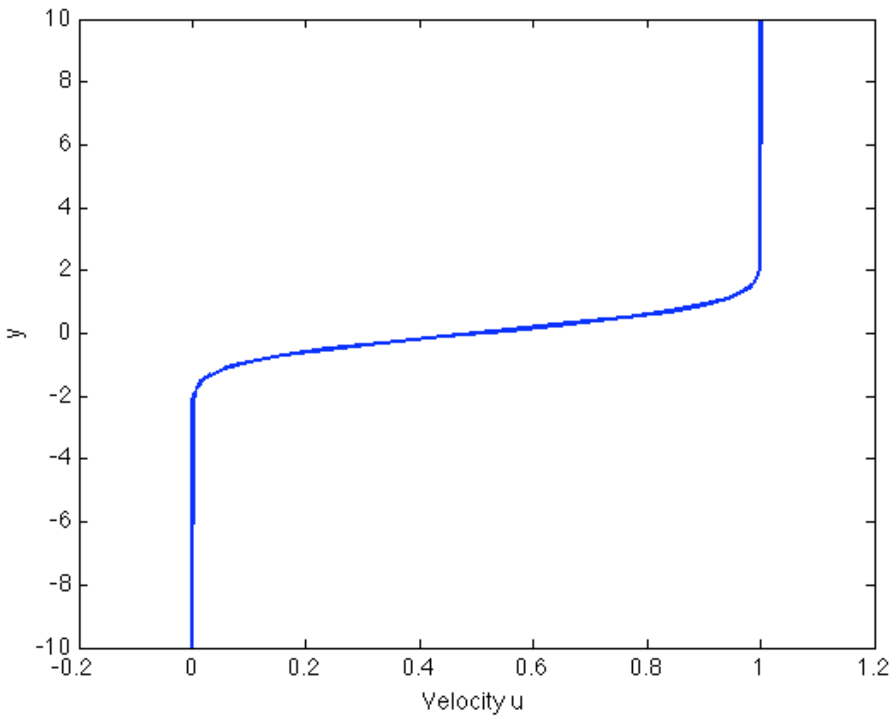
- Symmetry reduction using template fitting
  - Representing the quotient space using slices
  - Dynamics on the slice
  - Reconstruction equation
  - Example: free shear layer



Model reduction for self-similar problems

- Naive projection onto modes works poorly
  - Need many modes even to capture simple phenomena such as pure self-similar evolution

$$P_n u(x,t) = \sum_{j=1}^{n} a_j(t)\varphi_j(x)$$



  - Better to let the modes scale:

$$P_n u(x,t) = \sum_{j=1}^{n} a_j(t)\varphi_j(c(t)x)$$

Models need to specify $\dot{c}$

  - Other examples: traveling waves

$$P_n u(x,t) = \sum_{j=1}^{n} a_j(t)\varphi_j(x + c(t))$$

  - Do this in a systematic, general way

Setting

- Dynamics on M

$$\dot{x} = f(x) \qquad x \in M$$

Euler-Poincare $M = TG$

Lie-Poisson $M = T^*G$

- Group G acts on M (action is free and proper)

$$\Phi_g : M \to M \qquad g \in G$$

- Symmetry: equivariance of f

$$f \circ \Phi_g = T\Phi_g \circ f$$

- Get well-defined dynamics on the quotient space M/G

Euler-Poincare $TG/G \cong \mathfrak{g}$

Lie-Poisson $T^*G/G \cong \mathfrak{g}^*$

- Question: for general M, G, how to parameterize M/G?
  - Here, use slices and template fitting

Self-similar dynamics

- Dynamics on M

$$\dot{x} = f(x) \qquad\qquad x \in M$$

- Modified equivariance of f

$$m(g) f \circ \Phi_g = T\Phi_g \circ f$$

$m : G \to \mathbb{R}$ is a homomorphism
$m(g_1 g_2) = m(g_1) m(g_2)$

- write

$$x(t) = g(\tau) \cdot r(\tau)$$

$dt = m(g) d\tau$
$r \in M$

- equations become

$$r_\tau = f(r) - \xi_M(r) \qquad\qquad \xi = g^{-1}\dot{g} \in \mathfrak{g}$$

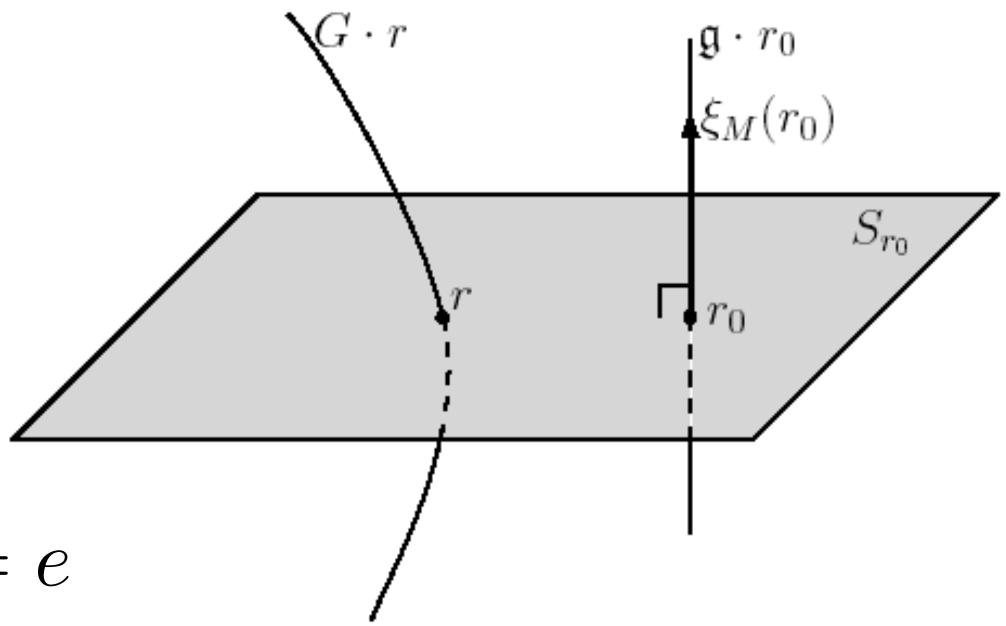
- Key idea: define ξ such that r evolves in a subspace that is locally isomorphic to M/G (a slice)

CWR, Kevrekidis, Marsden, Lust, *Nonlinearity* (2003)

Slices and template fitting

- Slices
  - Consider the set of functions $r \in M$ that are (locally) aligned with a template function $r_0 \in M$

$$\left.\frac{d}{ds}\right|_{s=0} \|r - g(s) \cdot r_0\|^2 = 0, \quad g(0) = e$$



  - Letting $\xi = \dot{g}(0) \in \mathfrak{g}$ this becomes

$$-2\langle\langle r - r_0, \xi_M(r_0)\rangle\rangle = 0$$

  - the slice at r_0 is defined as

$$S_{r_0} = \left\{ r \in M \;\middle|\; \langle\langle r - r_0, \xi_M(r_0)\rangle\rangle = 0, \text{for all } \xi \in \mathfrak{g} \right\}$$

and contains all functions (locally) aligned with r_0

Dynamics on the slice

- Dynamics in the scaled frame:

