Boundary Control for Parabolic PDEs and Turbulent Flows

Miroslav Krstic

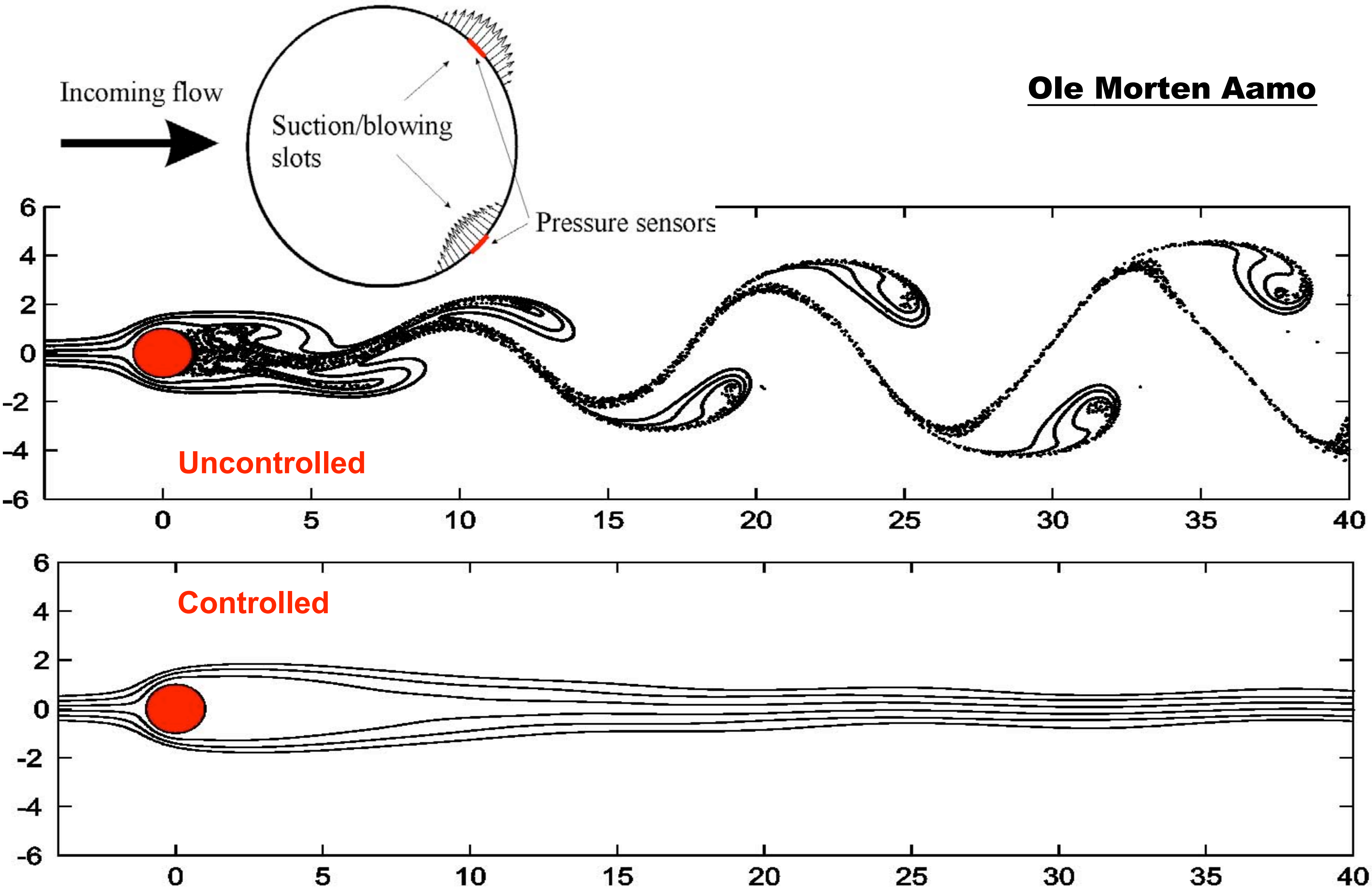
(with Andrey Smyshlyaev, Rafael Vazquez, and Ole-Morten Aamo)

University of California, San Diego

Stanford University, December 1, 2004

Example of a Flow Control Problem: Cylinder Wake/Vortex Shedding

Ole Morten Aamo



Boundary Control (Linear)

- **Eigenvalue Assignment.** Triggiani (1980); Lasiecka and Triggiani (1983). Moving (rather than arbitrarily placing) infinitely many eigenvalues: Russell (1977)

- **LQR.** Lasiecka and Triggiani book (2000) and various other previous authors. Related optimal control results: Bensoussan, Da Prato, Delfour, Mitter book (1993)

- **Boundary Observers/Output Feedback.** Nambu (1984)

- **Frequency Domain.** Surveys by Curtain (1990) and Logemann (1993)

- **Abstract Boundary Control Theory:** Fattorini (1968)

Backstepping Boundary Control of PDEs

Example plant $u_t = u_{xx} + \lambda_0 u \qquad x \in (0,1)$

Backstepping Boundary Control of PDEs

Example plant $$u_t(t,x) = u_{xx}(t,x) + \lambda_0 u(t,x) \qquad x \in (0,1)$$

Backstepping Boundary Control of PDEs

Example plant

$$u_t = u_{xx} + \lambda_0 u \qquad x \in (0,1)$$

$$u(x = 0, t) = 0$$

Backstepping Boundary Control of PDEs

Example plant

$$u_t = u_{xx} + \lambda_0 u \qquad x \in (0,1)$$

$$u(0) = 0$$

Control

$$u(1) = \int_0^1 k(1, y)\, u(y)\, dy$$

Backstepping Boundary Control of PDEs

Example plant

$$u_t = u_{xx} + \lambda_0 u$$
$$u(0) = 0$$

$$\left(\begin{array}{l} \dot{u}_i = \dfrac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + \lambda_0 u_i \\ u_0 = 0; \qquad u_N = \text{control} \end{array} \right)$$

Control

$$u(1) = \int_0^1 k(1, y)\, u(y)\, dy$$

Backstepping Boundary Control of PDEs

Example plant

$$u_t = u_{xx} + \lambda_0 u \qquad x \in (0,1)$$

$$u(0) = 0$$

Control

$$u(1) = \int_0^1 k(1, y)\, u(y)\, dy$$

Transformation

$$w(x) = u(x) - \int_0^x k(x, y)\, u(y)\, dy$$

Target system

$$w_t = w_{xx}$$

Backstepping Boundary Control of PDEs

Example plant

$$u_t = u_{xx} + \lambda_0 u \qquad x \in (0,1)$$

$$u(0) = 0$$

Control

$$u(1) = \int_0^1 k(1, y)\, u(y)\, dy$$

Transformation

$$w(x) = u(x) - \int_0^x k(x, y)\, u(y)\, dy$$

Target system

$$w_t = w_{xx} - c\, w \qquad c \geq 0$$

$$w(0) = w(1) = 0$$

UCSD Jacobs | Mechanical and Aerospace Engineering

Backstepping Boundary Control of PDEs

Example plant

$$u_t = u_{xx} + \lambda_0 u \qquad x \in (0,1)$$

$$u(0) = 0$$

Control

$$u(1) = \int_0^1 k(1, y)\, u(y)\, dy$$

Transformation

$$w(x) = u(x) - \int_0^x k(x, y)\, u(y)\, dy$$

Target system

$$w_t = w_{xx} - c\, w \qquad c \geq 0$$

$$w(0) = w(1) = 0$$

Kernel PDE

$$k_{xx}(x,y) - k_{yy}(x,y) = \lambda k(x,y) \qquad \lambda = \lambda_0 + c$$

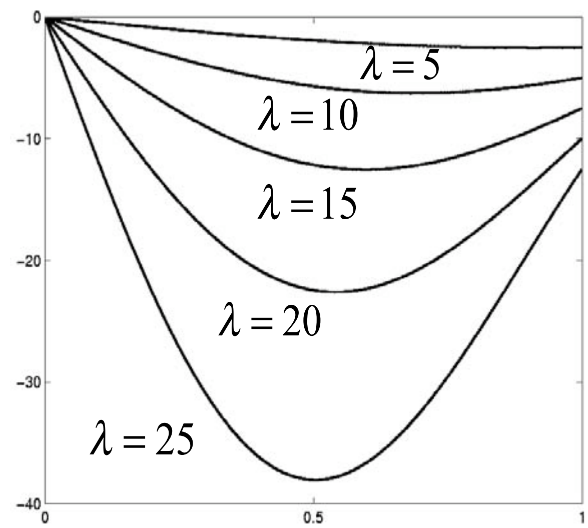
$$k(x,0) = 0$$

$$k(x,x) = -\frac{\lambda}{2}x$$

Bessel function

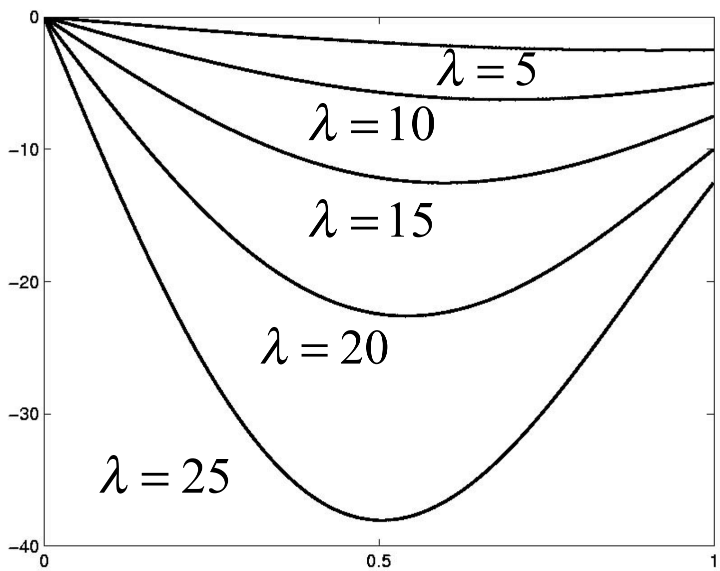
closed form solution

$$k(x,y) = -\lambda y \frac{I_1\left(\sqrt{\lambda(x^2-y^2)}\right)}{\sqrt{\lambda(x^2-y^2)}}$$

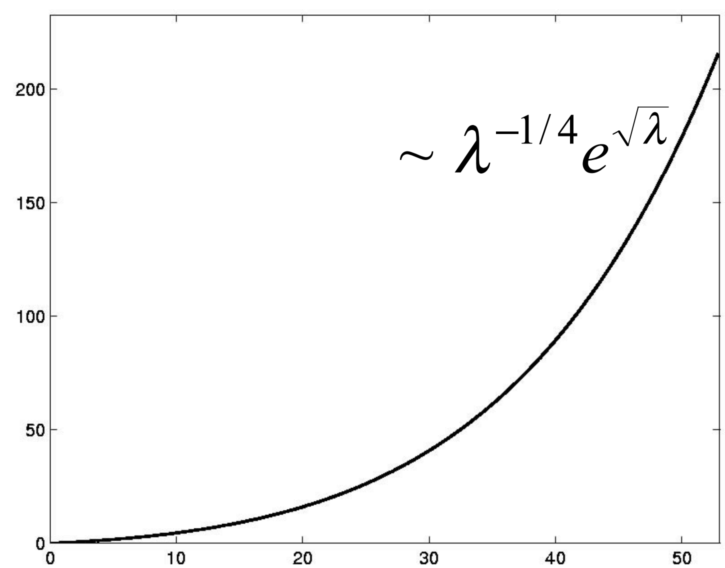


… the Kernel

total gain $$E = -\int_0^1 k(1,y)\,dy = I_0(\sqrt{\lambda}) - 1$$



Dependence of the kernel on level of instability



Amount of total gain as a function of λ

Unstable Heat Equation Example

plant

$$u_t(x,t) = u_{xx}(x,t) + \lambda_0 u(x,t)$$
$$u(0,t) = 0$$

controller

$$u(1,t) = -\int_0^1 \lambda \frac{I_1\left(\sqrt{\lambda(1-y^2)}\right)}{\sqrt{\lambda(1-y^2)}} u(y,t)\,dy \qquad \lambda = \lambda_0 + c$$

closed-loop solution

$$u(x,t) = \sum_{n=1}^{\infty} e^{-\overbrace{\left(c+\pi^2 n^2\right)}^{\text{eigenvalue}} t} \overbrace{\frac{2\pi n}{\sqrt{\lambda + \pi^2 n^2}} \sin\left(\sqrt{\lambda + \pi^2 n^2}\, x\right)}^{\text{eigenfunction}}$$
$$\times \int_0^1 \left(\sin(\pi n \xi) - \int_\xi^1 \xi \frac{I_1\left(\sqrt{\lambda(\eta^2 - \xi^2)}\right)}{\sqrt{\lambda(\eta^2 - \xi^2)}} \sin(\pi n \eta)\,d\eta \right) u_0(\xi)\,d\xi$$

UCSD Jacobs | Mechanical and Aerospace Engineering

Adaptive Control

Plant

$$u_t = u_{xx} + \lambda u$$

Control

$$u(1) = -\int_0^1 \hat{\lambda} y \frac{I_1\left(\sqrt{\hat{\lambda}\left(1-y^2\right)}\right)}{\sqrt{\hat{\lambda}\left(1-y^2\right)}} u(y)\,dy$$

Transformation

$$w(x) = u(x) + \int_0^x \hat{\lambda} y \frac{I_1\left(\sqrt{\hat{\lambda}\left(x^2-y^2\right)}\right)}{\sqrt{\hat{\lambda}\left(x^2-y^2\right)}} u(y)\,dy$$

"Observer"

$$\hat{w}_t = \hat{w}_{xx} + \dot{\hat{\lambda}} \int_0^x \frac{y}{2} w(y)\,dy + (w(x) - \hat{w}(x)) \int_0^1 w(y)^2\,dy$$

Parameter Update Law

$$\dot{\hat{\lambda}} = \int_0^1 (w(x) - \hat{w}(x)) w(x)\,dx$$

Inverse Optimal Stabilization

Theorem

The control law

$$u_x^*(1) = -\frac{1}{R}\left(u(1) + \int_0^1 \lambda \frac{I_1\left(\sqrt{\lambda(1-y^2)}\right)}{\sqrt{\lambda(1-y^2)}} u(y)\,dy \right)$$

stabilizes the system in $L_2(0,1)$ and **minimizes** the cost

$$J(u) = \int_0^\infty \left(Q(u(t)) + R\,u_x(1,t)^2 \right) dt$$

where R is sufficiently small and

$$Q(u) \geq \int_0^1 w_x (u(x))^2\,dx$$

Observer

plant

$$u_t(x,t) = u_{xx}(x,t) + \lambda_0 u(x,t)$$

$$u(0,t) = 0$$

$$\lambda = \lambda_0 + c$$

$$u_x(1,t) = U(t)$$

Observer

plant

$$u_t(x,t) = u_{xx}(x,t) + \lambda_0 u(x,t)$$

$$u(0,t) = 0$$

$$u(1,t) = \text{sensor}$$

$$\lambda = \lambda_0 + c$$

Observer

plant

$$u_t(x,t) = u_{xx}(x,t) + \lambda_0\, u(x,t)$$

$$u(0,t) = 0$$

$$\lambda = \lambda_0 + c$$

$u(1,t) =$ sensor

observer

$$\hat{u}_t(x,t) = \hat{u}_{xx}(x,t) + \lambda_0\, \hat{u}(x,t)$$ copy of the system $$+\text{"}L\left(y - C\hat{x}\right)\text{"}$$

Observer

plant

$$u_t(x,t) = u_{xx}(x,t) + \lambda_0\, u(x,t)$$

$$u(0,t) = 0$$

$$\lambda = \lambda_0 + c$$

$$u(1,t) = \text{sensor}$$

observer

$$\hat{u}_t(x,t) = \hat{u}_{xx}(x,t) + \lambda_0\, \hat{u}(x,t) \; + L(x)\left[u(1,t) - \hat{u}(1,t)\right] \quad +\text{"}L\left(y - C\hat{x}\right)\text{"}$$