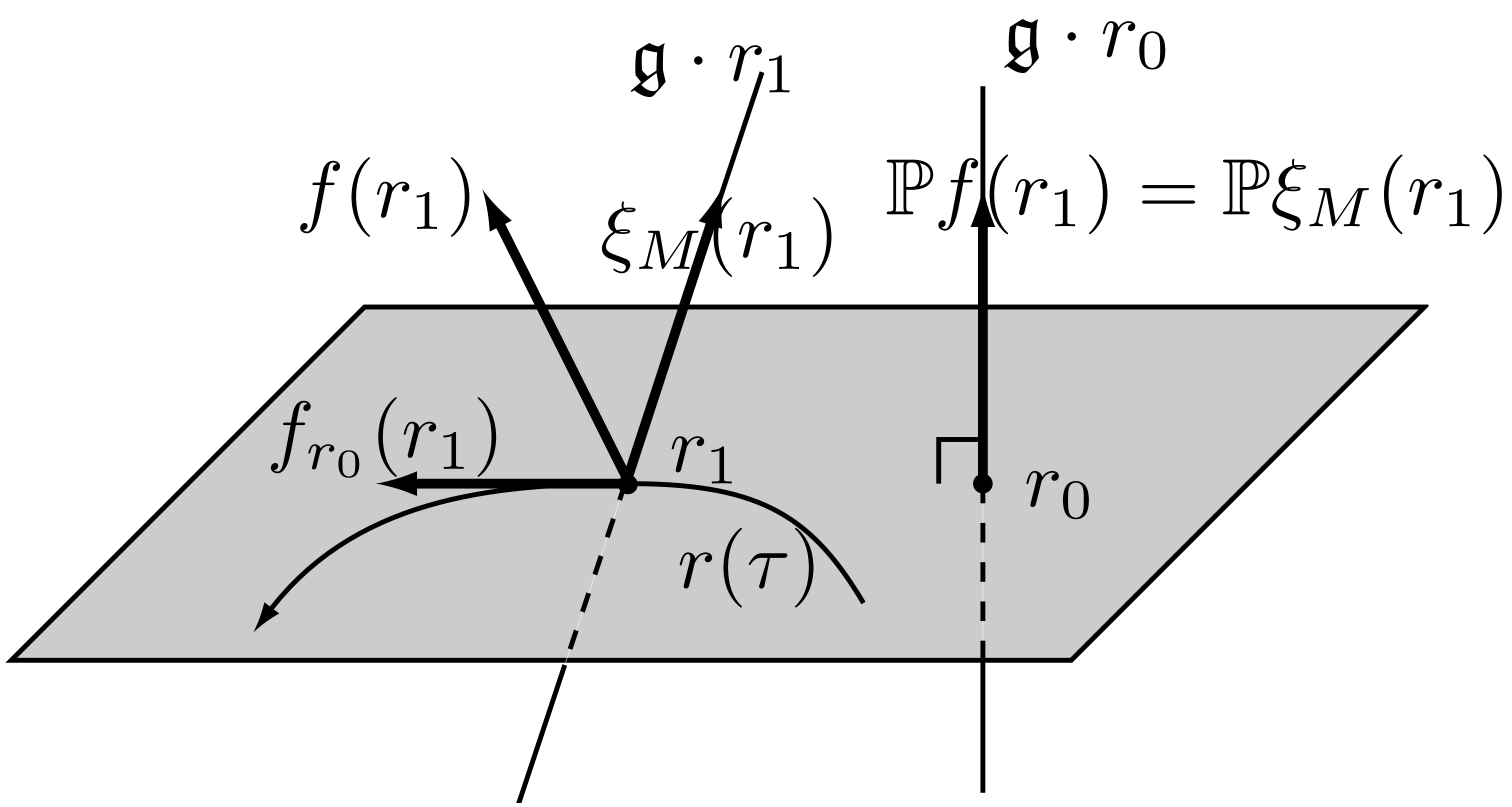
$$r_\tau = f(r) - \xi_M(r)$$

$$\dot{x} = f(x)$$
$$x(t) = g(\tau) \cdot r(\tau)$$

  - Solve for ξ

$$\mathbb{P}\xi_M(r) = \mathbb{P}f(r)$$

$\mathbb{P}$ projection onto $\mathfrak{g} \cdot r_0$

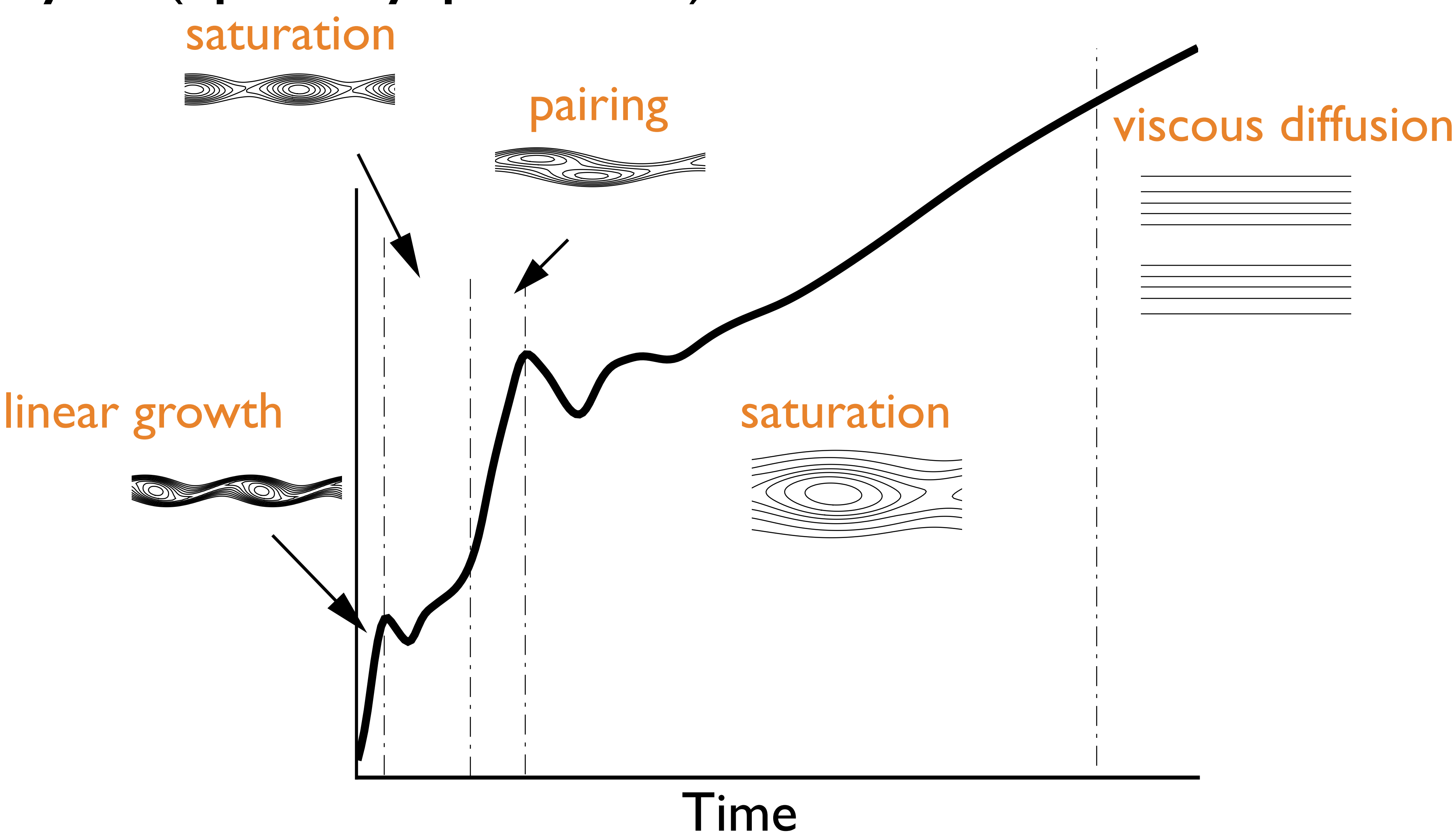


  - Substitute into dynamics for r

$$r_\tau = f_{r_0}(r) := f(r) - \xi_M(r)$$

Modeling free shear layers

- Evolution history of thickness for temporal shear layer (spatially periodic):



- Model initial linear growth, saturation, pairing, and eventual viscous diffusion

Methodology

- Scale POD modes dynamically in y direction to account for shear layer spreading
- Scaling invariants:
  - divergence of velocity field
  - inner product
- Key idea: template fitting
- Main result: an equation for the shear layer spreading rate:
  - as usual, also get equations for time coefficients of POD modes

Scaling basis functions

- Write solution in scaled reference frame

$$\mathbf{q} = (u, v)$$

$$\mathbf{q}(x, y, t) = G(g)\tilde{\mathbf{q}}(x, g(t)y, t)$$

- Choose $G(g) = \begin{bmatrix} 1 & 0 \\ 0 & 1/g \end{bmatrix}$: $\boxed{\mathrm{div}\ \mathbf{q} = \mathrm{div}\ \tilde{\mathbf{q}}}$
- Expand scaled variable $\tilde{\mathrm{q}}$ in terms of POD modes

$$\tilde{\mathbf{q}}(x, y, t) = \mathbf{u}_0(y) + \sum_{j=1}^{n} a_j(t)\varphi_j(x, y)$$

- Advantage of the scaling: capture similar-looking structures as shear layer spreads
- Advantage of divergence-invariant mapping: auto-satisfy continuity equation; simplify pressure term

Template fitting

- How do we choose the scaling g(t)?
  - Choose g(t) so that $\tilde{\mathbf{q}}(x, y, t)$ lines up best with a preselected template (here, the base flow):

$$\left.\frac{d}{ds}\right|_{s=0} \|\tilde{\mathbf{q}}(x, y, t) - \mathbf{u}_0(x, h(s)y)\|^2 = 0$$

for any curve $h(s) > 0$ with $h(0) = 1$

  - This means the scaled solution $\tilde{\mathbf{q}}(x, y, t)$ satisfies

$$\left\langle y\frac{\partial \mathbf{u}_0}{\partial y}, \tilde{\mathbf{q}} - \mathbf{u}_0 \right\rangle = 0$$

  - Geometrically, the set of all "properly scaled" functions $\tilde{\mathbf{q}}$ is an affine space through $\mathbf{u}_0$ and orthogonal to $y\partial_y \mathbf{u}_0$
  - This enables one to write dynamics for how the thickness g (t) evolves

$$\frac{\dot{g}}{g} = \frac{\langle f_g^1(\tilde{u}), y\partial_y u_0\rangle}{\langle y\partial_y \tilde{u}, y\partial_y u_0\rangle}$$

Equation for evolution of the thickness

- How does g(t) evolve in time?
  - We have a constraint ($\tilde{\mathbf{q}}(x, y, t)$ lines up best with template $\mathbf{u}_0$):

$$\left\langle y \frac{\partial \mathbf{u}_0}{\partial y}, \tilde{\mathbf{q}} - \mathbf{u}_0 \right\rangle = 0$$

  - Differentiate:

$$\left\langle y \frac{\partial \mathbf{u}_0}{\partial y}, \frac{\partial \tilde{\mathbf{q}}}{\partial t} \right\rangle = 0$$

  - Use equations of motion

$$\frac{\partial \tilde{\mathbf{q}}}{\partial t} = f_g(\tilde{\mathbf{q}}) - \frac{\dot{g}}{g} y \frac{\partial \tilde{\mathbf{q}}}{\partial y} - G(1/g)\dot{G}(g, \dot{g})\tilde{\mathbf{q}}(x, y, t)$$

  - This gives an equation for g:

$$\boxed{\frac{\dot{g}}{g} = \frac{\langle f_g^1(\tilde{u}), y\partial_y u_0 \rangle}{\langle y\partial_y \tilde{u}, y\partial_y u_0 \rangle}}$$

Galerkin equations for the shear layer

- Equation for the POD mode coefficients:
  - retain only modes k=1, n=1 and 2:

$$\dot a_{1,1} = \frac{g^2 c_{11g} + c_{11}}{g^2 n_{1g} + n_1} a_{1,1} + \frac{g^2 c_{12g} + c_{12}}{g^2 n_{1g} + n_1} a_{1,2} + \frac{1}{\mathrm{Re}} \left[-(\frac{2\pi}{L})^2 + \frac{g^2 d_{1g} + d_1}{g^2 n_{1g} + n_1} g^2 \right] a_{1,1} + \frac{g^2 e_{1g} + e_1}{g^2 n_{1g} + n_1} \frac{\dot g}{g} a_{1,1},$$

$$\dot a_{1,2} = \frac{g^2 c_{21g} + c_{21}}{g^2 n_{2g} + n_2} a_{1,1} + \frac{g^2 c_{22g} + c_{22}}{g^2 n_{2g} + n_2} a_{1,2} + \frac{1}{\mathrm{Re}} \left[-(\frac{2\pi}{L})^2 + \frac{g^2 d_{2g} + d_2}{g^2 n_{2g} + n_2} g^2 \right] a_{1,2} + \frac{g^2 e_{2g} + e_2}{g^2 n_{2g} + n_2} \frac{\dot g}{g} a_{1,2},$$

- Equation for the scaling g:

$$\dot g = \frac{c_{01}}{n_0} a_{1,1} a^*_{1,1} g + \frac{c_{02}}{n_0} a_{1,2} a^*_{1,2} g + \frac{c_{03}}{n_0} a_{1,1} a^*_{1,2} g + \frac{c_{04}}{n_0} a_{1,2} a^*_{1,1} g + \frac{1}{\mathrm{Re}} \frac{d_0}{n_0} g^3$$

- Retaining modes k=1 and 2, n=1 and 2 also tractable, but messy

- Use inner product that is preserved under scaling:

$$\langle \tilde{\mathbf{q}}_1, \tilde{\mathbf{q}}_2 \rangle_g = \int_\Omega (\frac{1}{g} \tilde u_1 \tilde u_2 + \frac{1}{g^3} \tilde v_1 \tilde v_2) dx dy$$

Results

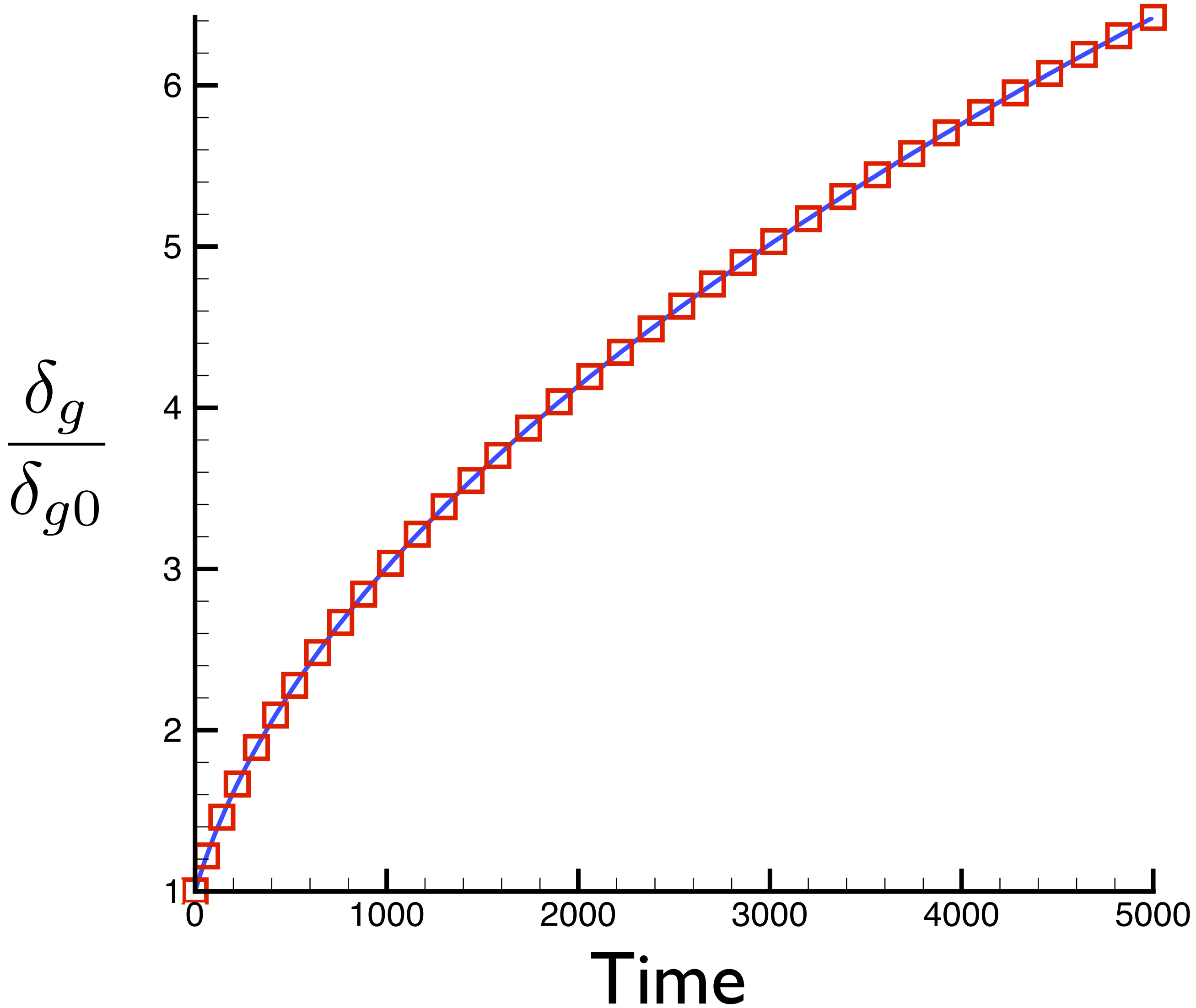
- Base flow with small perturbation
  - Base flow: $$u_0 = \frac{1}{2}\,\mathrm{erfc}(-y)$$
  - Perturbation is along the unstable eigenfunction of the linearized problem
- Consider three separate cases
  - No perturbation: viscous growth
  - Initial perturbation with k=1: vortex roll-up
  - Initial perturbation with k=2:
    - vortex roll-up
    - pairing
    - k=1 mode arises through pairing

Model results: k=0

- Only one equation left for g:

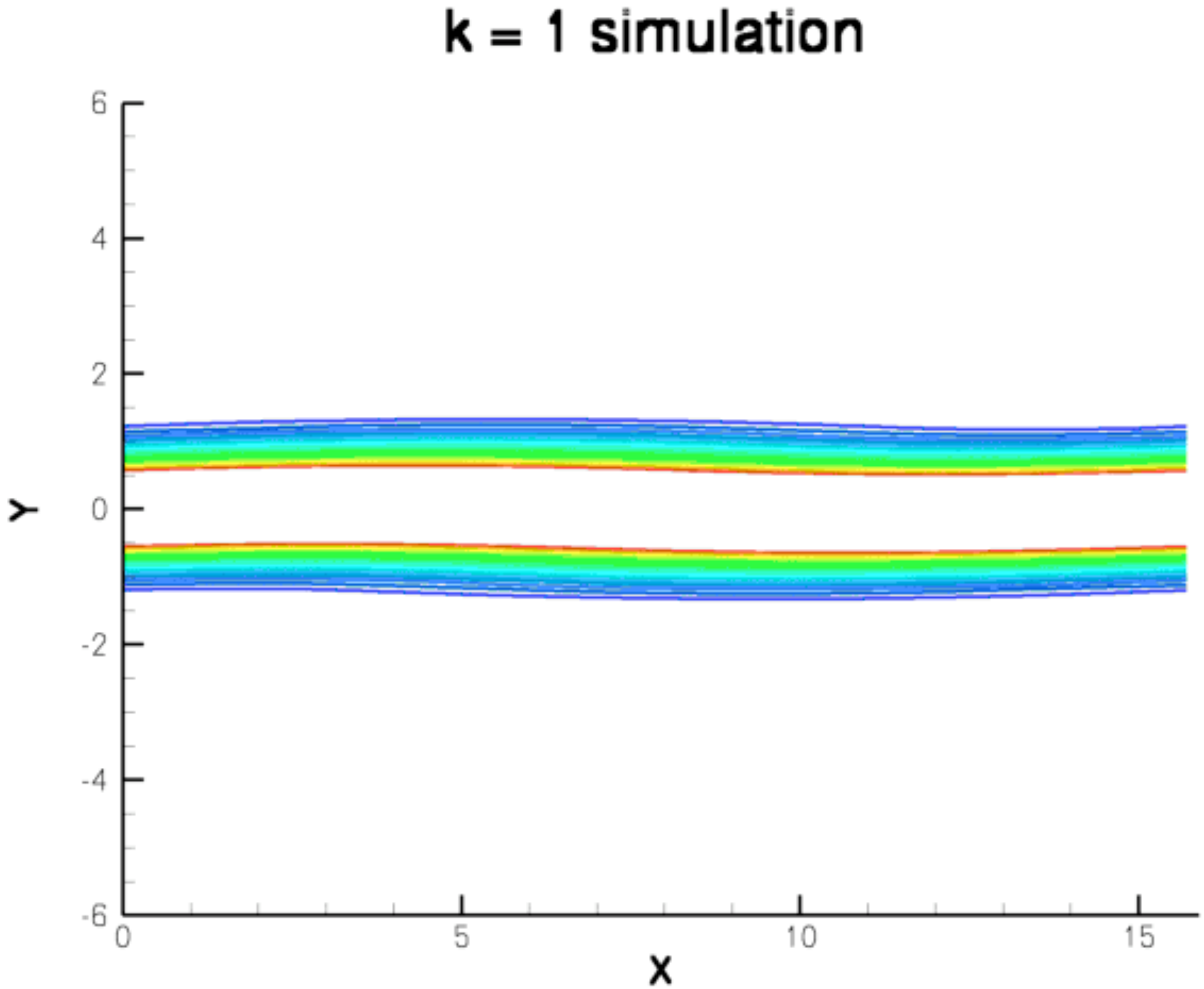
$$\dot{g} = -\frac{2}{\mathrm{Re}} g^3 \qquad g(t) = \left(\frac{\mathrm{Re}}{4t + \mathrm{Re}} \right)^{1/2}$$
$$g(0) = 1$$

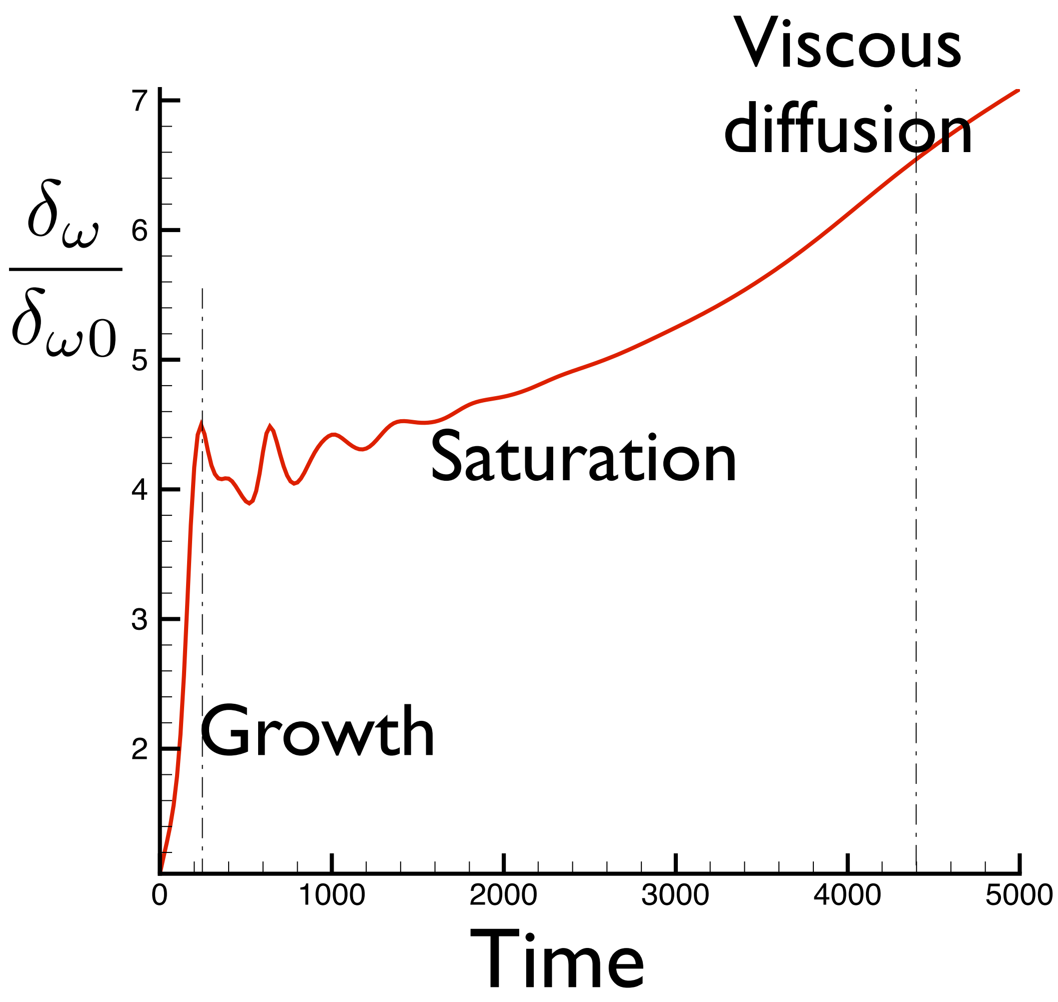
- Recovers exact theoretical growth rate for Stokes problem:



Movie of DNS

- Initial condition with k=1 (Re = 200)