Observer

plant

$$u_t(x,t) = u_{xx}(x,t) + \lambda_0 u(x,t)$$

$$u(0,t) = 0$$

$$u(1,t) = \text{sensor}$$

$$\lambda = \lambda_0 + c$$

observer

$$\hat{u}_t(x,t) = \hat{u}_{xx}(x,t) + \lambda_0 \hat{u}(x,t) + \underbrace{\frac{\lambda x}{1-x^2} I_2\left(\sqrt{\lambda(1-x^2)}\right)}_{\text{output injection gain function}} \underbrace{[u(1,t) - \hat{u}(1,t)]}_{\text{output estimation error}}$$

$$\hat{u}(0,t) = 0$$

$$\hat{u}_x(1,t) = U(t) + \underbrace{\frac{\lambda}{2}}[u(1,t) - \hat{u}(1,t)]$$

controller

$$u_x(1,t) = U(t) = -\frac{\lambda}{2} u(1,t) - \int_0^1 \frac{\lambda y}{1-y^2} I_2\left(\sqrt{\lambda(1-y^2)}\right) \hat{u}(y,t)\,dy$$

Output Feedback: Explicit Solution

$$u(x,t)=\sum_{n=0}^{\infty} e^{-(c+(\pi n+\pi/2)^2)t}\phi_n(x)$$

$$\times\left(\int_0^1 \psi_n(\xi)u_0(\xi)d\xi-(-1)^n\left(tC_n+\sum_{m=0,m\neq n}^{\infty}\frac{1-e^{\pi^2(n-m)(n+m+1)t}}{\pi^2(n-m)(n+m+1)}C_m\right)\right)$$

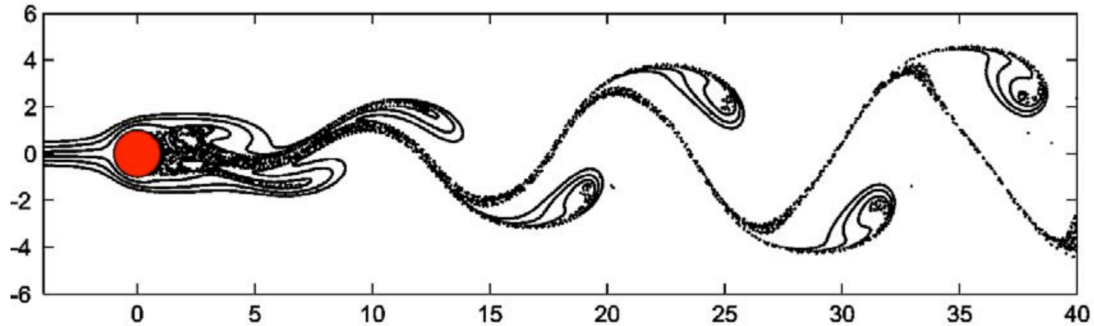
$$\phi_n(x)=\frac{2\pi n+\pi}{\sqrt{\lambda+(\pi n+\pi/2)^2}}\sin\left(\sqrt{\lambda+(\pi n+\pi/2)^2}\,x\right)$$

$$\psi_n(x)=\sin((\pi n+\pi/2)x)+\int_x^1 \lambda x\frac{I_1\left(\sqrt{\lambda(\xi^2-x^2)}\right)}{\sqrt{\lambda(\xi^2-x^2)}}\sin((\pi n+\pi/2)\xi)\,d\xi$$

$$C_n=\left(\int_0^1 \frac{\lambda\xi}{1-\xi^2}I_2\left(\sqrt{\lambda(1-\xi^2)}\right)\psi_n(\xi)\,d\xi\right)\left(\int_0^1 \phi_n(\xi)(u_0(\xi)-\hat{u}_0(\xi))\,d\xi\right)$$

UCSD Jacobs | Mechanical and Aerospace Engineering

Ginzburg-Landau Model of Vortex Shedding



UCSD Jacobs | Mechanical and Aerospace Engineering

Ginzburg-Landau Model of Vortex Shedding

$$\rho_t = a_R \rho_{xx} - a_I \iota_{xx} + \left(b_R(x) + c_R(x)\,(\rho^2+\iota^2)\right)\rho$$
$$- \left(b_I(x) + c_I(x)\,(\rho^2+\iota^2)\right)\iota$$

$$\iota_t = a_I \rho_{xx} + a_R \iota_{xx} + \left(b_I(x) + c_I(x)\,(\rho^2+\iota^2)\right)\rho$$
$$+ \left(b_R(x) + c_R(x)\,(\rho^2+\iota^2)\right)\iota$$

for $x \in (0,1)$ **with boundary conditions**

$$\rho(0,t) = 0, \quad \iota(0,t) = 0$$

UCSD Jacobs | Mechanical and Aerospace Engineering

Observer

$$\hat{\rho}_t = a_R \hat{\rho}_{xx} - a_I \hat{\iota}_{xx} + \left(b_R(x) + c_R(x)(\hat{\rho}^2 + \hat{\iota}^2) \right) \hat{\rho}$$
$$- \left(b_I(x) + c_I(x)(\hat{\rho}^2 + \hat{\iota}^2) \right) \hat{\iota}$$
$$+ p_1(x) \left(\rho(1,t) - \hat{\rho}(1,t) \right)$$
$$+ p_{c,1}(x) \left(\iota(1,t) - \hat{\iota}(1,t) \right)$$

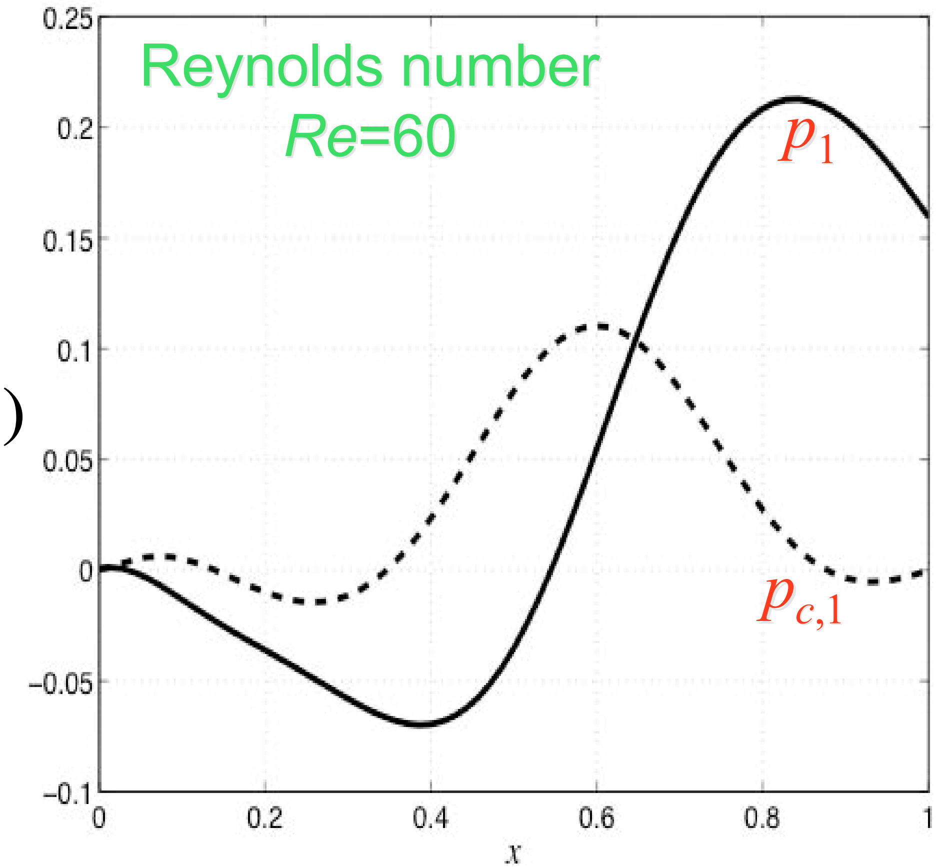
$$\hat{\iota}_t = a_I \hat{\rho}_{xx} + a_R \hat{\iota}_{xx} + \left(b_I(x) + c_I(x)(\hat{\rho}^2 + \hat{\iota}^2) \right) \hat{\rho}$$
$$+ \left(b_R(x) + c_R(x)(\hat{\rho}^2 + \hat{\iota}^2) \right) \hat{\iota}$$
$$- p_{c,1}(x) \left(\rho(1,t) - \hat{\rho}(1,t) \right)$$
$$+ p_1(x) \left(\iota(1,t) - \hat{\iota}(1,t) \right)$$

$$\hat{\rho}(0,t) = 0, \quad \hat{\iota}(0,t) = 0$$

Observer

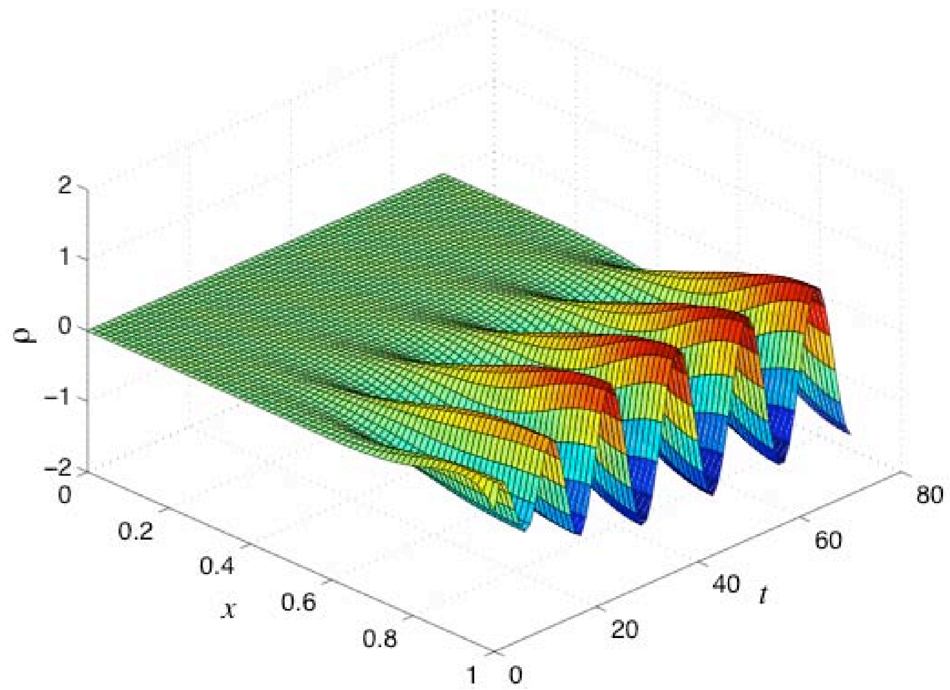
$$\hat{\rho}_t = a_R\hat{\rho}_{xx} - a_I\hat{\iota}_{xx} + \left(b_R(x) + c_R(x)(\hat{\rho}^2 + \hat{\iota}^2)\right)\hat{\rho}$$
$$- \left(b_I(x) + c_I(x)(\hat{\rho}^2 + \hat{\iota}^2)\right)\hat{\iota}$$
$$+ p_1(x)\left(\rho(1,t) - \hat{\rho}(1,t)\right)$$
$$+ p_{c,1}(x)\left(\iota(1,t) - \hat{\iota}(1,t)\right)$$

$$\hat{\iota}_t = a_I\hat{\rho}_{xx} + a_R\hat{\iota}_{xx} + \left(b_I(x) + c_I(x)\right.$$
$$+ \left(b_R(x) + c_R(x)(\hat{\rho}^2 + \hat{\iota}^2)\right)\hat{\iota}$$
$$- p_{c,1}(x)\left(\rho(1,t) - \hat{\rho}(1,t)\right)$$
$$+ p_1(x)\left(\iota(1,t) - \hat{\iota}(1,t)\right)$$

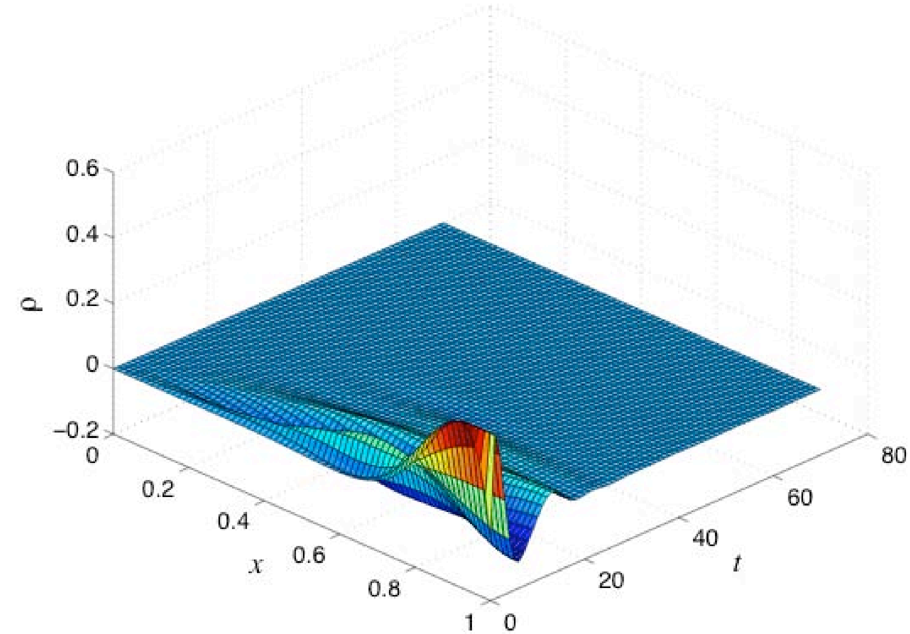


$$\hat{\rho}(0,t) = 0, \quad \hat{\iota}(0,t) = 0$$

Observer Simulation - No Control



Open-loop



Observer error

UCSD Jacobs | Mechanical and Aerospace Engineering

Control law

$$\rho_x(1,t) = \int_0^1 \left(k_x(1,y)\hat{\rho}(y,t) + k_{c,x}(1,y)\hat{\imath}(y,t) \right) dy$$
$$+ k(1,1)\rho(1,t) + k_c(1,1)\imath(1,t)$$

$$\imath_x(1,t) = \int_0^1 \left(-k_{c,x}(1,y)\hat{\rho}(y,t) + k_x(1,y)\hat{\imath}(y,t) \right) dy$$
$$- k_c(1,1)\rho(1,t) + k(1,1)\imath(1,t)$$

$$\hat{\rho}_x(1,t) = \rho_x(1,t)$$
$$\hat{\imath}_x(1,t) = \imath_x(1,t)$$

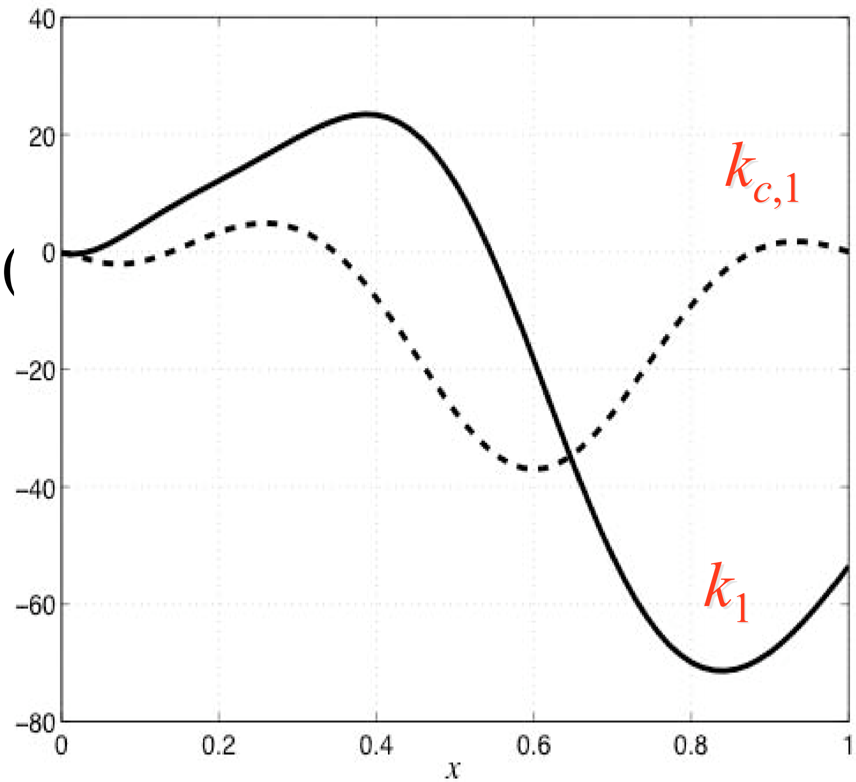
Control law

$$\rho_x(1,t) = \int_0^1 \Big(k_x(1,y)\hat{\rho}(y,t) + k_{c,x}(1,y)\hat{\iota}(y,t) \Big) dy$$

$$+ k(1,1)\rho(1,t) + k_c(1,1)\iota(1,t)$$

$$\iota_x(1,t) = \int_0^1 \Big(-k_{c,x}(1,y)\hat{\rho}(y,t) + k_x(1,y)\hat{\iota}($$