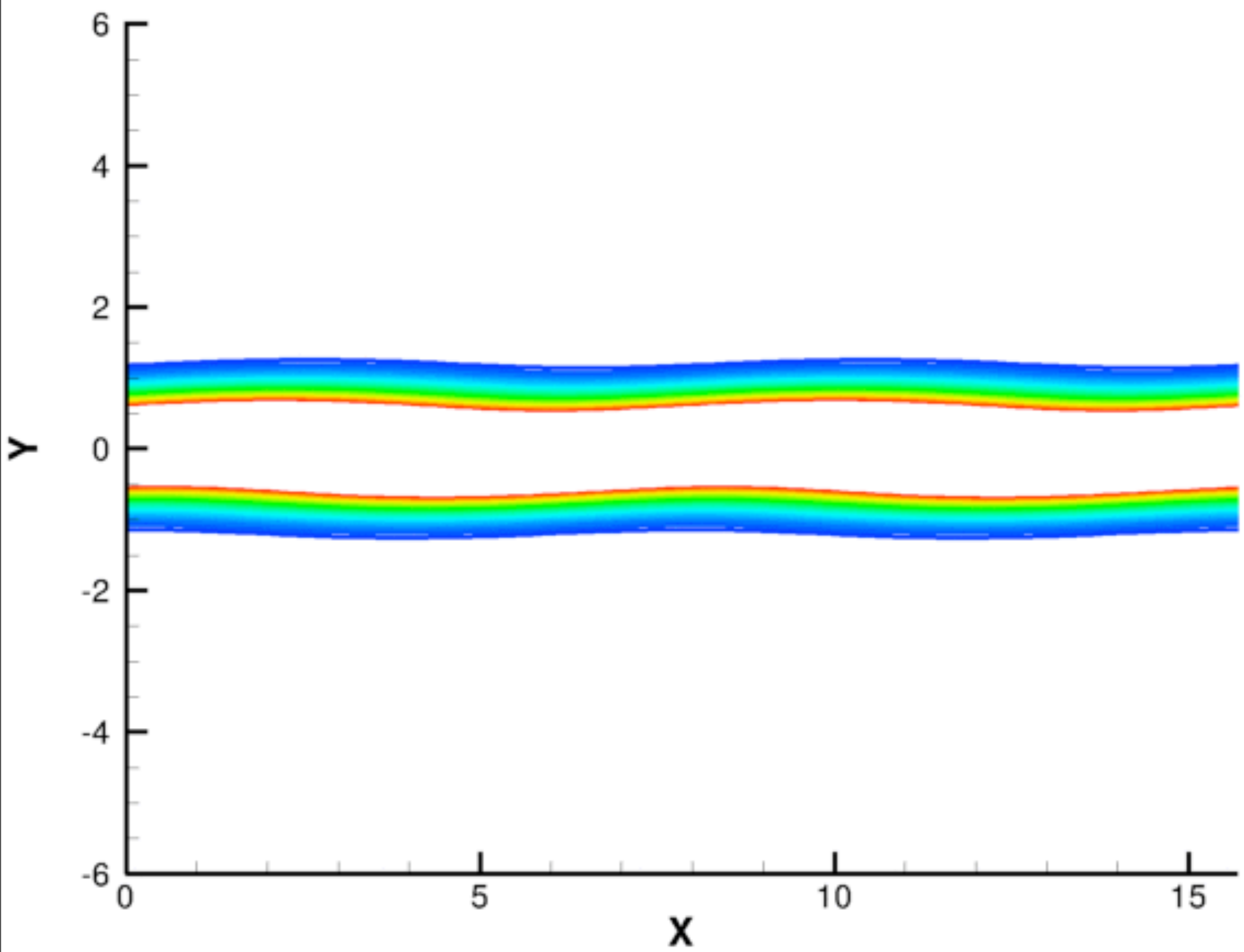


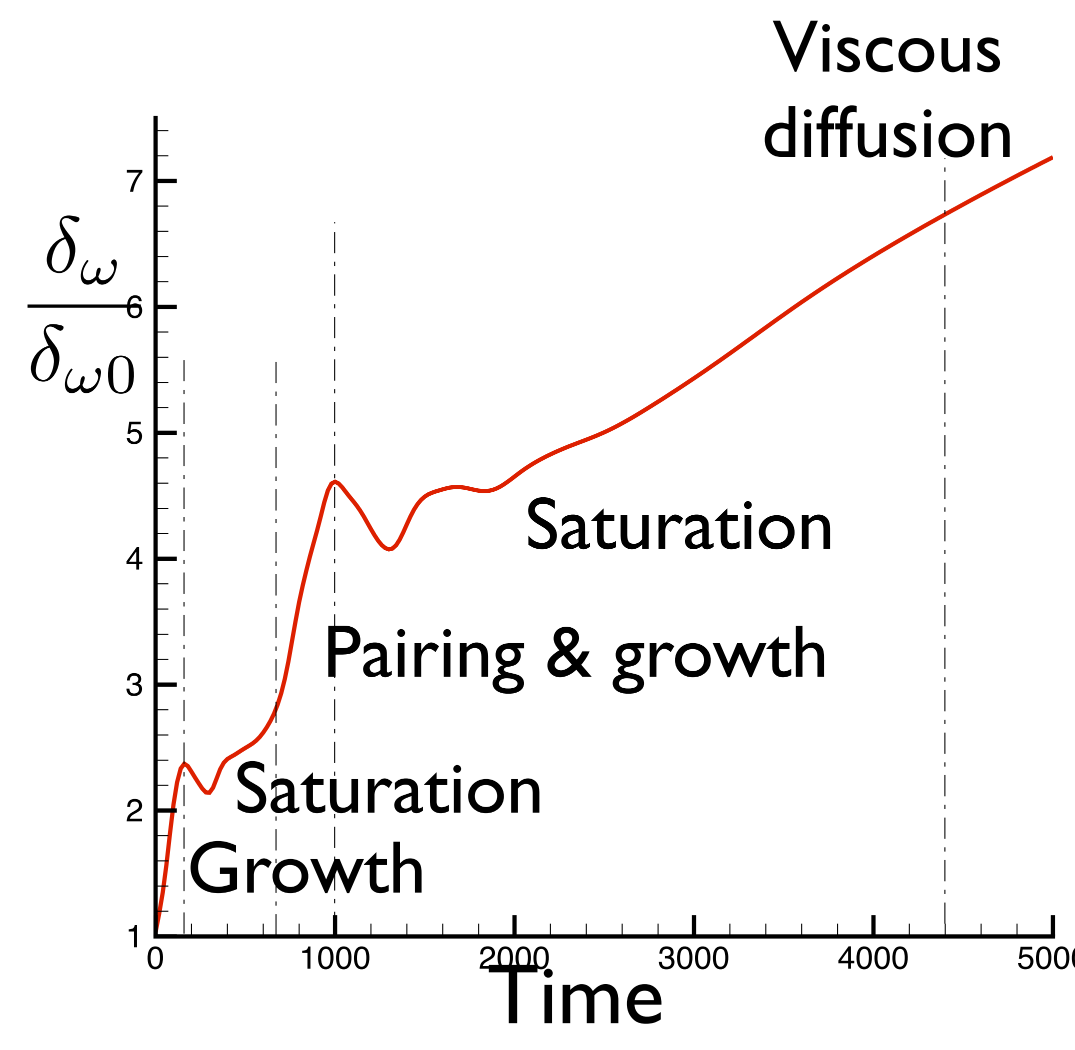


Movie of DNS

- Initial condition with k=2 (Re = 200)

k=2 simulation





POD modes

- Energy contained in modes (k=1 initial condition)

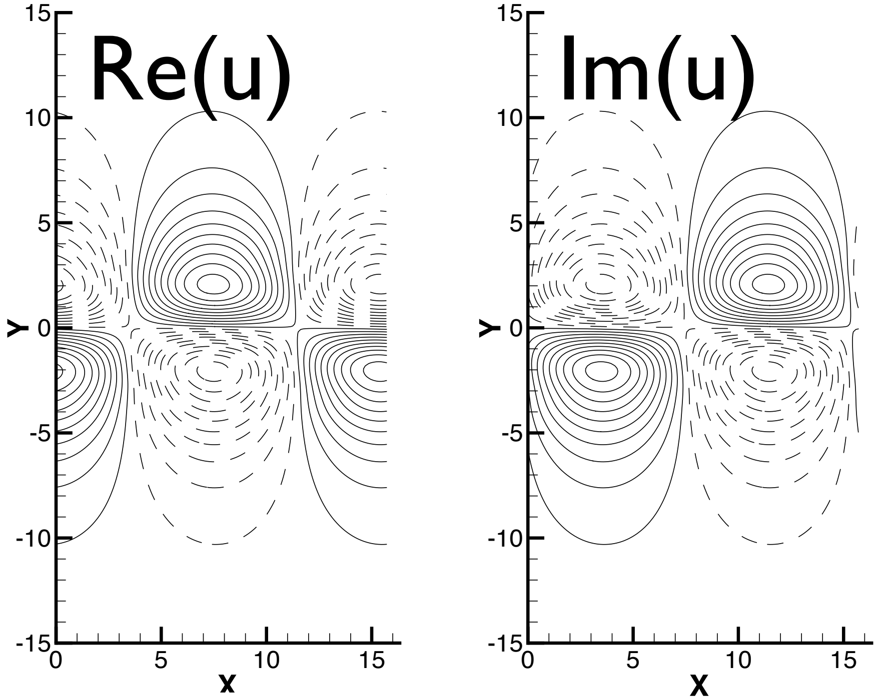
(k,n)	lambda	Energy (%)
(1,1)	130.3	91.0
(1, 2)	6.8	4.8
(2, 1)	4.5	3.1
all k=0		0.4

  - Zero mode contains very little energy - scaling was effective at removing the mean spreading

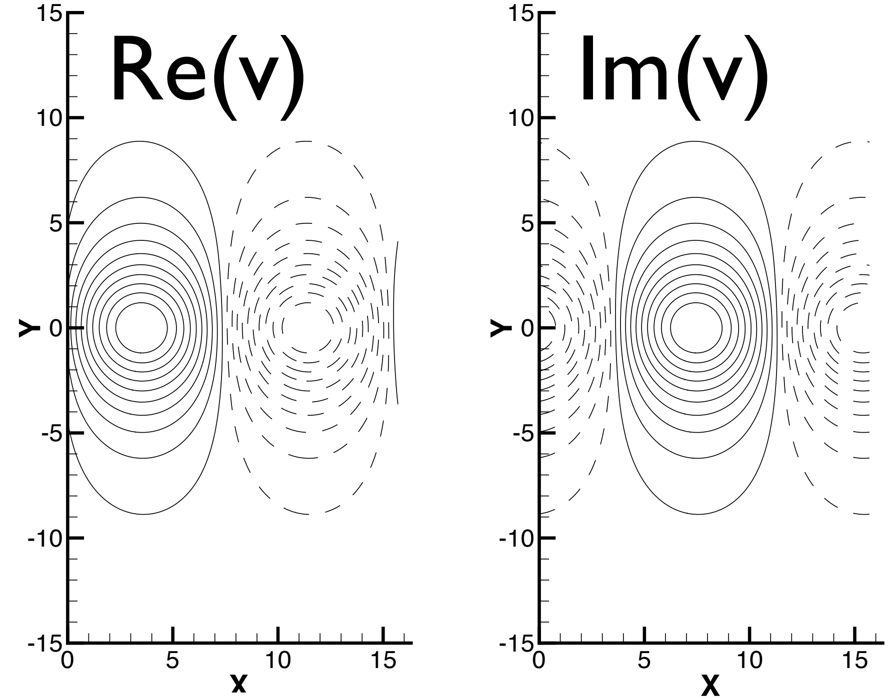
POD modes

- Initial condition with k=1

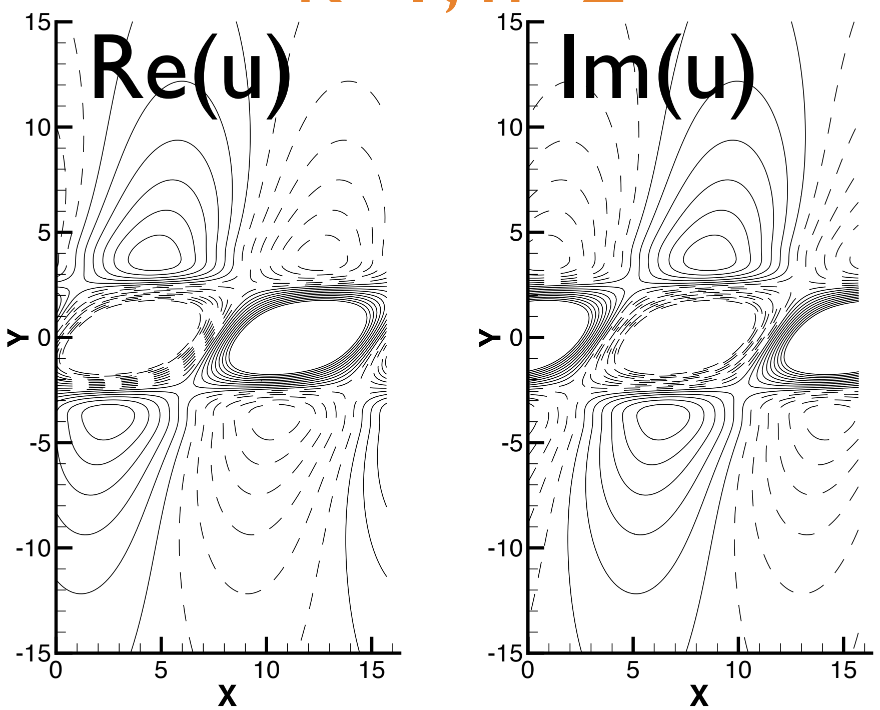
k=1, n=1



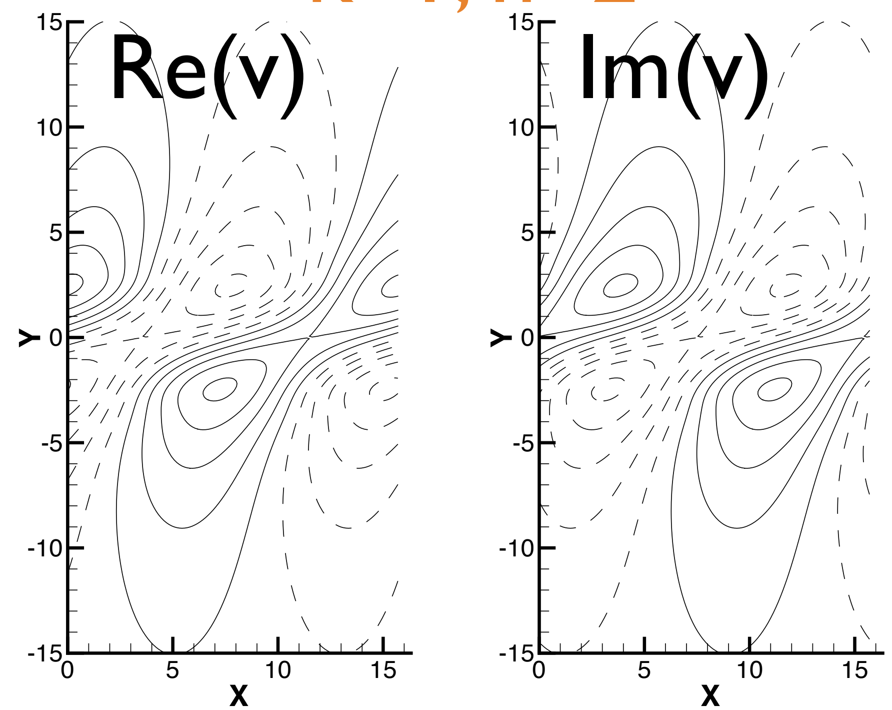
k=1, n=1



k=1, n=2



k=1, n=2



POD modes

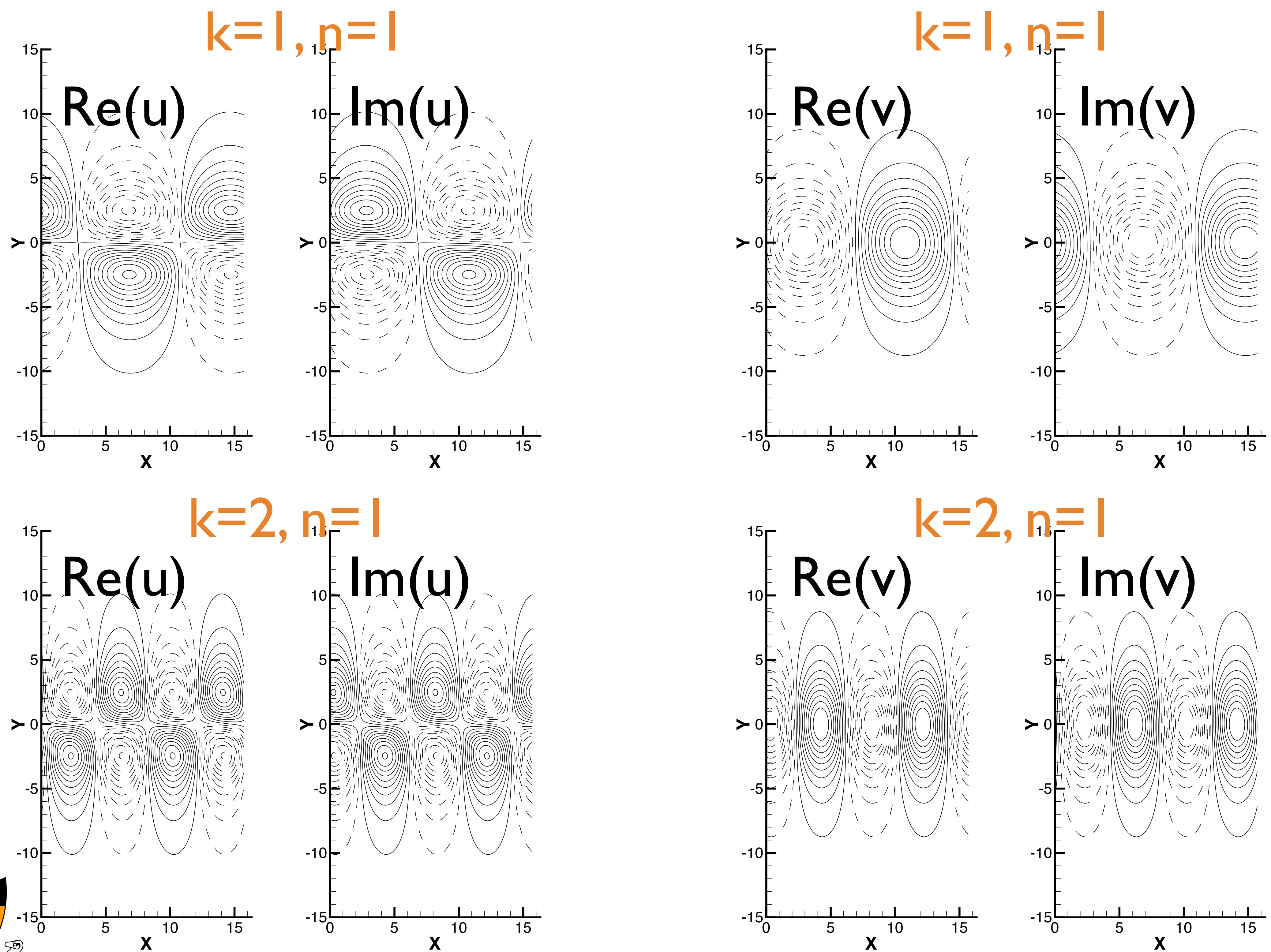
- Energy contained in modes (k=2 initial condition)