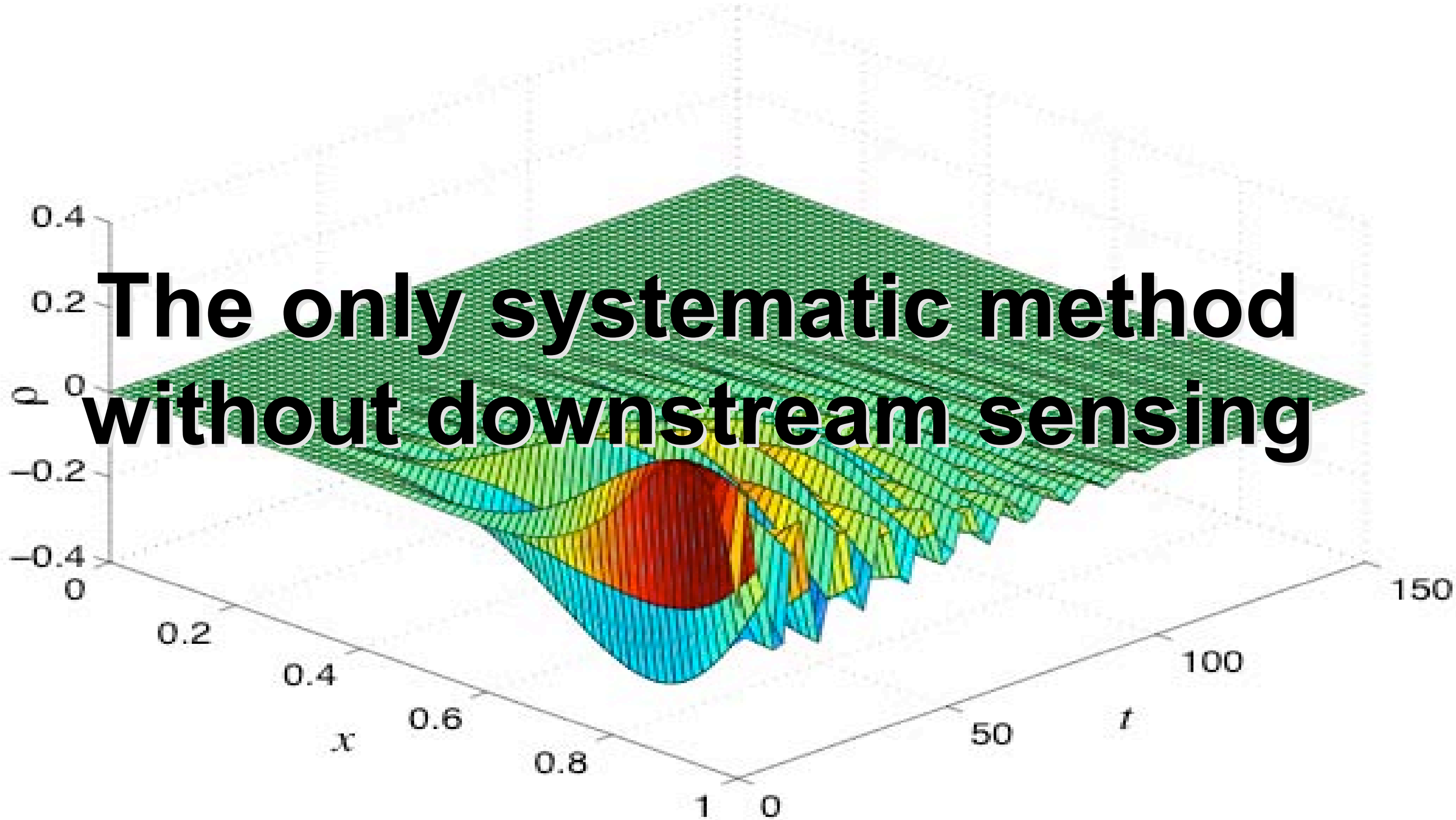
$$- k_c(1,1)\rho(1,t) + k(1,1)\iota(1,t)$$



$$\hat{\rho}_x(1,t) = \rho_x(1,t)$$

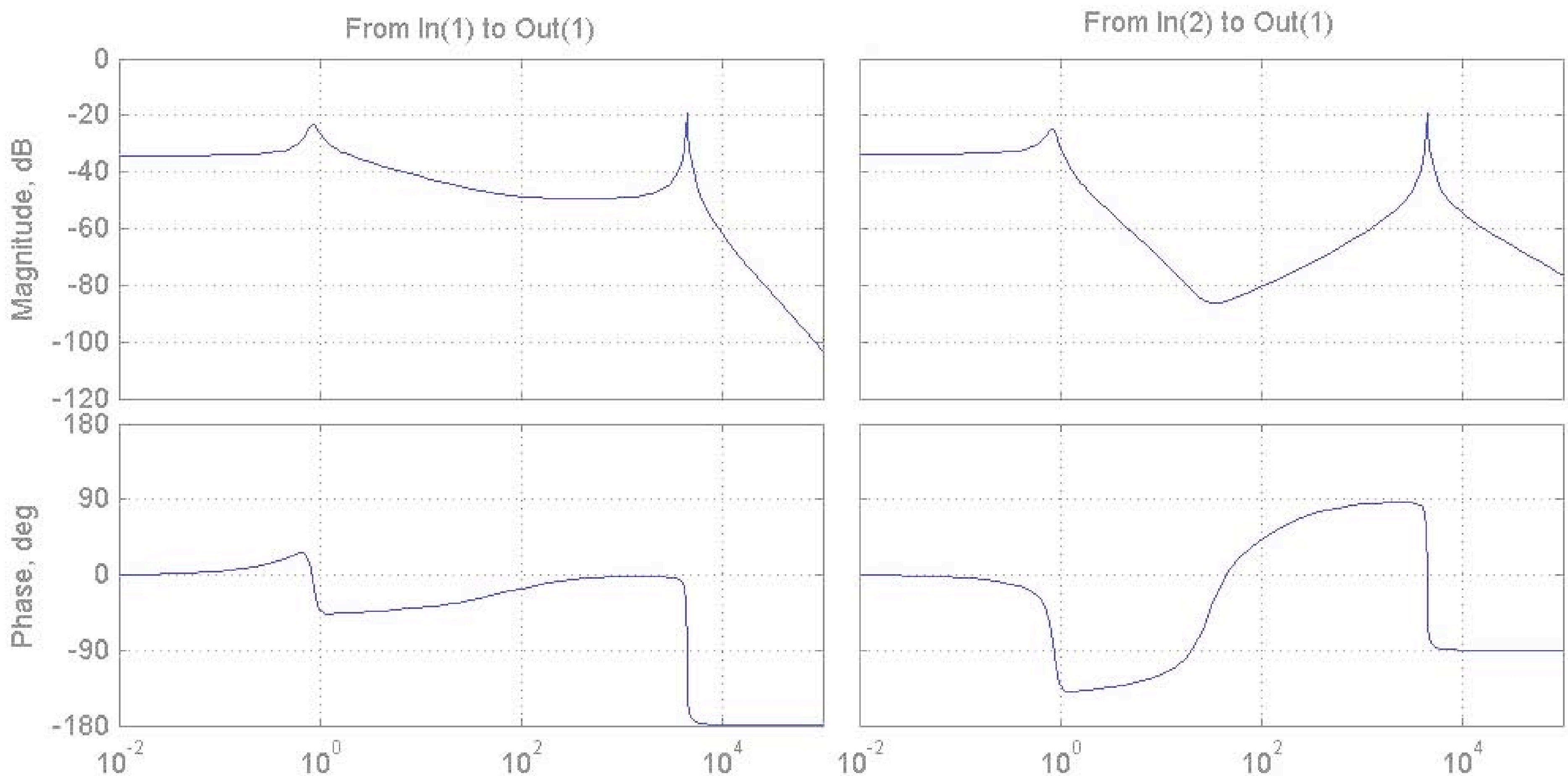
$$\hat{\iota}_x(1,t) = \iota_x(1,t)$$

Closed loop simulation



The only systematic method without downstream sensing

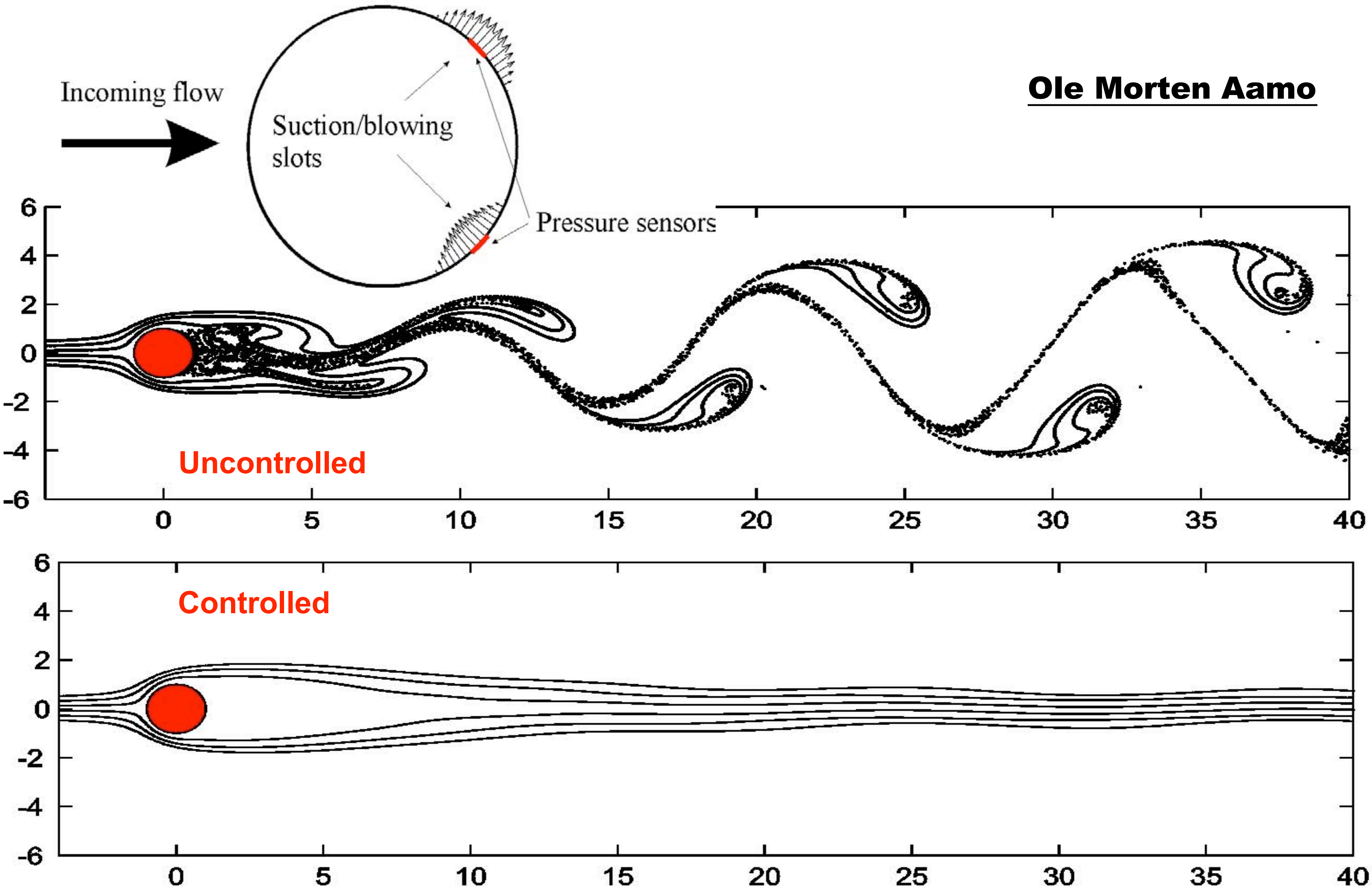
Controller transfer functions: from pressure sensing to velocity actuation



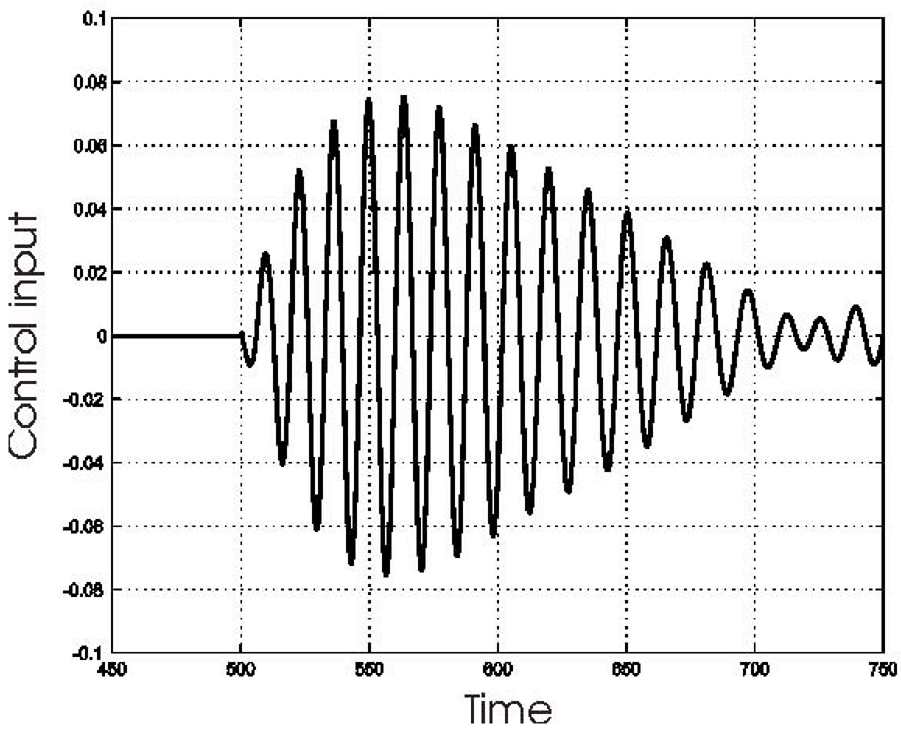
Stable, min-phase transfer functions, can be approximated by order 10