(k,n)	lambda	Energy (%)
(1,1)	27.5	40.1
(2,1)	37.9	55.2
(1,2)	0.9	1.3
(2,2)	1.6	2.3
all k=0		0.6

  - Scaling still effective at removing the mean spreading (zero mode has small energy)

POD modes

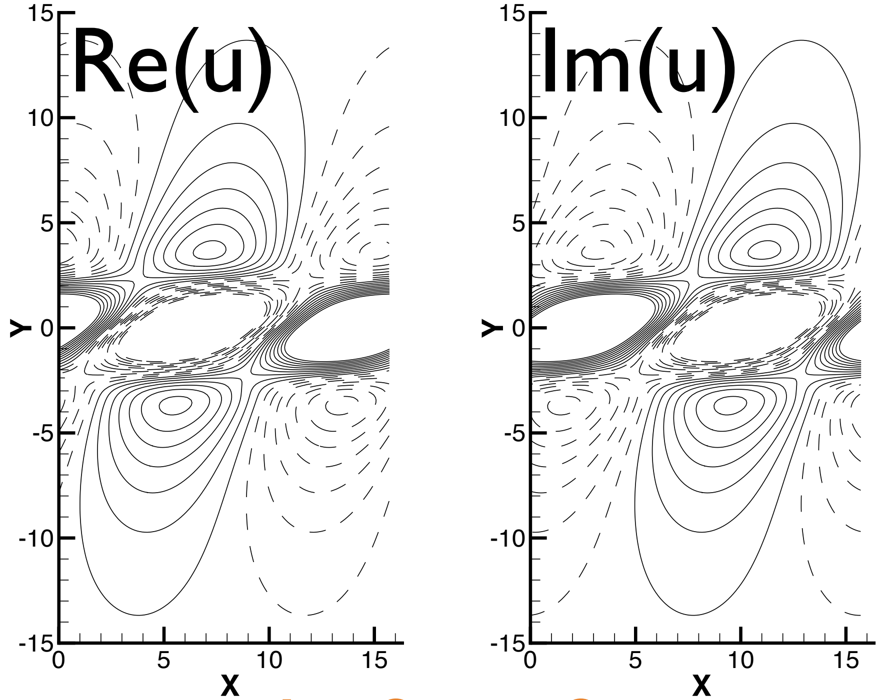
- Initial condition with k=2



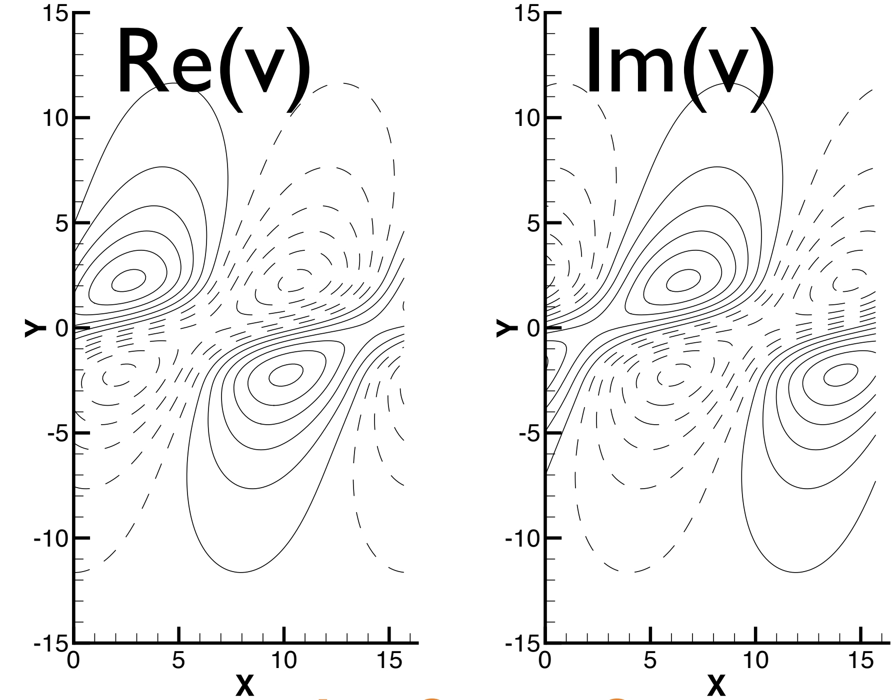
POD modes

- Initial condition with k=2

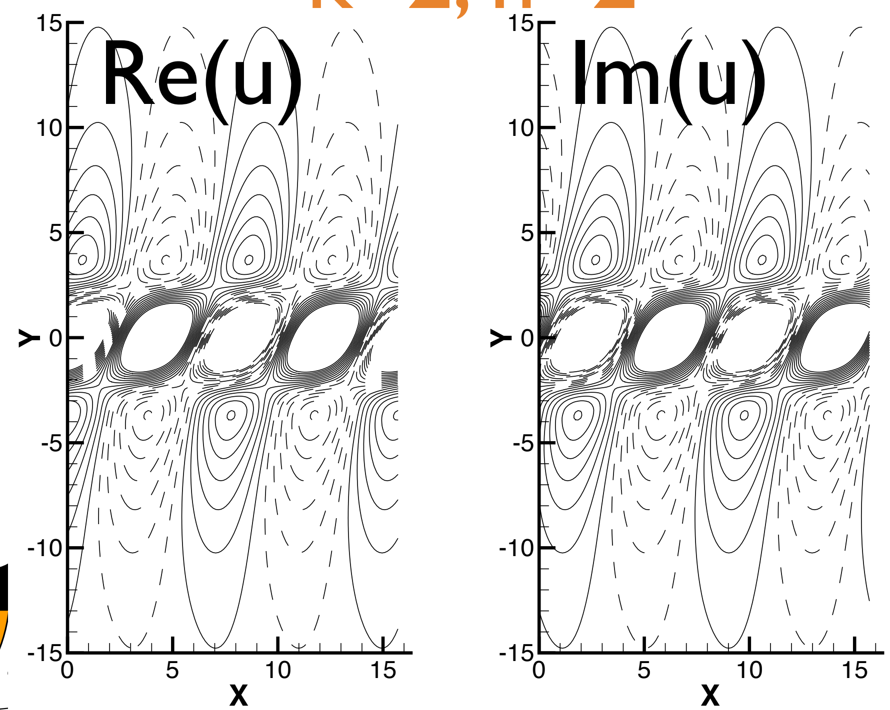
k=1, n=2



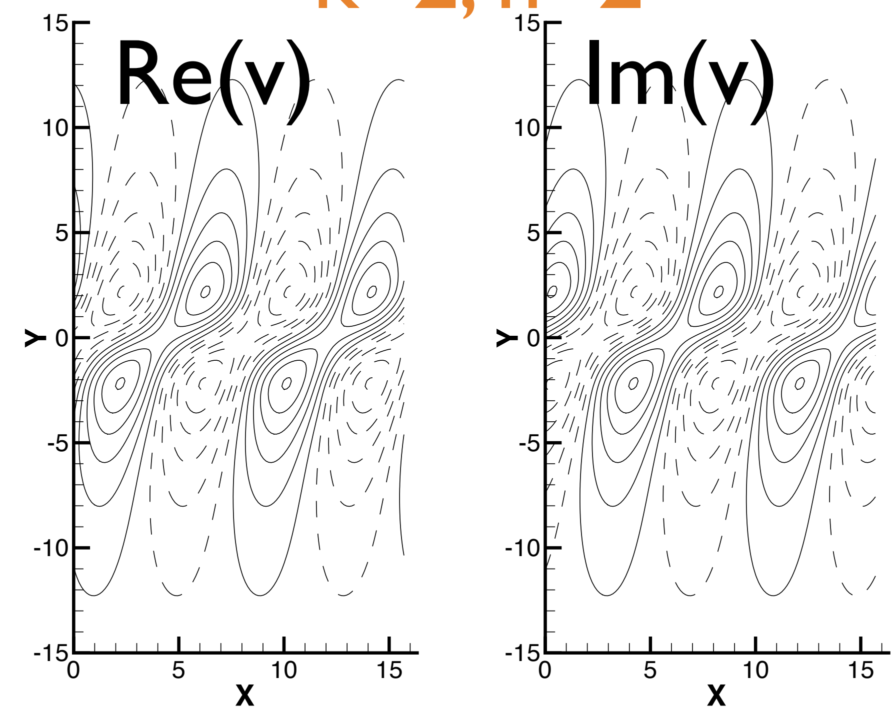
k=1, n=2



k=2, n=2



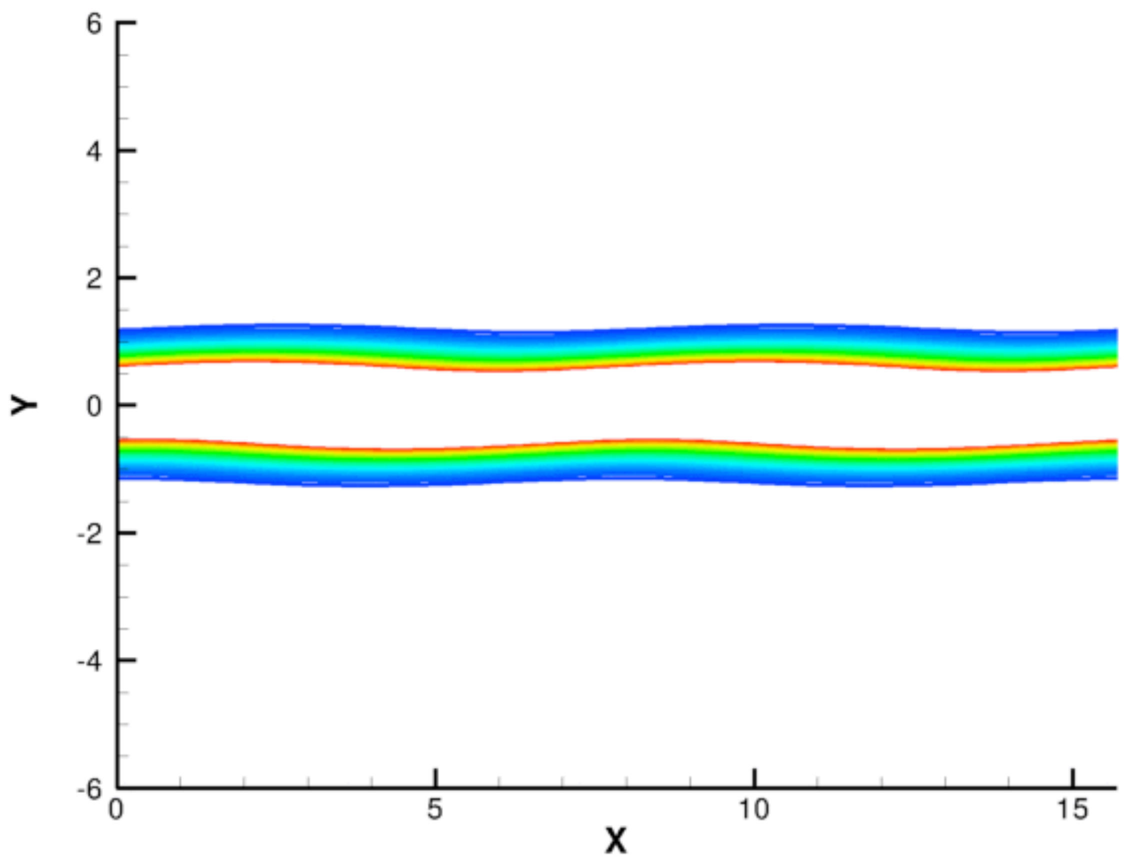
k=2, n=2

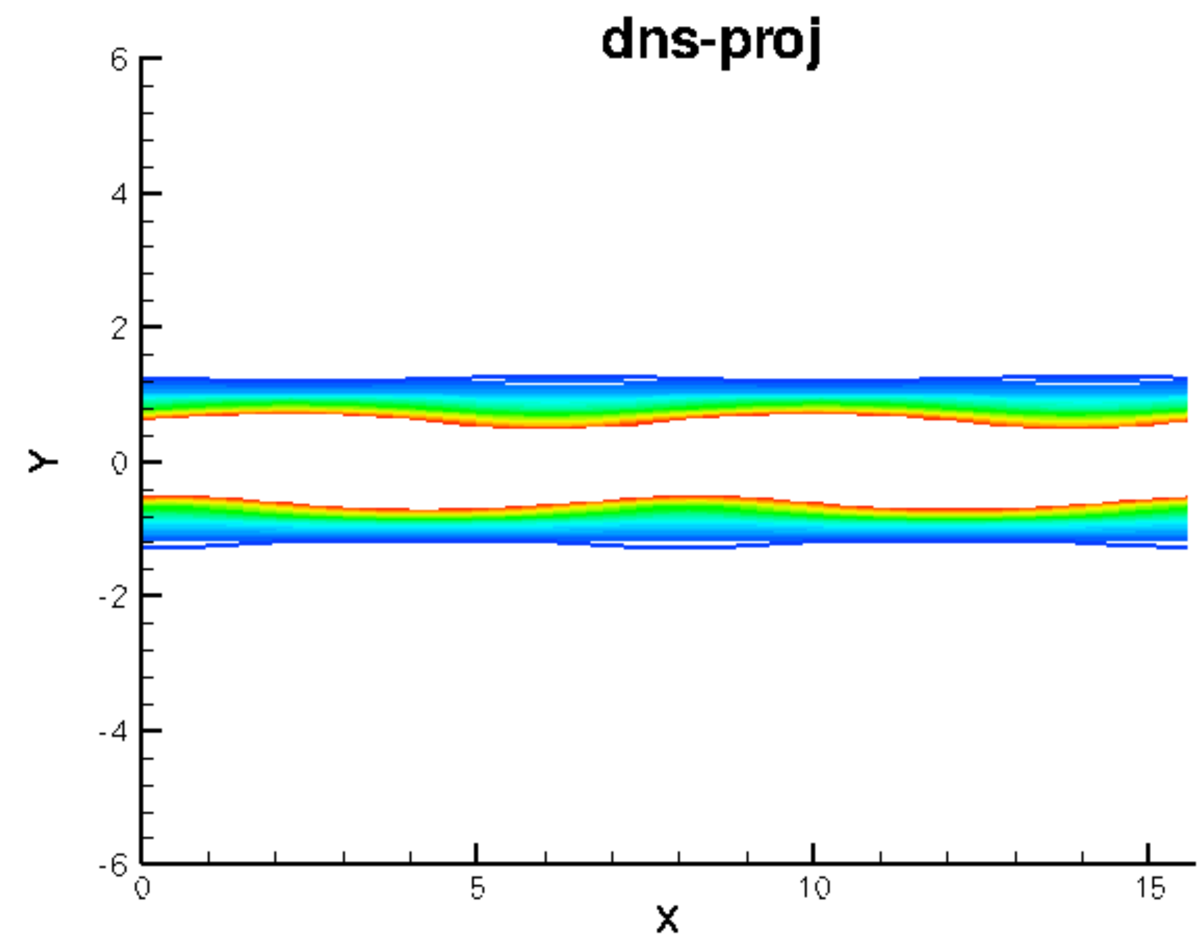


Full simulation vs. projection onto POD modes

- Comparison of direct numerical simulation, and projection onto four complex POD modes

Full simulation

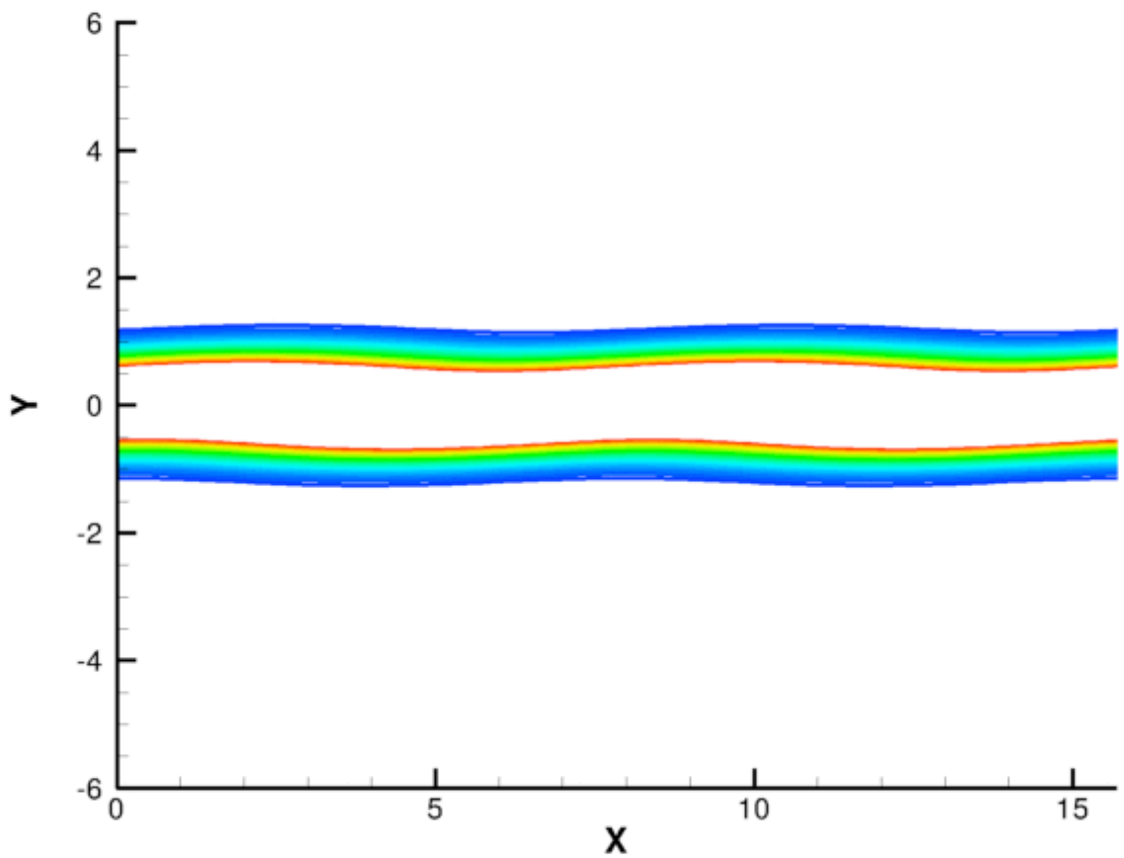


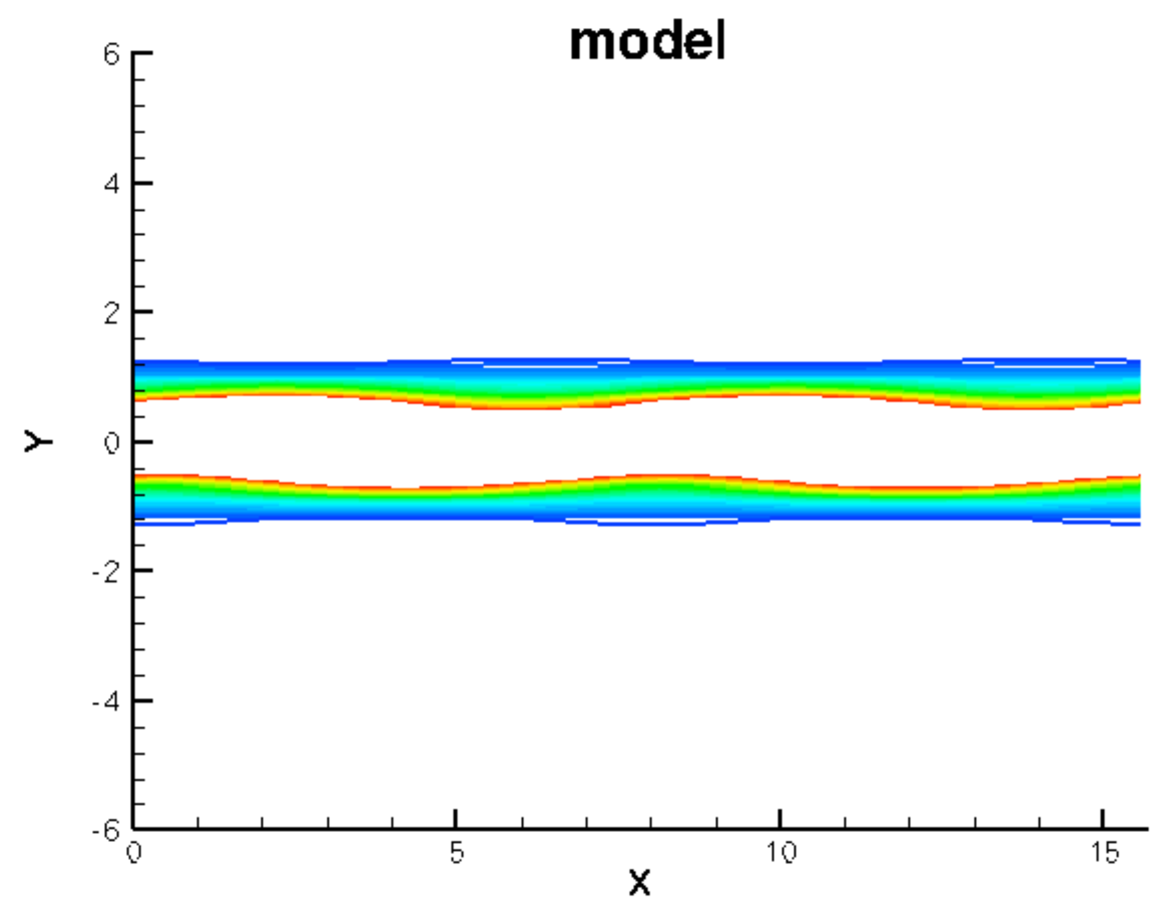


Full simulation vs. reduced-order model

- Comparison of direct numerical simulation, and reduced-order model using four complex POD modes