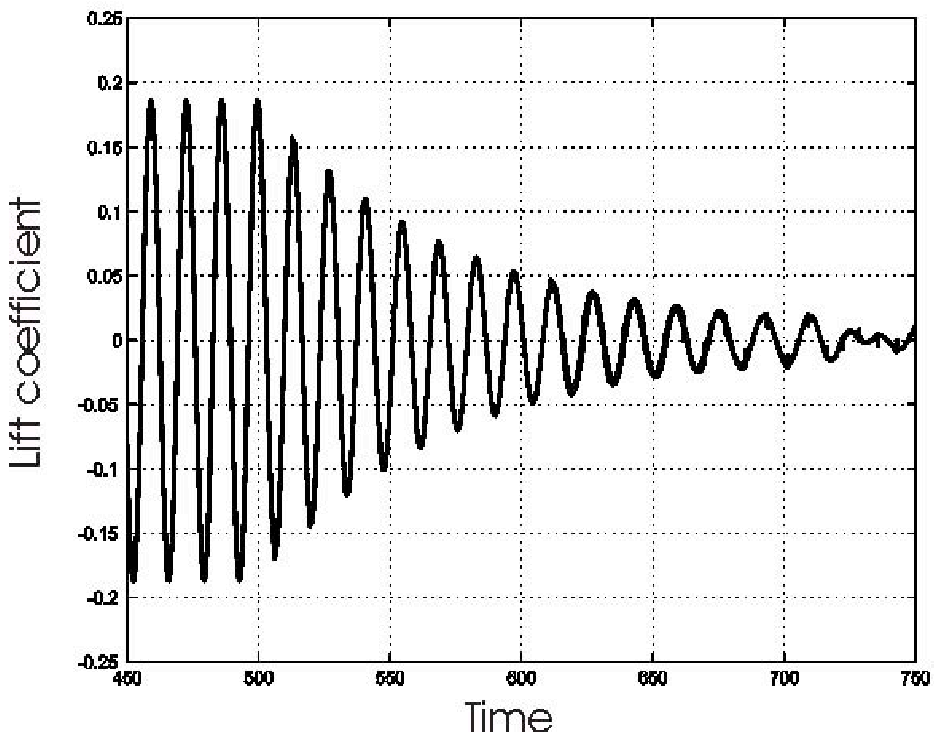
2D (Navier-Stokes) View

Ole Morten Aamo



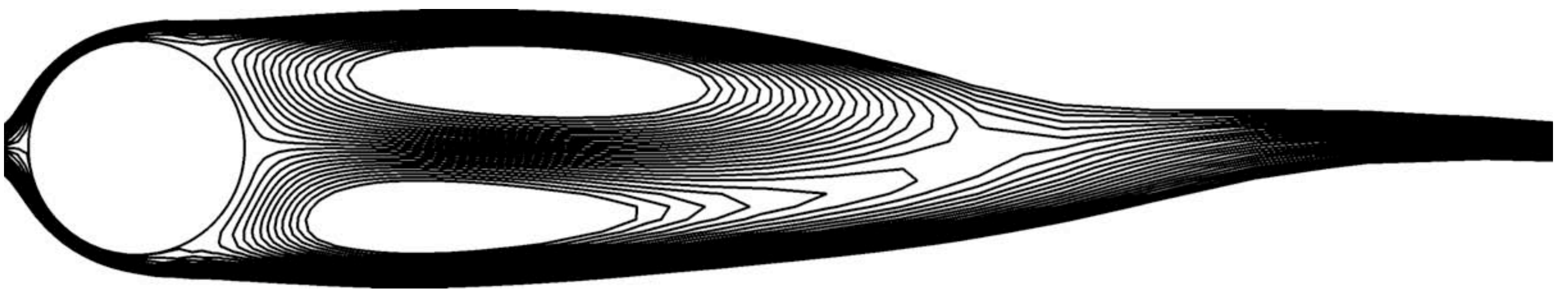
Control Effort and Lift





UCSD Jacobs | Mechanical and Aerospace Engineering

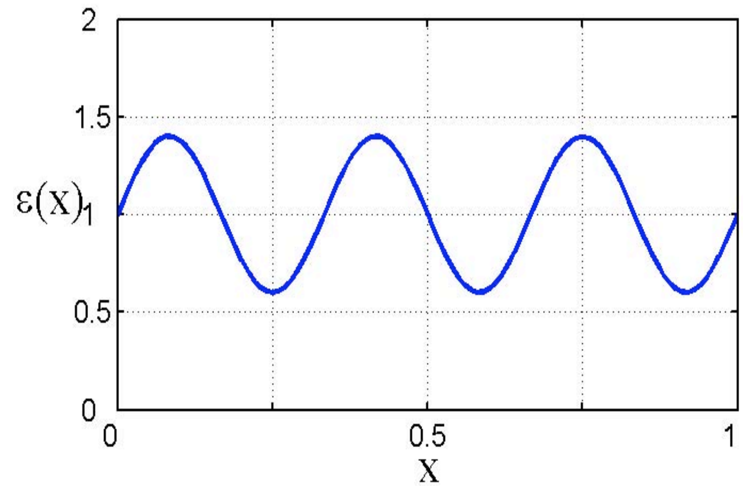
Equilibrium Profile Identified by Control



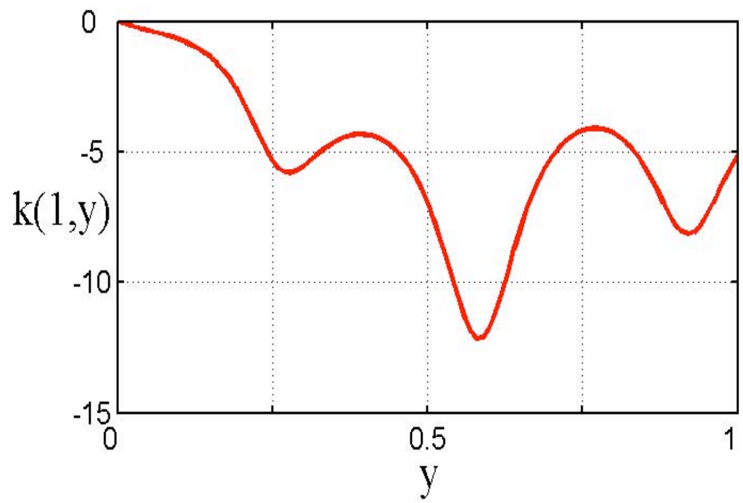
Stabilized flow by feedback control reveals the underlying **equilibrium profile** and separation point!

Arbitrary spatially-dependent diffusion

$$u_t(x,t) = \varepsilon(x) u_{xx}(x,t) + \lambda_0 u(x,t)$$
$$u(0,t) = 0$$

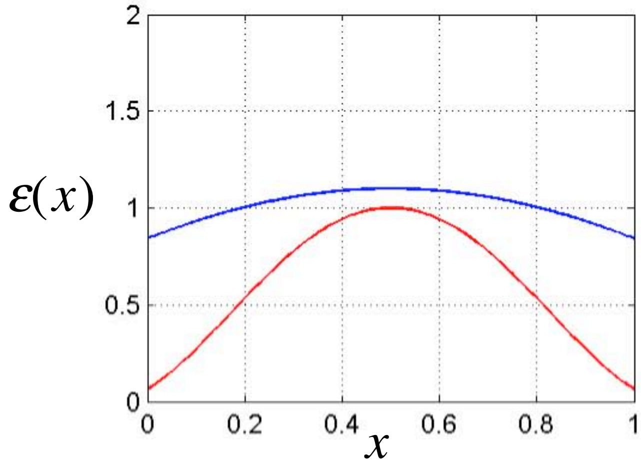


$$\varepsilon(x) k_{xx} - (\varepsilon(y) k)_{yy} = \lambda_0 k$$
$$k(x,0) = 0$$
$$k(x,x) = -\frac{1}{2\sqrt{\varepsilon(x)}} \int_0^x \frac{\lambda_0}{\sqrt{\varepsilon(\xi)}} d\xi$$

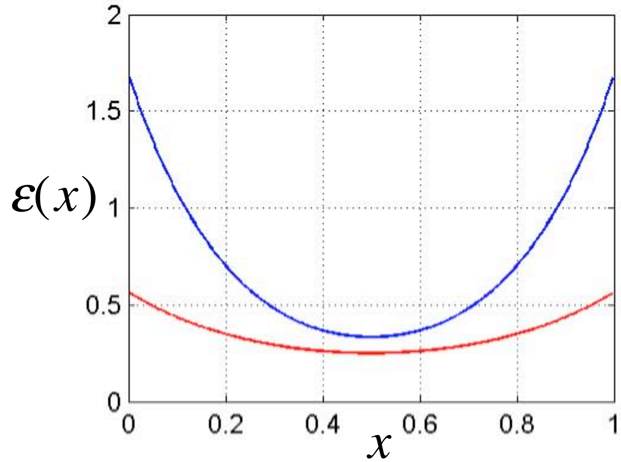


UCSD Jacobs | Mechanical and Aerospace Engineering

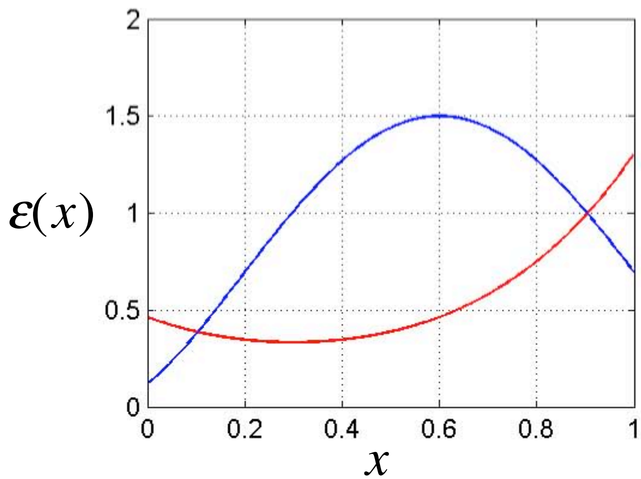
Diffusion parametrization $\varepsilon(x) = \varepsilon_0 \left(1 + \theta\left(x - x^*\right)^2\right)^2$



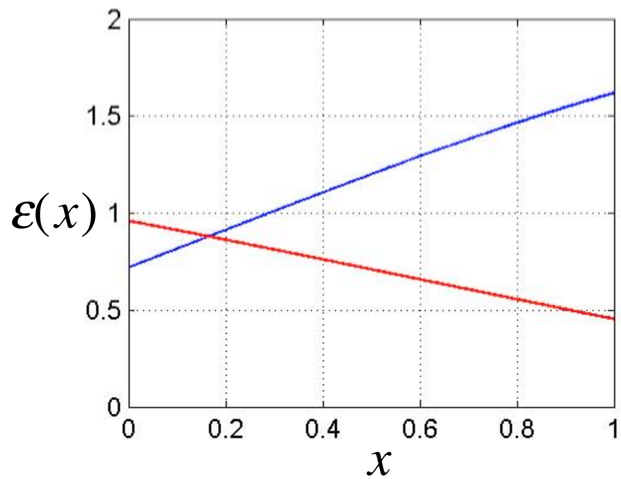
Extremum has arbitrary value and flatness



Peak or well



Extremum has arbitrary location



Linear functions easily matched

… explicit gain kernel

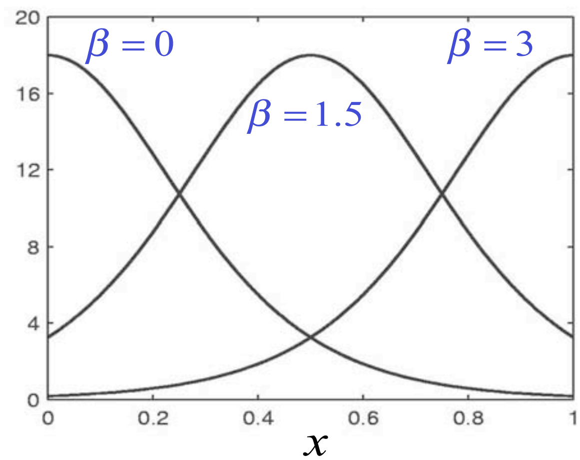
$$k_1(y) = \varepsilon^{1/4}(1)\sqrt{\lambda_0 + c}\,\frac{\varphi(y)}{\varepsilon^{3/4}(y)}\,\frac{I_1\left(\sqrt{\varepsilon^{-1}(0)\left(\lambda_0 + c\right)\left(\varphi(1)^2 - \varphi(y)^2\right)}\right)}{\sqrt{\varphi(1)^2 - \varphi(y)^2}}$$

$$\varphi(\xi) = \frac{1 + \theta x^{*2}}{\sqrt{\theta}}\left(\arctan\left(\sqrt{\theta}(\xi - x^*)\right) + \arctan\left(\sqrt{\theta}x^*\right)\right)$$

Chemical Tubular Reactor Example

plant $$u_t(x,t) = u_{xx}(x,t) + \lambda(x)\, u(x,t)$$

$$\lambda(x) = \frac{2\alpha^2}{\cosh^2(\alpha x - \beta)}$$



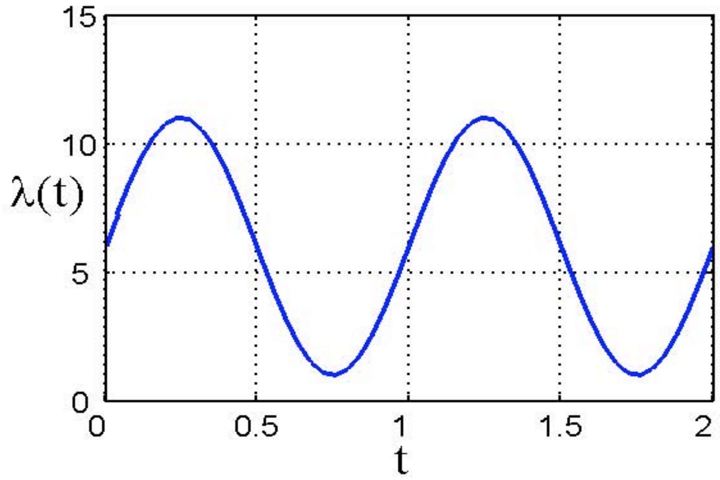
$\alpha = 3$

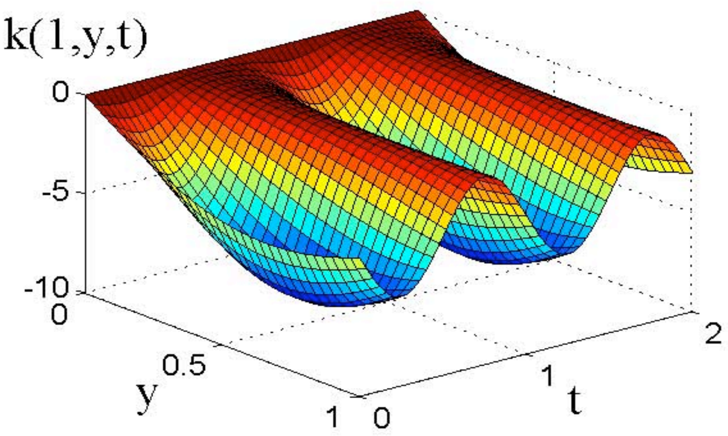
gain kernel $$k_1(y) = \alpha e^{\alpha \tanh(\beta(1-y))}(\tanh\beta - \tanh(\beta - \alpha y))$$

Time-dependent reactivity

$$u_t(x,t) = u_{xx}(x,t) + \lambda(t)u(x,t)$$
$$u(0,t) = 0$$
$$u(1,t) = \int_0^1 k(1,y,t)u(y,t)dy$$

$$k_t = k_{xx} - k_{yy} - \lambda(t)k$$
$$k(x,0,t) = 0$$
$$k(x,x,t) = -\frac{x}{2}\lambda(t)$$





UCSD Jacobs | Mechanical and Aerospace Engineering

Solid Rocket Propellant Example

plant

$$u_t(x,t) = u_{xx}(x,t) + g\,u(\mathbf{0},t)$$
$$u_x(0,t) = 0$$

gain kernel

$$k_1(y) = \sqrt{g}\,\sinh(\sqrt{g}(1-y))$$

closed-loop solution

$$u(x,t) = 2\sum_{n=0}^{\infty} e^{-\mu_n^2 t}\left(\cos(\mu_n x) - \frac{g}{\mu_n^2 + g}\right)$$
$$\times \int_0^1 \left(\cos(\mu_n \xi) + (-1)^n \frac{\sqrt{g}}{\mu_n}\sinh\left(\sqrt{g}(1-\xi)\right)\right) u_0(\xi)\,d\xi$$

$$\mu_n = \pi n + \frac{\pi}{2}$$

UCSD Jacobs | Mechanical and Aerospace Engineering

Compensator as a Transfer Function

plant

$$u_t(x,t) = u_{xx}(x,t) + g\,u(0,t)$$
$$u_x(0,t) = 0$$

compensator

$$u(1,s) = \frac{g}{s}\left(1 - \frac{(s-g)\cosh\sqrt{s}\cosh\sqrt{g}}{s\cosh\sqrt{s} - g\cosh\sqrt{g}}\right)u(0,s)$$

$$u(1,s) \approx 60\frac{s+17}{s^2+25s+320}u(0,s) \qquad (g=8)$$

PDEs w/ destabilizing boundary conditions

plant

$$u_t(x,t) = \varepsilon u_{xx}(x,t)$$

$$u_x(0,t) = q u(0,t)$$

$$q < 0$$

gain kernel

$$k_1(y) = -c\frac{I_1\left(\sqrt{c(1-y^2)}\right)}{\sqrt{c(1-y^2)}} + \frac{qc}{\sqrt{c+q^2}} \times$$

$$\times \int_0^1 e^{-q\eta/2} \sinh\left(\frac{\sqrt{c+q^2}}{2}\eta\right) I_1\left(\sqrt{c(1+y)(1-y-\eta)}\right) d\eta$$

$$c > \varepsilon q^2$$

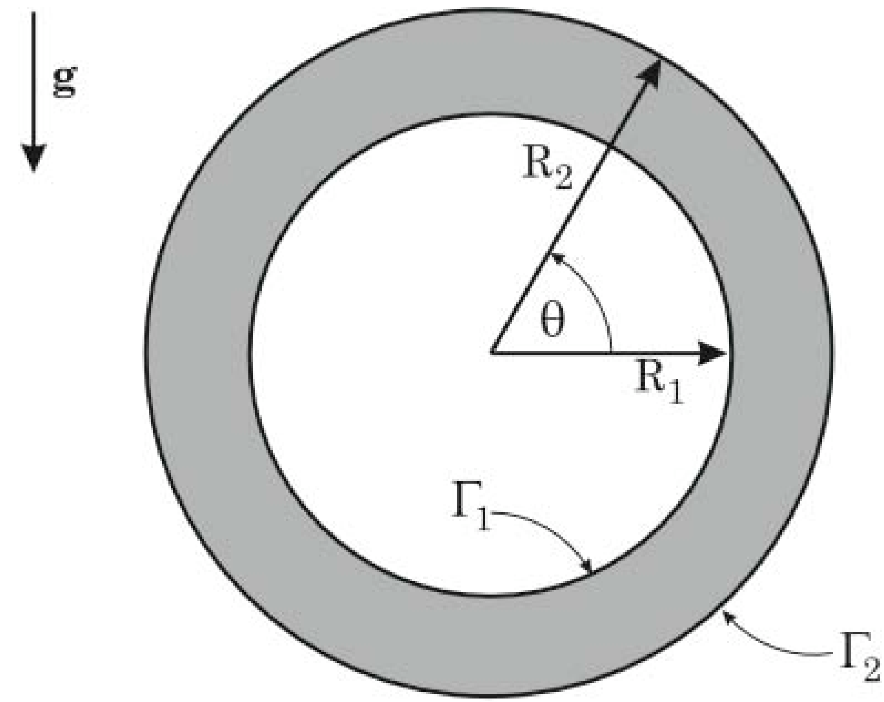
UCSD Jacobs | Mechanical and Aerospace Engineering