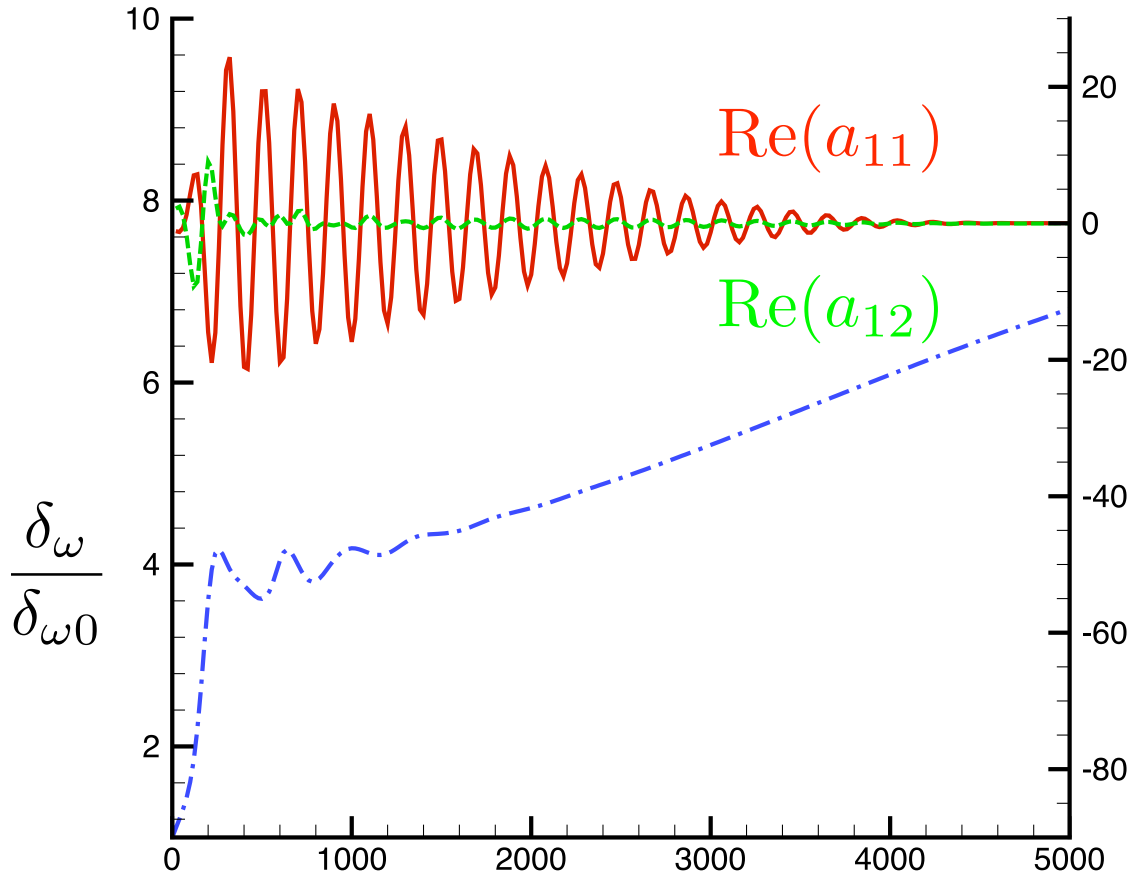
Full simulation



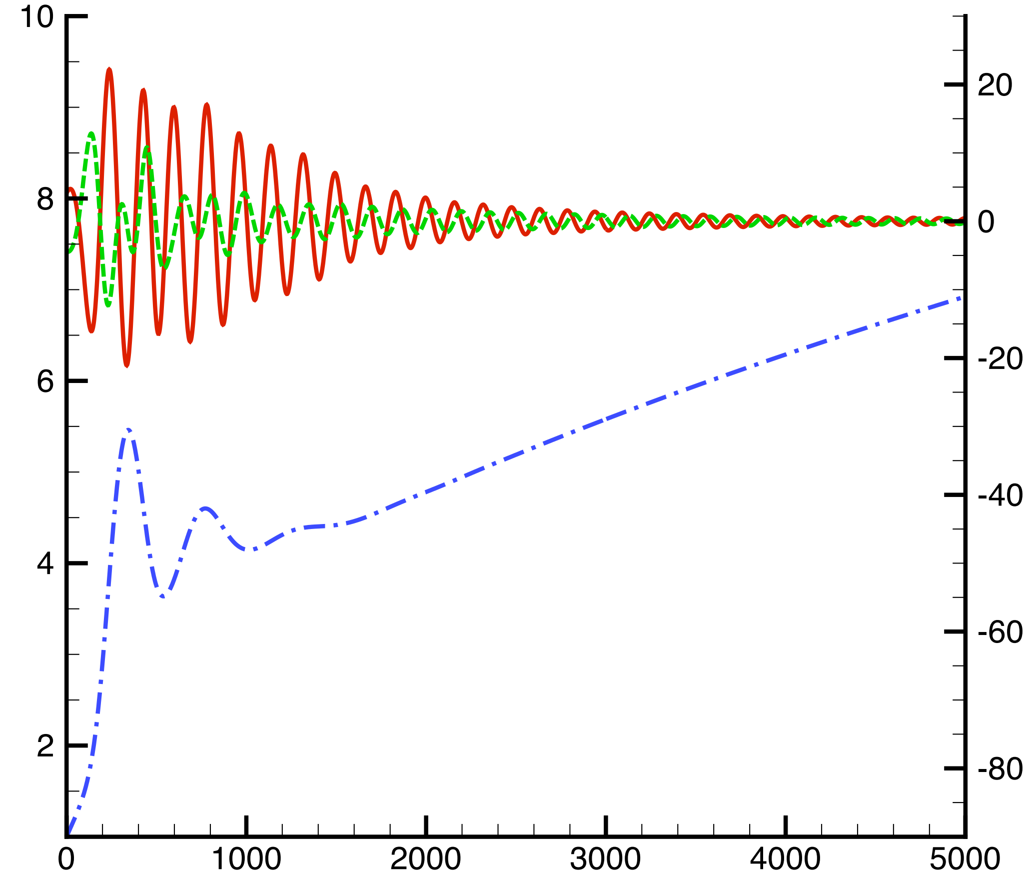


Model results: k=1

- Thickness and amplitude of POD modes for k=1 initial condition: projection of full simulation



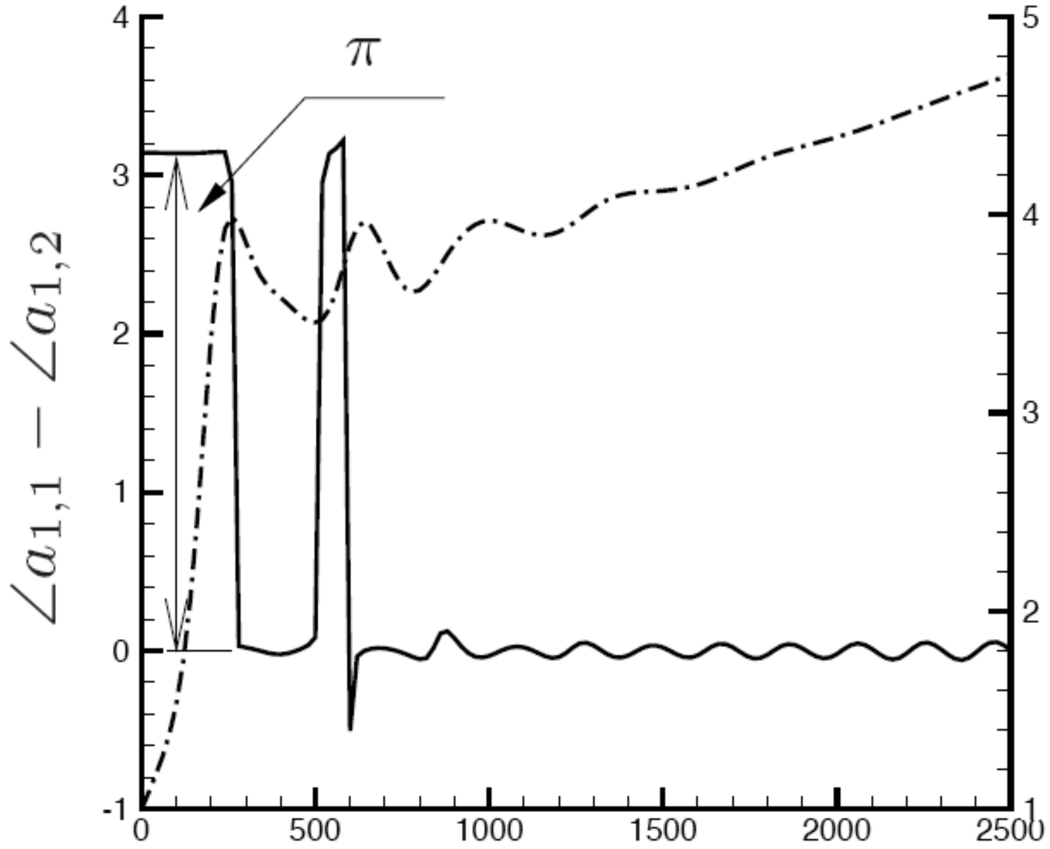
- Thickness and amplitude of POD modes for k=1 initial condition: low-dimensional model

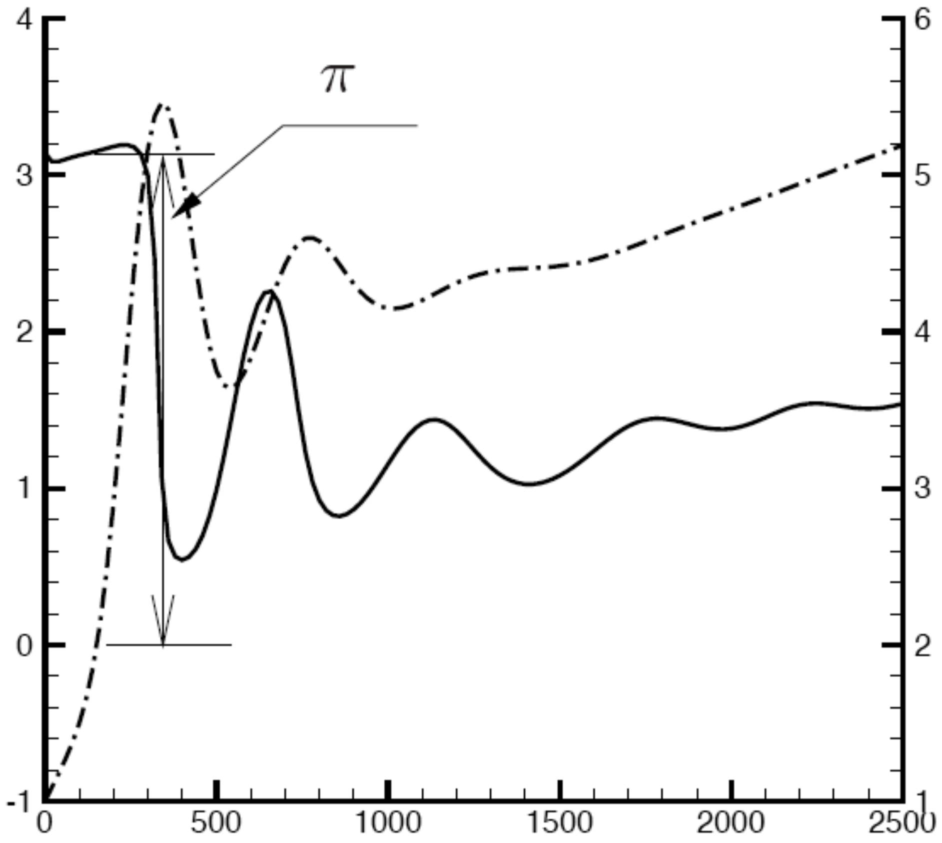


Phase shift phenomenon: Modes 1 and 2 are out of phase during linear growth, in phase after saturation

Model results: k=1

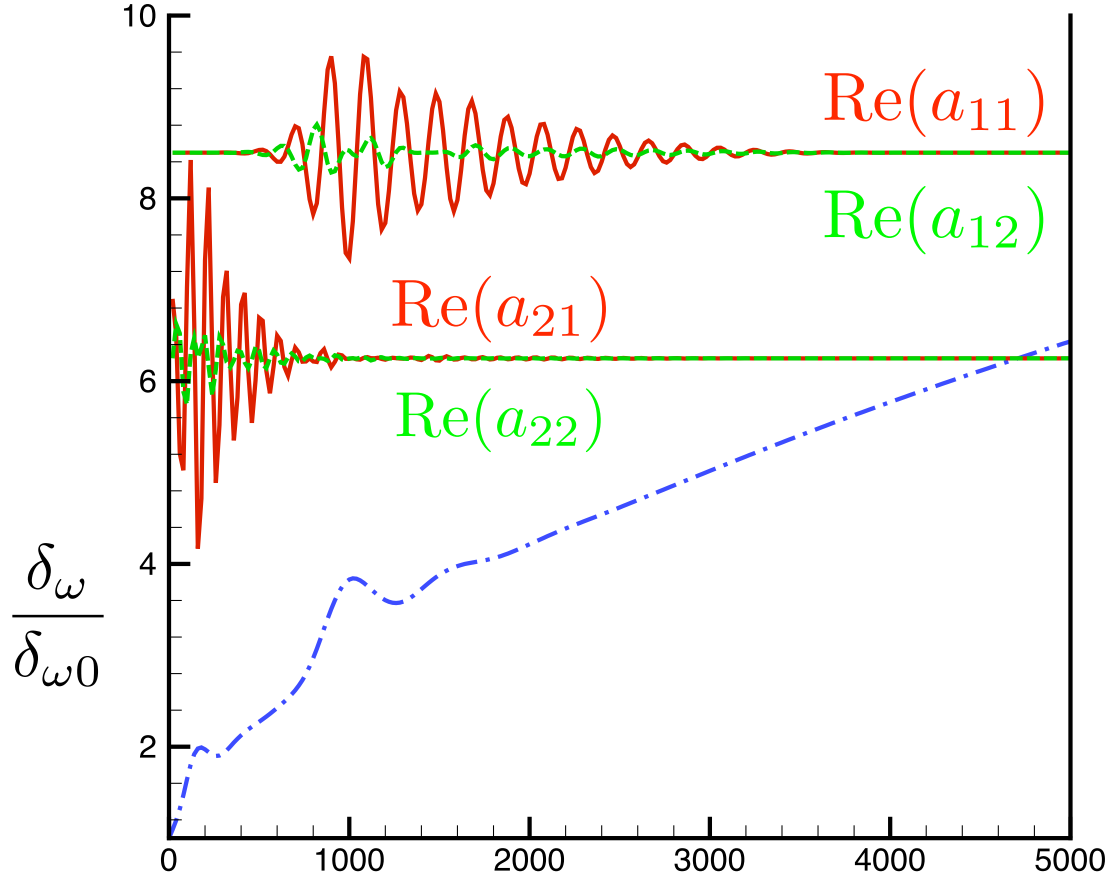
- Phase delay between the first 2 POD modes: projection of full simulation
- Phase delay between the first 2 Pod modes: low-dimensional model