Combining Different Solutions

$$
\begin{tikzcd}
u_t = u_{xx} + (\lambda_{\alpha\beta}(x) + \lambda_0)u \arrow[d, "k^{\alpha\beta}(x,y)"] \arrow[dd, bend right, "k(x,y)"'] \\
w'_t = w'_{xx} + \lambda_0 w' \arrow[d, "k^{\lambda}(x,y)"] \\
w''_t = w''_{xx} - cw''
\end{tikzcd}
$$

$$
k(x,y) = k^{\alpha\beta}(x,y) + k^{\lambda}(x,y) - \int_y^x k^{\alpha\beta}(x,\tau)k^{\lambda}(\tau,y)\,d\tau
$$

Thermal Convection Loop (2D)



- Heated at the bottom, cooled at the top.
- Its discretization produces the "Lorenz attractor."

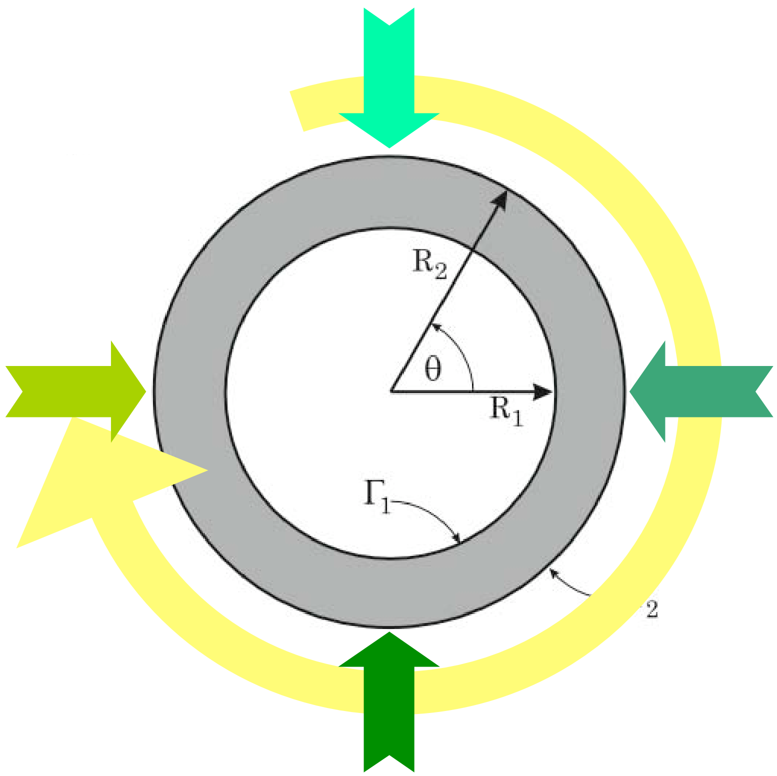
Rafael Vazquez

UCSD Jacobs | Mechanical and Aerospace Engineering

Actuation

$v(t, R_2) = V(t)$ - rotation of outer boundary

$T_r(t, R_2, \phi) = U(t, \phi)$ - heating/cooling the outer boundary



Model (linearized around equilibrium profile $T = Kr\sin\theta$)

$$\varepsilon v_t = v_{rr} + \frac{v_r}{r} - \frac{v}{r^2} + A_1 \int_0^{2\pi} \tau(r,\phi)\cos\phi \, d\phi$$

Quasi-steady state eqn: TPBVP in $v(r)$

$$\tau_t = \tau_{rr} + \frac{\tau_r}{r} + \frac{\tau_{\theta\theta}}{r^2} + A_2 v \cos\theta$$

ε small – singular perturbation form

- $v(r,t)$ – nondimensionalized azimuthal component of velocity
- $\tau(r,\phi,t)$ – nondimensionalized temperature fluctuation around equilibrium profile
- ε – thermal diffusivity/kinematic viscosity
- A_1, A_2 – parameters that depend on R_1, R_2, viscosity, thermal diffusivity, thermal gradient, and the coefficient of thermal expansion

Singular Perturbation Analysis - Quasi-Steady State

Want the solution to have a "strict-feedback form:"

General solution:

$$\text{Set } V = -\frac{A_1}{2}\int_{R_1}^{R_2}\int_0^{2\pi}\frac{R_2^2 - s^2}{R_2}\cos\phi\,\tau(t,s,\phi)\,ds\,d\phi$$

$$v(r) = \frac{r^2 - R_1^2}{R_2^2 - R_1^2}\frac{R_2}{r}\left(V + \frac{A_1}{2}\int_{R_1}^{R_2}\int_0^{2\pi}\frac{R_2^2 - s^2}{R_2}\cos\phi\,\tau(t,s,\phi)\,ds\,d\phi\right)$$

$$-\frac{A_1}{2}\int_{R_1}^{r}\int_0^{2\pi}\frac{r^2 - s^2}{r}\cos\phi\,\tau(t,s,\phi)\,ds\,d\phi$$

QUASI-STEADY STATE

Singular Perturbation Analysis - Reduced Model

Control law (outer cylinder):

$$v(t,R_2) = -\frac{A_1}{2}\int_{R_1}^{R_2}\cos\phi\int_0^{2\pi}\frac{R_2^2 - s^2}{R_2}\tau(t,s,\phi)dsd\phi$$

Makes reduced model strict-feedback

$$\tau_t = \tau_{rr} + \frac{\tau_r}{r} + \frac{\tau_{\theta\theta}}{r^2} - A_1A_2\int_{R_1}^{r}\int_0^{2\pi}\frac{r^2 - s^2}{r}\cos\phi\cos\theta\,\tau(s,\phi)dsd\phi$$

Controller

Control law (outer cylinder):

$$v(t,R_2) = -\frac{A_1}{2}\int_{R_1}^{R_2}\cos\phi\int_0^{2\pi}\frac{R_2^2 - s^2}{R_2}\tau(t,s,\phi)ds d\phi$$

Makes reduced model strict-feedback

$$\tau_r(t,R_2,\theta) = q\tau(t,R_2,\theta) - \cos\theta\int_{R_1}^{R_2}\cos\phi\int_0^{2\pi}\sqrt{\frac{s}{R_2}}\left(\left(q+\frac{1}{2R_2}\right)\kappa(R_2,s) - \kappa_r(R_2,s)\right)\tau(t,s,\phi)ds d\phi$$

Stabilizes the reduced model

where $q = -1 - \frac{R_2}{4\left(R_2 - R_1\right)}$

The Kernel $\kappa(r,s)$

Kernel P(I)DE:

$$\kappa_{rr} - \kappa_{ss} = \frac{3}{4}\left(\frac{1}{r^2} - \frac{1}{s^2}\right)\kappa + A_1 A_2 \pi \int_s^r \frac{\rho^2 - s^2}{\sqrt{\rho s}} \kappa(r,\rho)\, d\rho - A_1 A_2 \frac{r^2 - s^2}{\sqrt{rs}}$$

$$\kappa_s(r, R_1) = \frac{1}{2R_1} \kappa(r, R_1)$$

$$\kappa(r,r) = 0$$

The Kernel $\kappa(r,s)$

Kernel P(I)DE:

$$\kappa_{rr} - \kappa_{ss} = \frac{3}{4}\left(\frac{1}{r^2} - \frac{1}{s^2}\right)\kappa + A_1 A_2 \pi \int_s^r \frac{\rho^2 - s^2}{\sqrt{\rho s}} \kappa(r,\rho)\, d\rho - A_1 A_2 \frac{r^2 - s^2}{\sqrt{rs}}$$

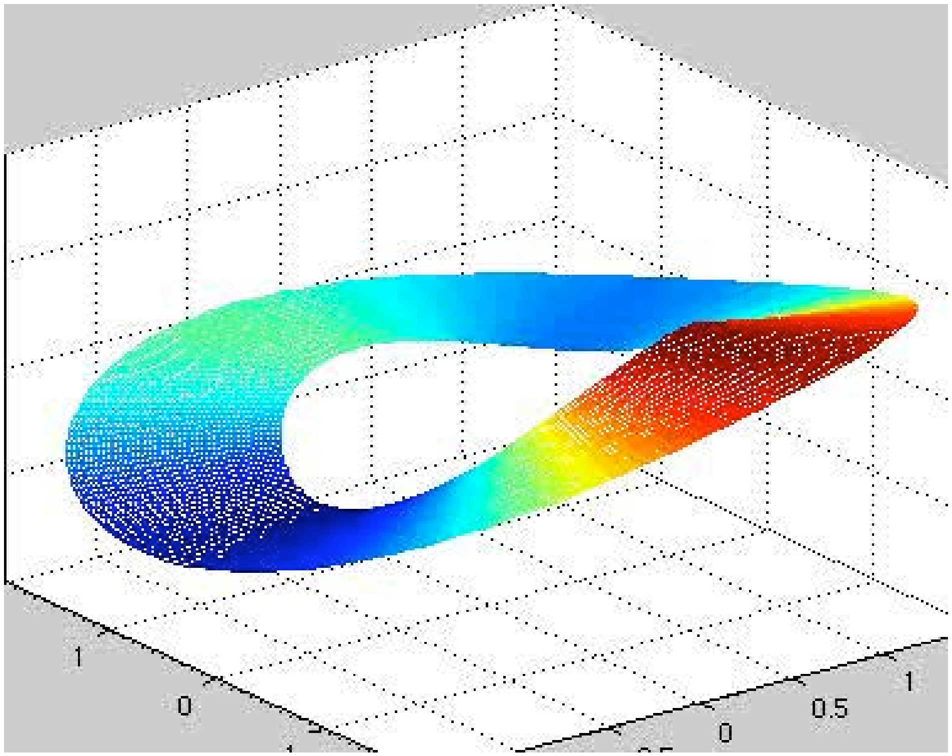
$$\kappa_s(r, R_1) = \frac{1}{2R_1}\kappa(r, R_1)$$

$$\kappa(r,r) = 0$$

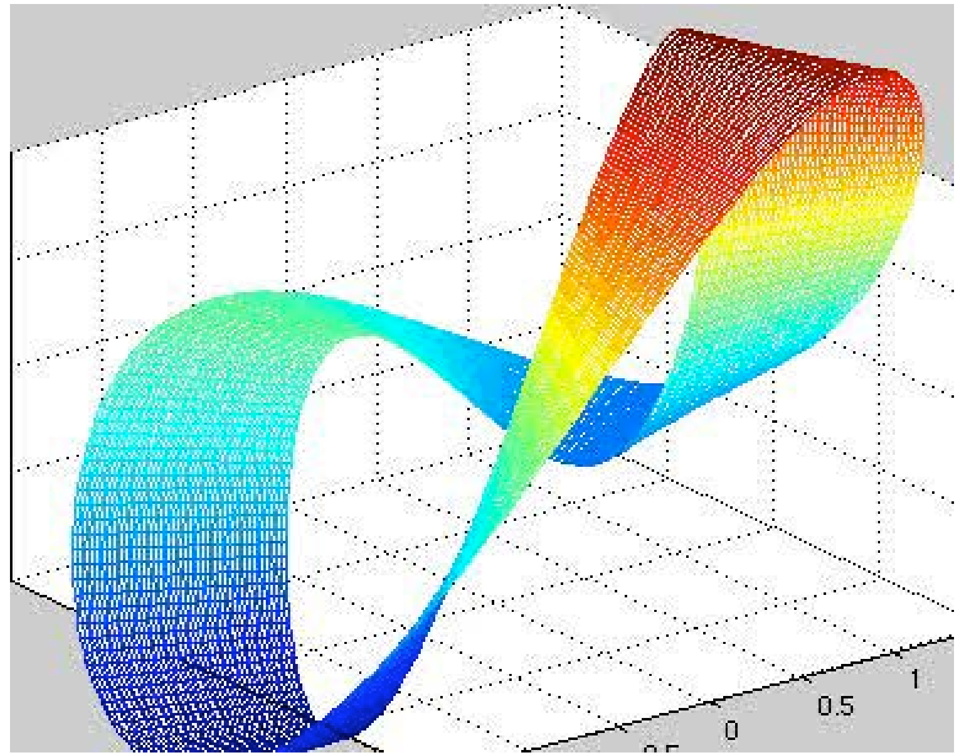
Explicit approximate solution:

$$\kappa\left(\frac{\xi+\eta}{2}, \frac{\xi-\eta}{2}\right) \approx -A_1 A_2 \left[\frac{1}{6}\left(\xi^3 - \eta^3 - \left(\xi^2 - \eta^2\right)^{\frac{3}{2}}\right) + \frac{5}{2}\sqrt{\pi} R_1^3 \mathrm{e}^{\left(1+\frac{\eta}{R_1}\right)}\left(\mathrm{erf}(1) - \mathbf{erf}\left(\sqrt{1+\frac{\eta}{R_1}}\right)\right)\right.$$
$$\left. + R_1^3\left(6\mathrm{e}^{\frac{\eta}{R_1}} - \frac{34}{3}\right) - 8R_1^2\eta - 2R_1\eta^2 + \frac{5}{3}\sqrt{R_1^2 + R_1\eta}\left(5R_1^2 + 2R_1\eta\right)\right]$$

Simulations



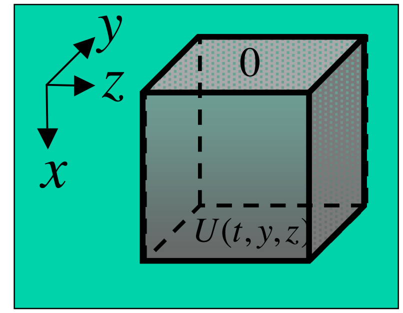
Uncontrolled



Controlled

3D

Plant: $$u_t = u_{xx} + u_{yy} + u_{zz} + \lambda(x,y,z)u$$



Kernel Equation:

$$k_{xx} + k_{yy} + k_{zz} = k_{\xi\xi} + k_{\eta\eta} + k_{\zeta\zeta}$$
$$+\left(\lambda(\xi,\eta,\zeta) - \lambda(x,y,z)\right)k(x,y,z,\xi,\eta,\zeta)$$
$$-\frac{\|\lambda\|_\infty + 2c}{\mu}\int_0^1\int_0^1 m\left(\frac{y-y'}{\mu},\frac{z-z'}{\mu}\right)k(x,y',z',\xi,\eta,\zeta)dy'dz'$$

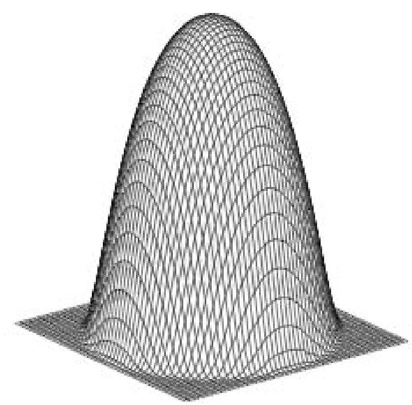
$$\mu < \frac{\sqrt{c}}{\|\lambda\|_\infty + 2c}$$

with B.C.: $$k(x,y,z,x,\eta,\zeta) = \frac{\|\lambda\|_\infty + 2c}{2\mu}m\left(\frac{y-\eta}{\mu},\frac{z-\zeta}{\mu}\right)x$$

Mollifier:

(approx. δ–fcn)

$$m(y,z) = \begin{cases} 2.25\exp\left(\dfrac{1}{y^2+z^2-1}\right), & y^2+z^2<1 \\ 0, & y^2+z^2 \geq 1 \end{cases}$$



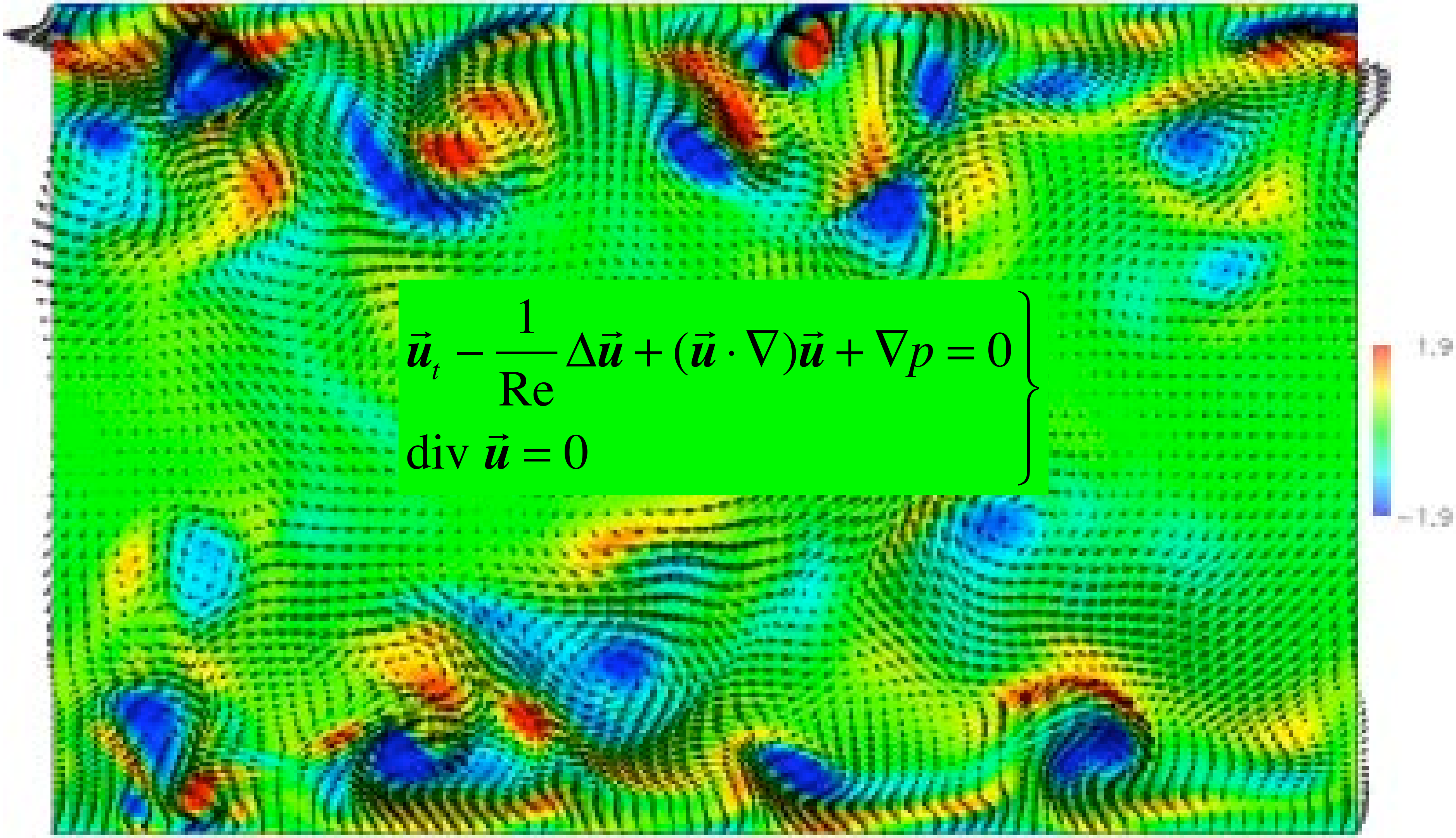
Summary of Features of Backstepping

- **Easier Analysis.** Design and existence of solutions analysis accessible with calculus.
- **Numerically More Manageable.** Our hyperbolic PDEs for kernels take ~20 times less time than Riccati equations even in 1D.
- Possibility of adaptive design

Open Problems

- **Hyperbolic PDEs.** Structures, acoustic, traffic models.
- **Nonlinear PDEs.** Spatial Volterra series.

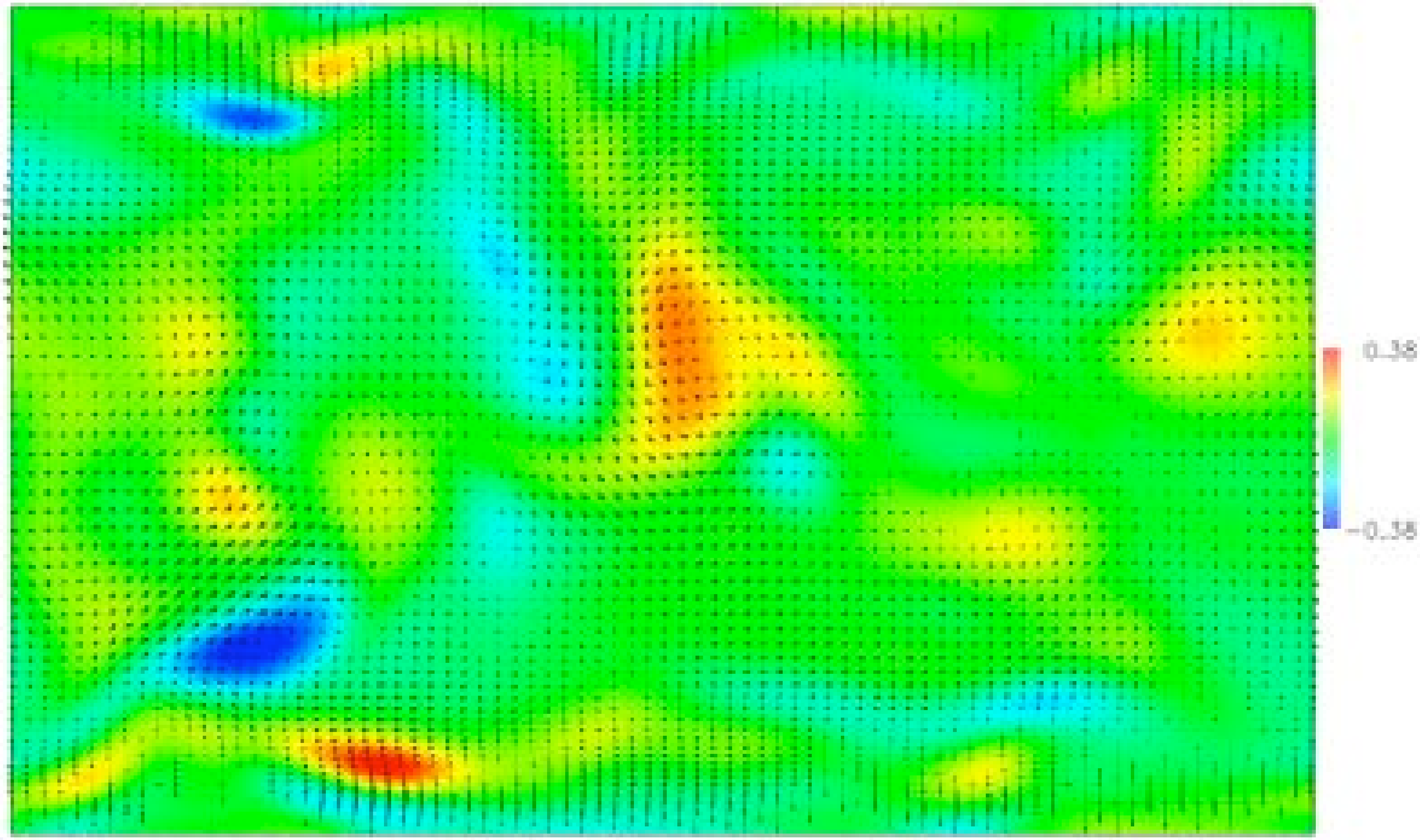
Conrol of Turbulence: Channel Flow



$$\left.\begin{aligned}&\vec{u}_t - \frac{1}{\mathrm{Re}}\Delta\vec{u} + (\vec{u}\cdot\nabla)\vec{u} + \nabla p = 0\\&\mathrm{div}\,\vec{u} = 0\end{aligned}\right\}$$

UCSD Jacobs | Mechanical and Aerospace Engineering

Stabilized Channel Flow

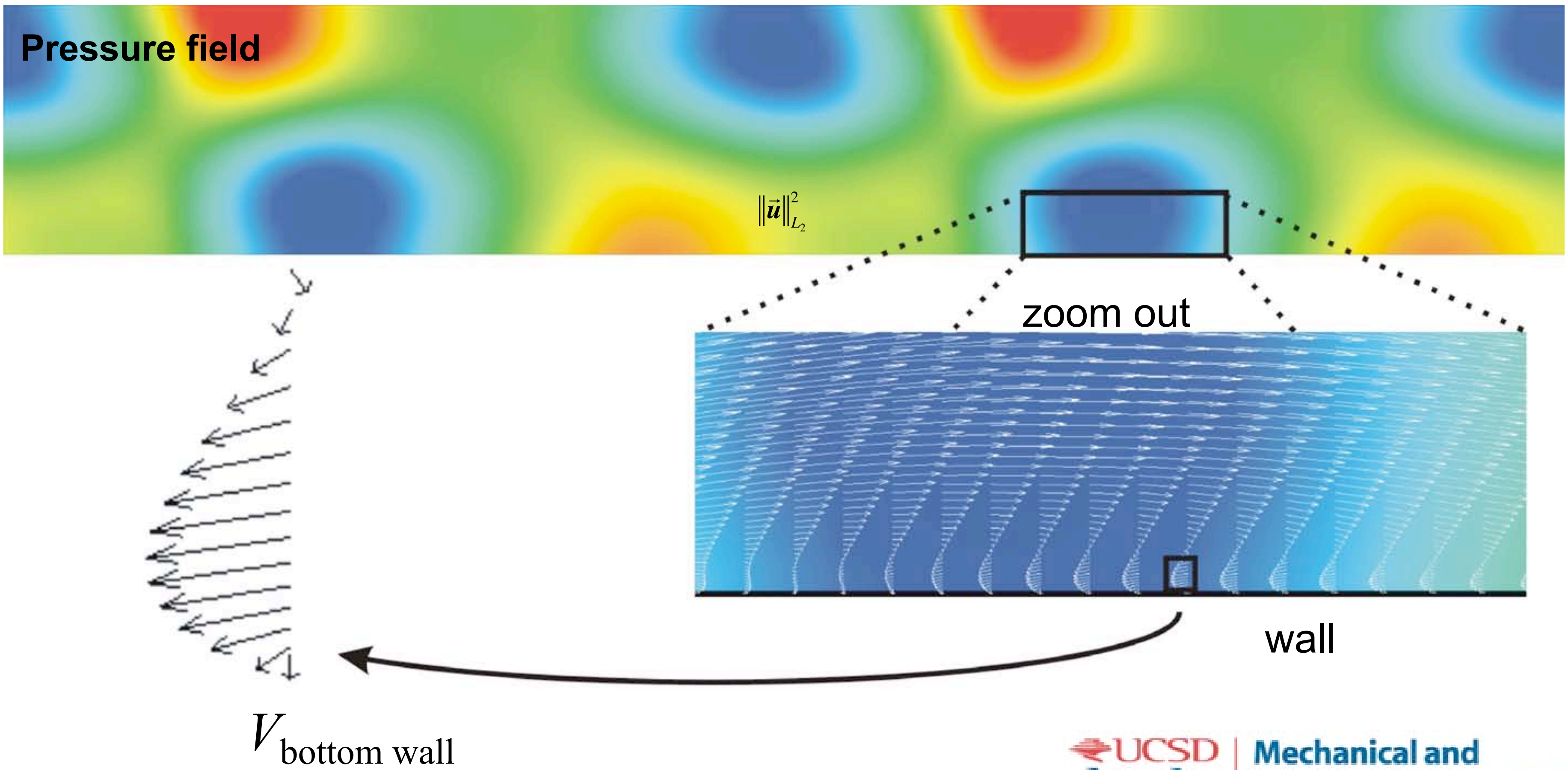


UCSD Jacobs | Mechanical and Aerospace Engineering

Control Law

"$\mathrm{L}_g V$" (Jurdjevic-Quinn) type controller w.r.t. $\|\vec{\boldsymbol{u}}\|_{L_2}^2$ as Lyap. fcn

$$V_{\text{bottom wall}}(x) = V_{\text{top wall}}(x) = k_P\left[P_{\text{top wall}}(x) - P_{\text{bottom wall}}(x)\right]$$



UCSD Jacobs | Mechanical and Aerospace Engineering

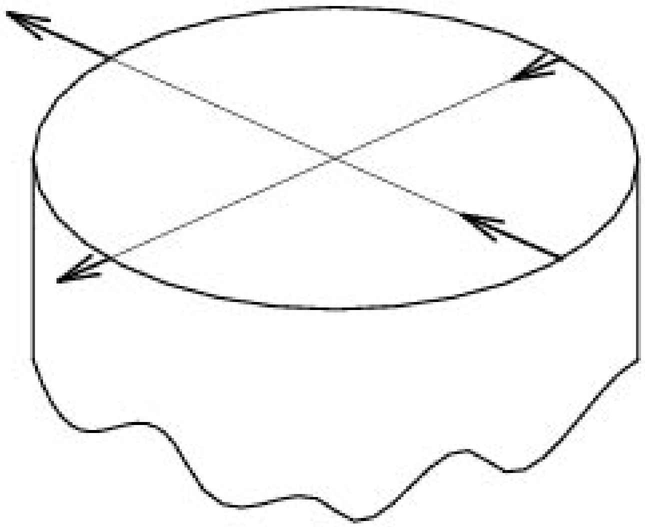
Turbulence Enhancement - Mixing

$$V_{\text{bottom wall}}(x) = V_{\text{top wall}}(x) = -k_P\left[P_{\text{top wall}}(x) - P_{\text{bottom wall}}(x)\right]$$