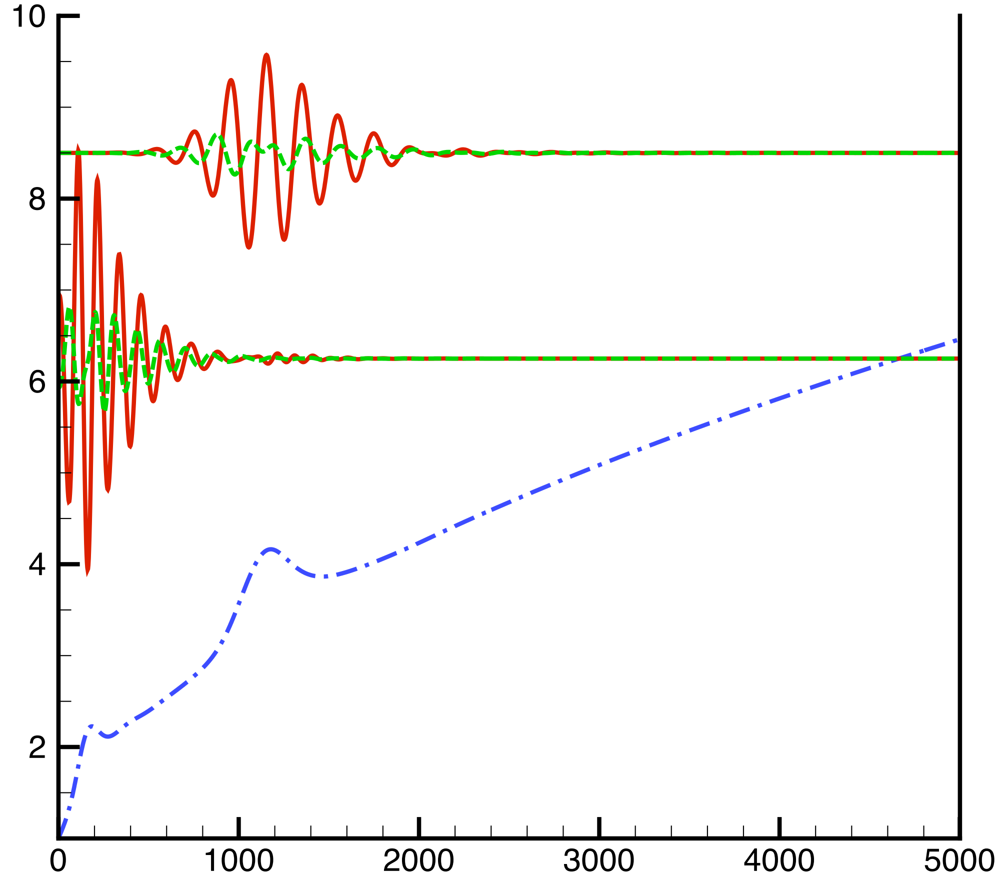


Model results: k=2

- Thickness and amplitude of POD modes for k=2 initial condition: projection of full simulation



- Thickness and amplitude of POD modes for k=2 initial condition: low-dimensional model



Example: control of Kuramoto-Sivashinsky equation

- Do the same procedure with a control input

$$\dot{x} = f(x) + g(x)u$$

  - Assume control action is equivariant

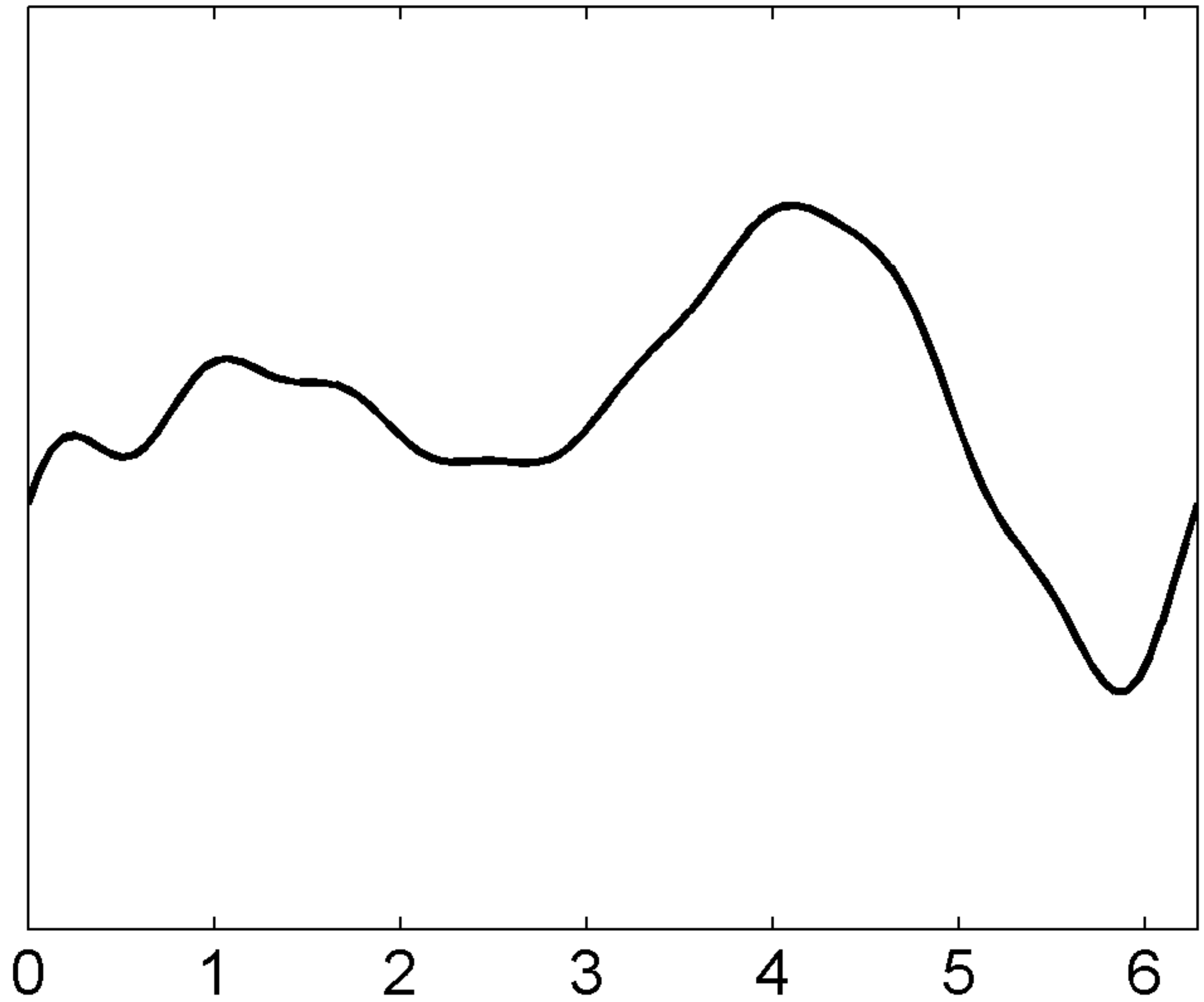
$$g(\Phi_g(x)) \circ \Psi_g = T_x\Phi_g \circ g(x)$$

  - Stabilize relative equilibria
  - Example: Kuramoto-Sivashinsky equation

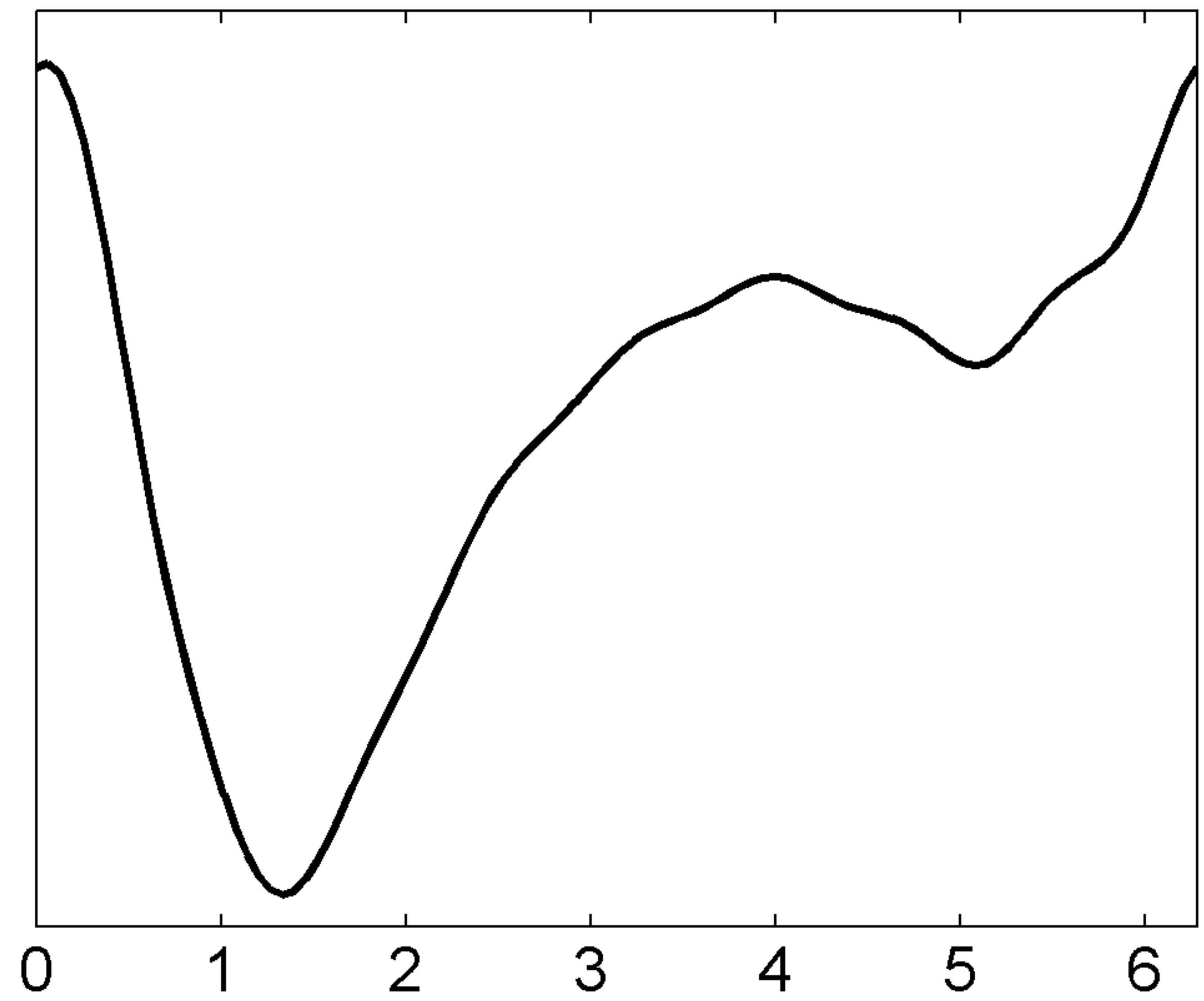
$$u_t = -uu_x - u_{xx} - \nu u_{xxxx}, \qquad x \in [0, 2\pi)$$

    - Stabilize family of fixed points
    - Stabilize a traveling wave solution
    - See poster by Sunil Ahuja

K-S equation: stabilize a fixed point



KS equation: stabilize a traveling wave



Summary

- Model reduction
  - Main idea: project dynamics onto a smaller-dimensional subspace
  - Two choices: subspace itself, and the direction of projection (i.e., the inner product)
  - Proper orthogonal decomposition is one method of determining a subspace
  - Balanced truncation (linear systems) determines an inner product as well, and usually works much better than POD
- Dynamically scaled modes
  - Symmetry reduction scales self-similar solutions appropriately
  - Dynamic scaling decreases number of modes required
  - Temporal shear layer dynamics modeled with 4 complex modes, including linear growth, saturation, pairing, and viscous diffusion