UCSD Jacobs | Mechanical and Aerospace Engineering

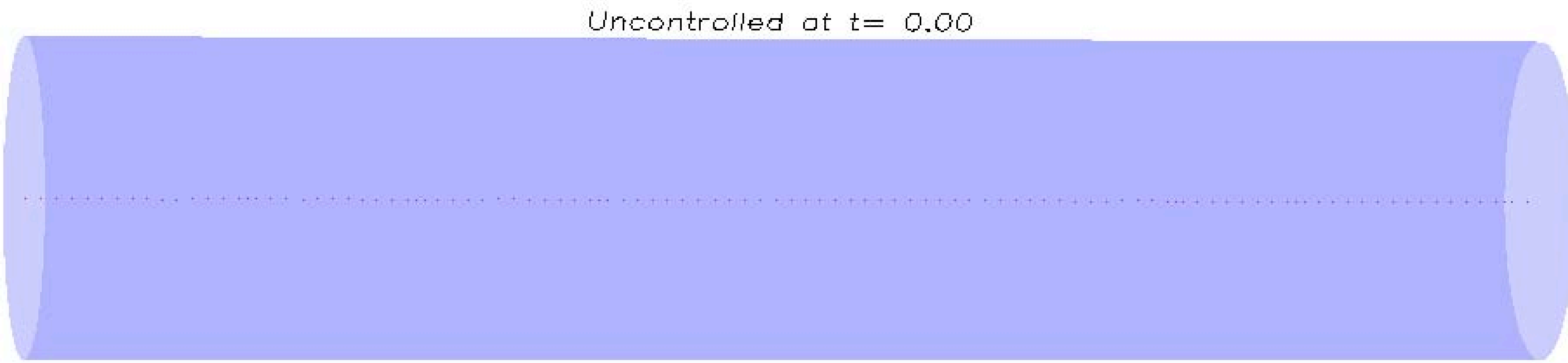
Pipe Flow ***Mixing*** Control

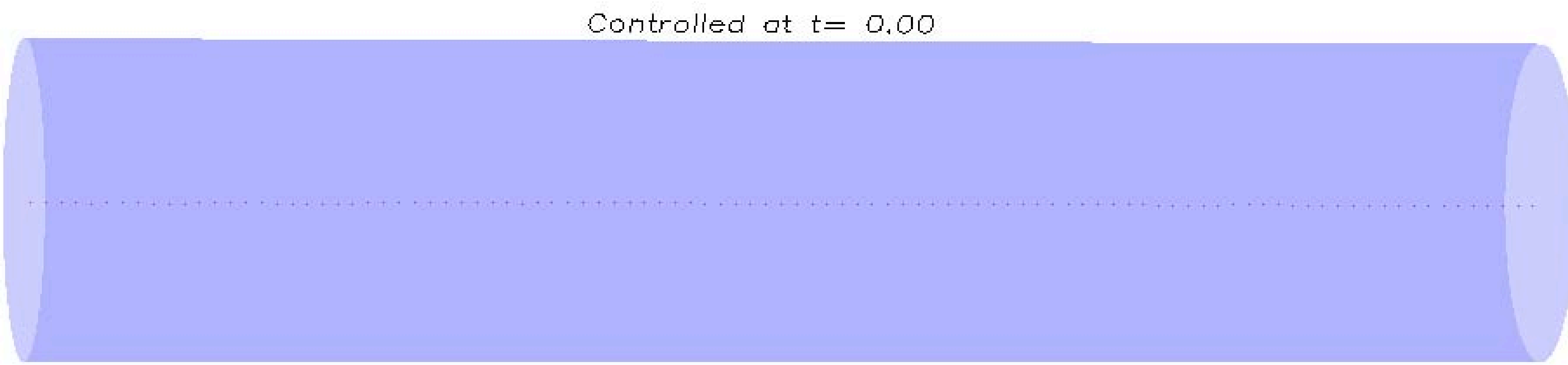
✓ Actuation: wall blowing



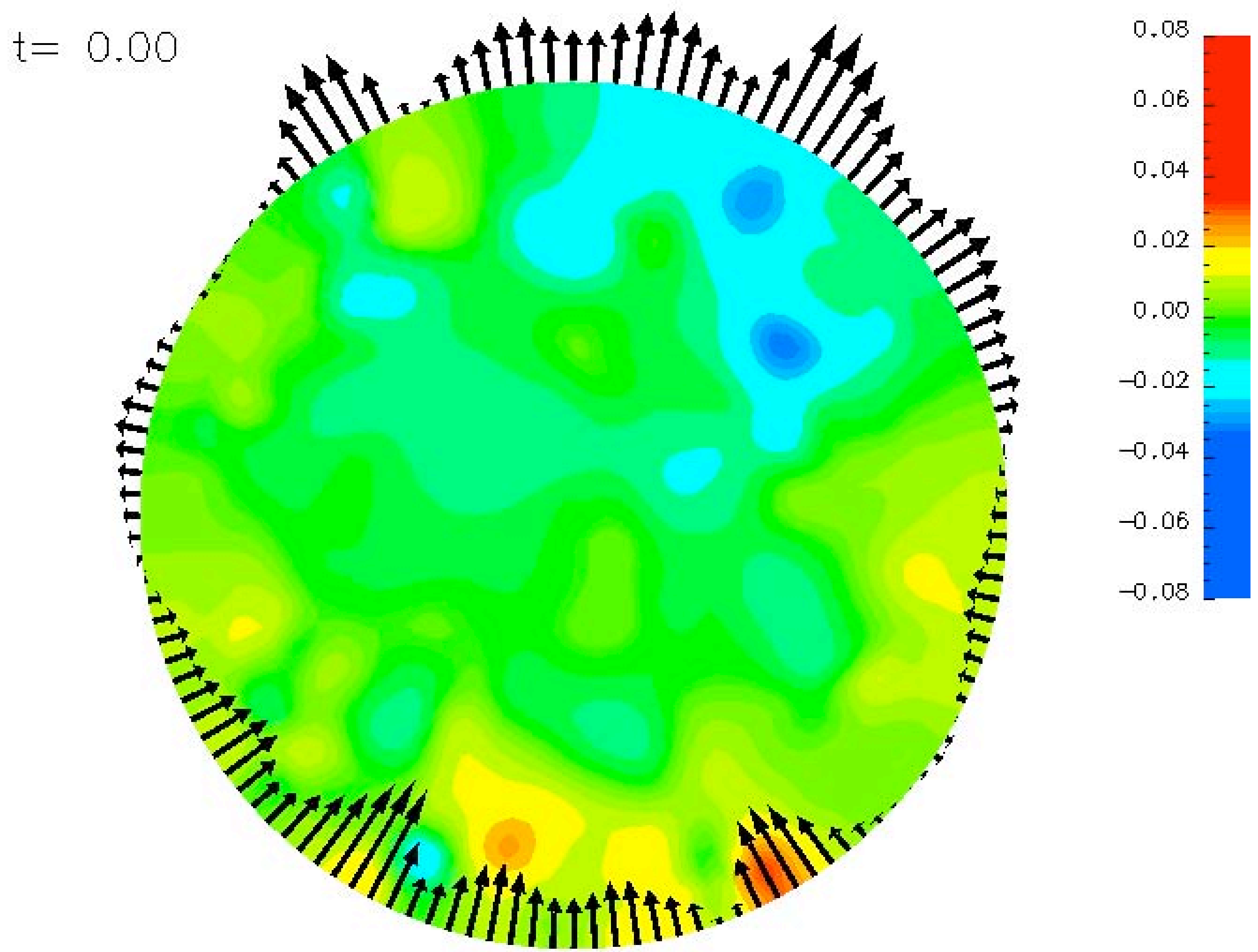
✓ Sensing: pressure difference (centrally symmetric)

Particle Tracking (Re=2100, *L/R*=3π)





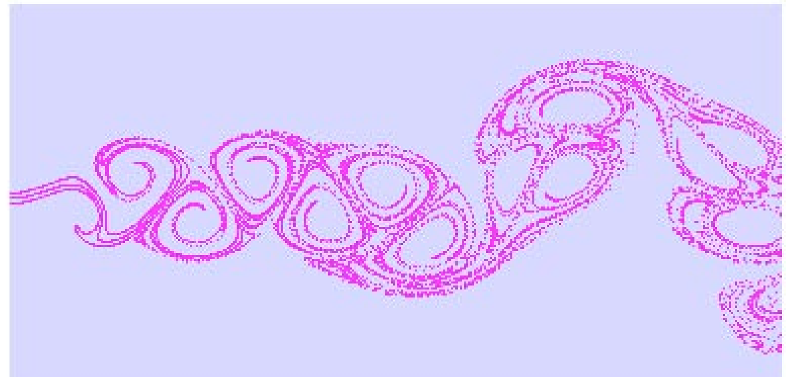
Pressure Field and Controlled Velocity



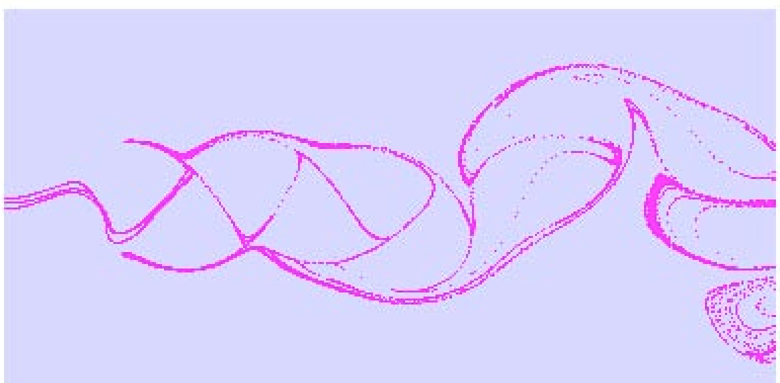
Control of 2D Jet Flow

Uncontrolled

Controlled - light particles



Controlled - heavy particles



Diffusive mixing

UCSD Jacobs | Mechanical and Aerospace Engineering

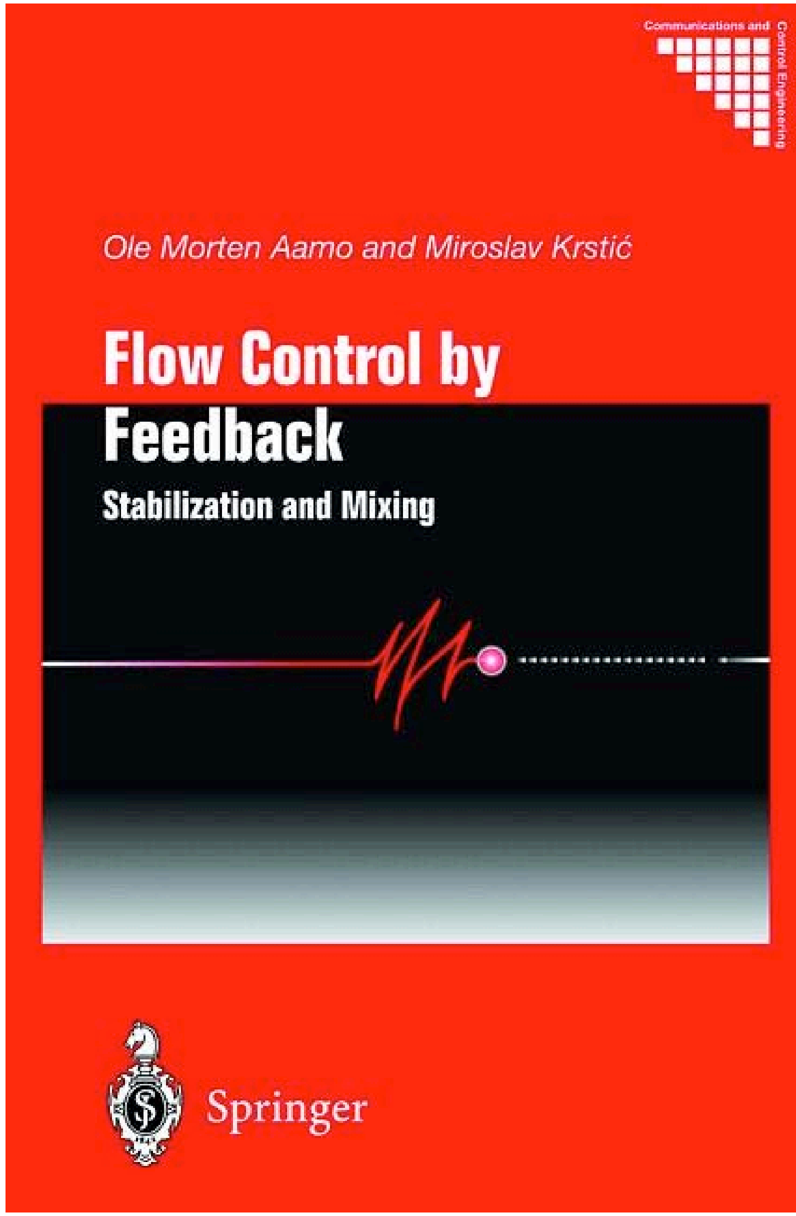
Optimality

Theorem:

The control law maximizes the gain from the (temporal) L_2 norm of the control input to the L_2 norm of *dissipation.*

dissipation = spatial L_2 norm of the gradient of the velocity vector

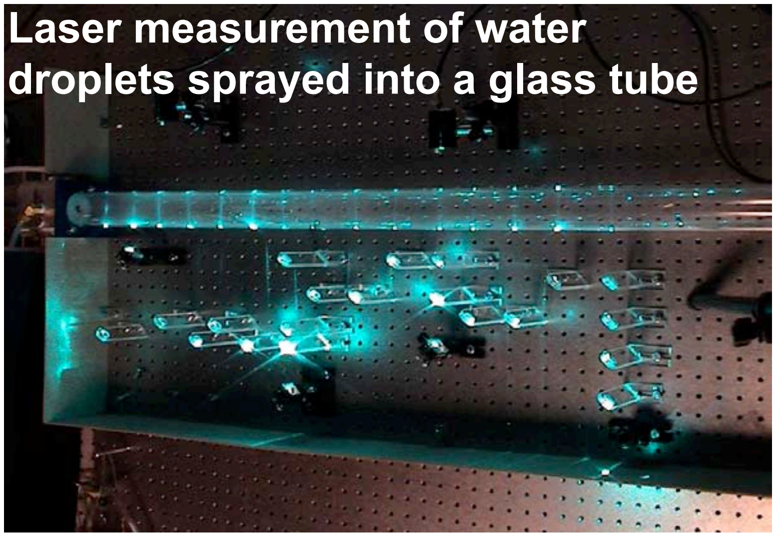
More in…

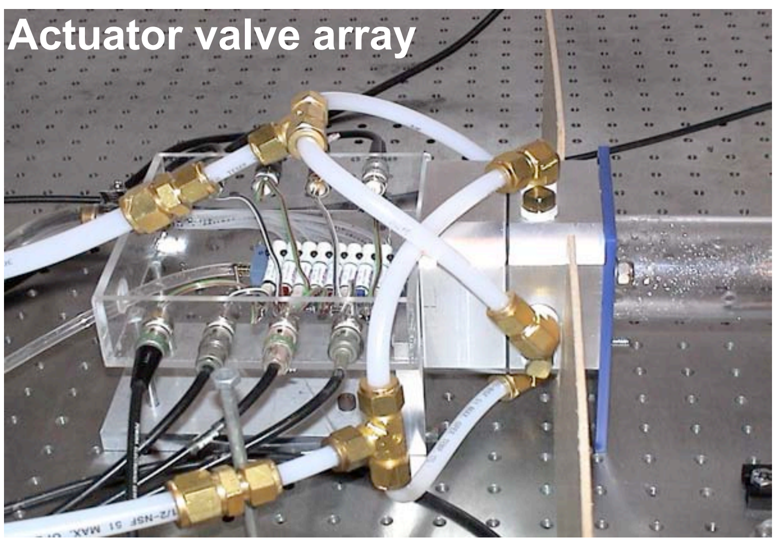


UCSD Jacobs | Mechanical and Aerospace Engineering

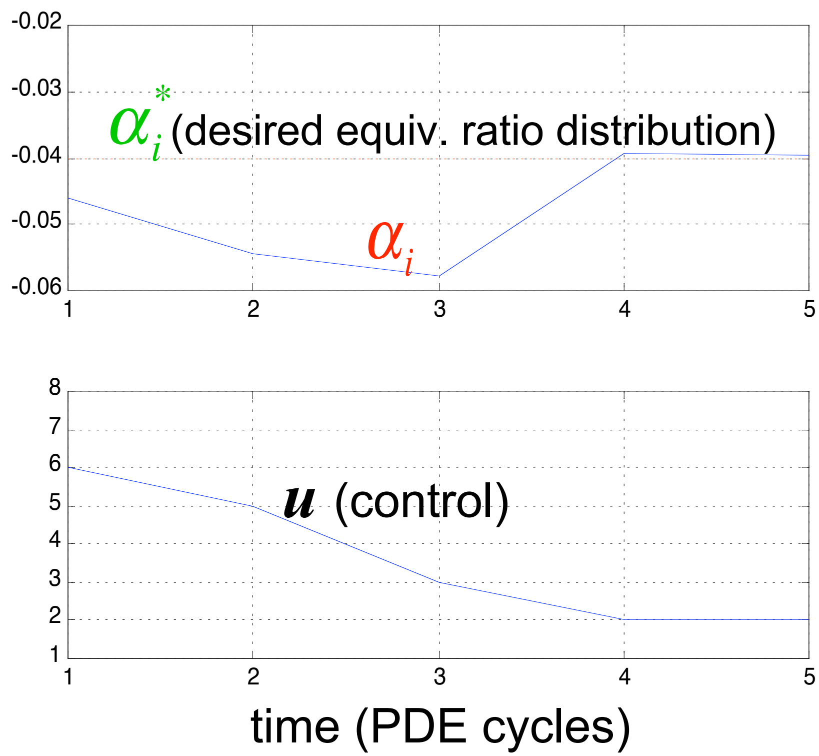
Tailored Fuel Injection for Pulsed Detonation Engines

(Aliseda, Ariyur, Lasheras, Krstic, and Williams)





Multivariable PI controller



UCSD Jacobs | Mechanical and Aerospace Engineering

COMBUSTION INSTABILITY CONTROL
via Extremum Seeking

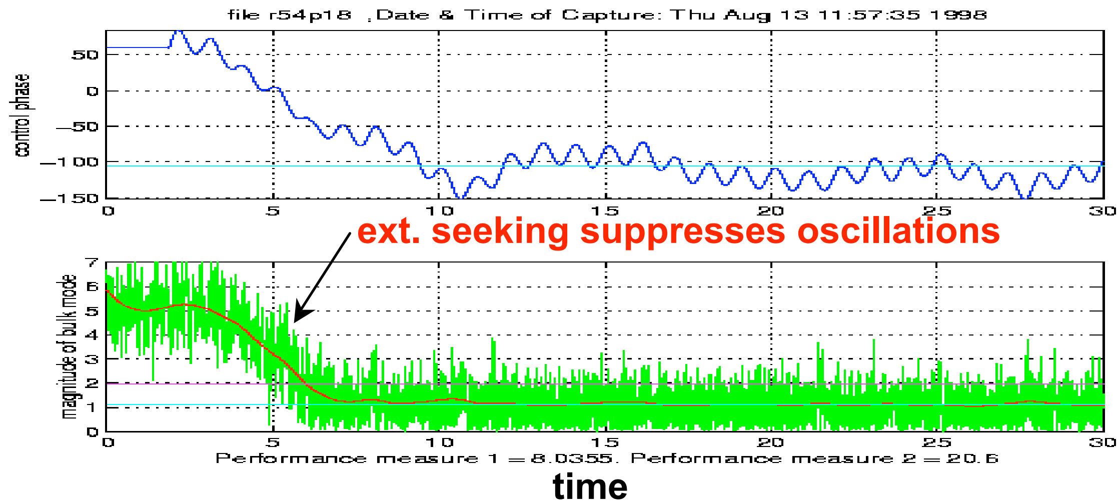
with UTRC

Problem Statement

- Rayleigh criterion-based controllers, which use phase-shifted **pressure measurements** and **fuel modulation**, have emerged as prevalent

- The length of the phase needed varies with operating conditions. The **tuning** method must be non-model based.

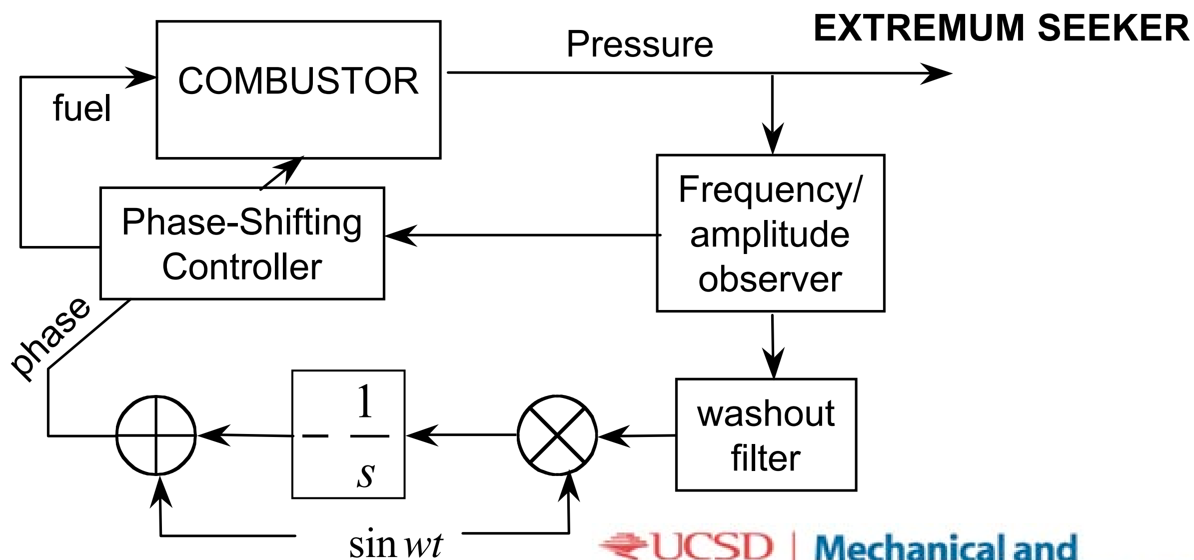
Experimental Results (4MW combustor)



Impact

- Tuning allows operation with **minimum oscillations** at **lean** conditions

- **Reduced** engine size, fuel consumption and NO_x **emissions**



UCSD Jacobs | Mechanical and Aerospace Engineering

AXIAL FLOW COMPRESSOR CONTROL by Extremum Seeking

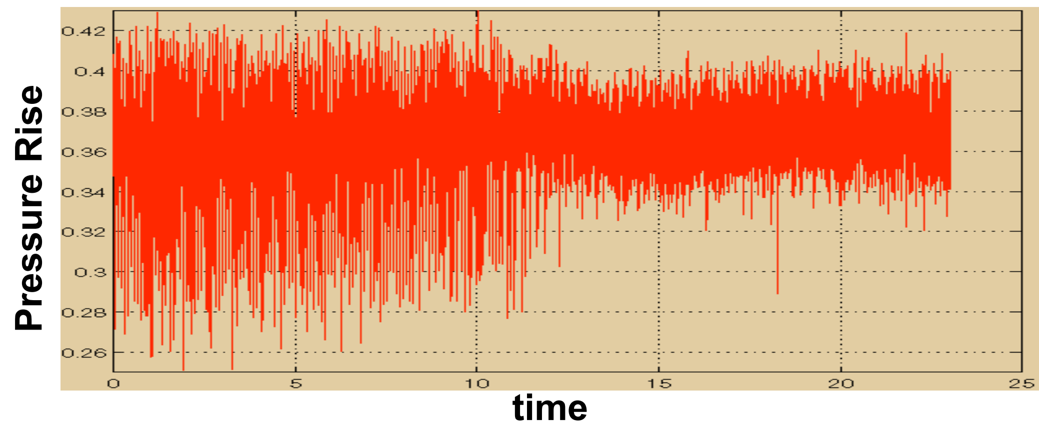
H.-H. Wang

Problem Statement

- Active controls for **rotating stall** only reduce the stall oscillations but they do not bring them to zero nor do they maximize pressure rise.
- Extremum seeking to optimize compressor operating point.

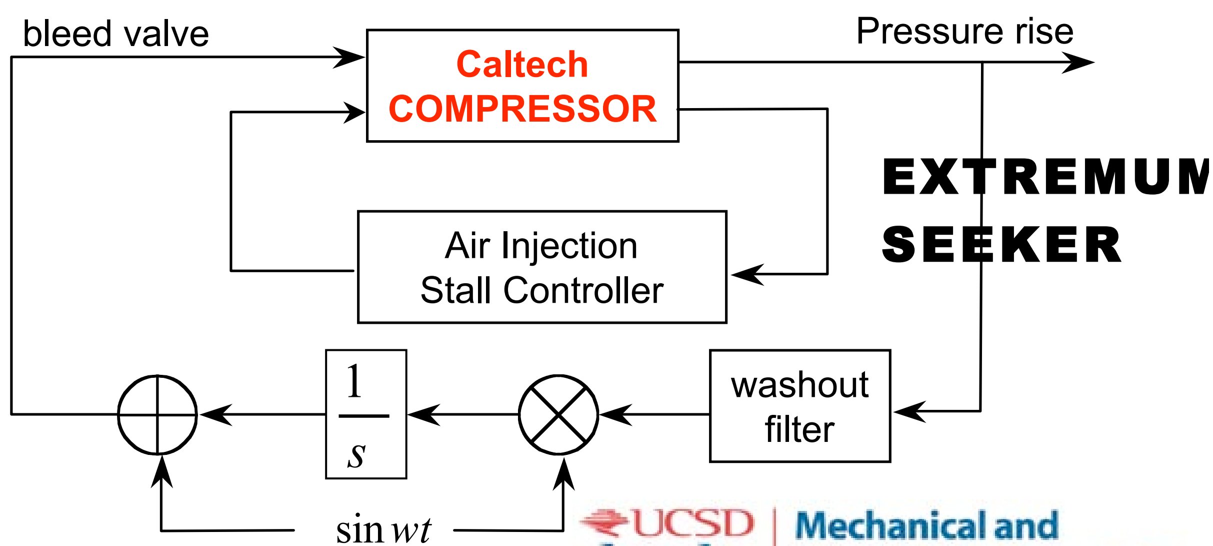
Experimental Results

Extremum seeking stabilizes the **maximum** pressure rise.



Impact

- Smaller, **lighter compressors**; higher payload in aircraft
- Patent issued (August 2000)

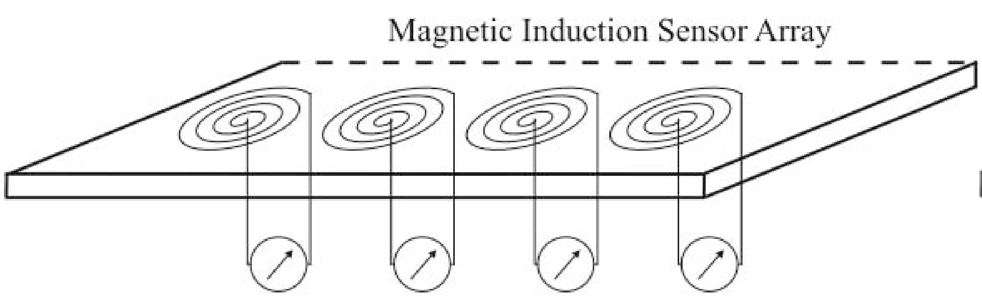


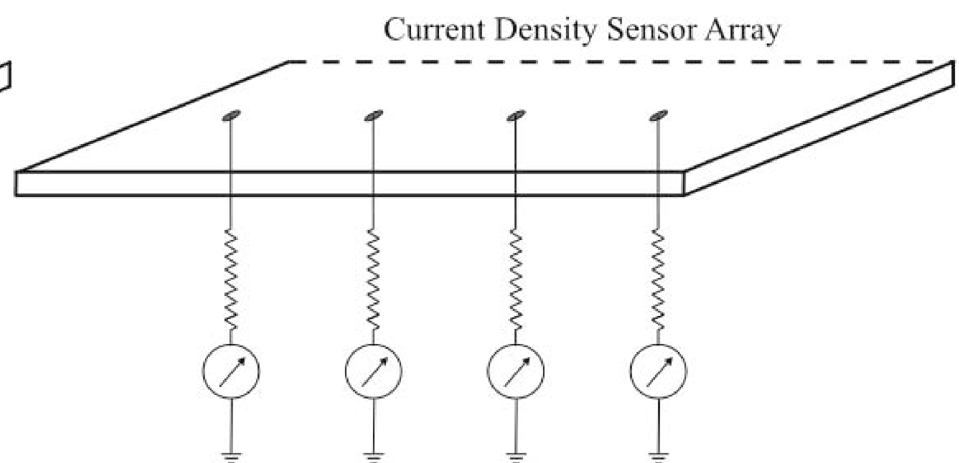
Control of Magnetohydrodynamic Flows

- For drag management in **hypersonic flight** (re-entry vehicles and SCRAMJET propulsion).

- For **liquid metal blankets** in fusion reactors.

- Control possible using purely **electrical** actuators and sensors (rather than MicroElectro**Mechanical**Systems).

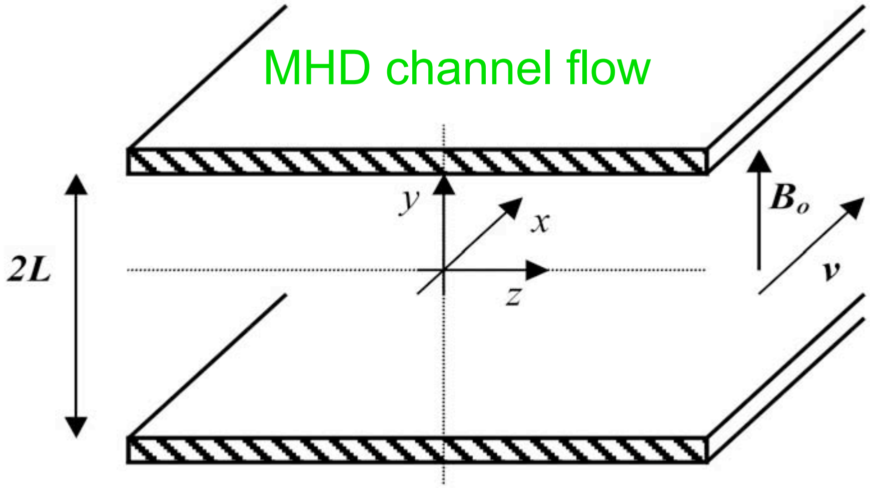


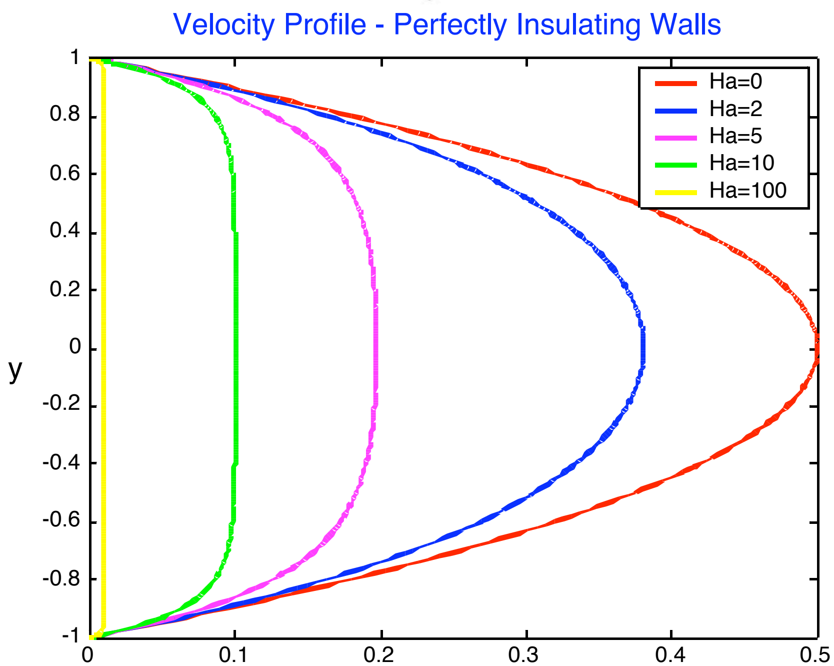


Eugenio Schuster

UCSD Jacobs | Mechanical and Aerospace Engineering

MHD Governing Equations





Navier-Stokes Equation

Magnetic Induction Equation

Incompressibility Condition

Magnetic Field Equation

Faraday's Law

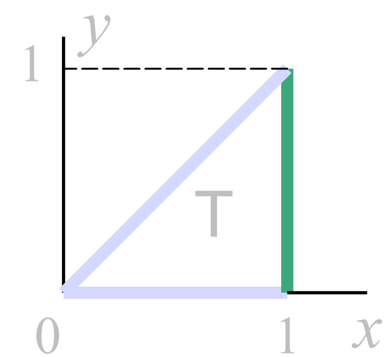
Ampere's Law

Ohm's Law

Appendix: Analysis of Kernel P(I)DE

$$\varepsilon k_{xx}(x,y)-\varepsilon k_{yy}(x,y)=(\lambda(y)+c)k(x,y)-f(x,y)+\int_y^x k(x,\tau)f(\tau,y)d\tau$$

$$(x,y)\in \boldsymbol{T}, \qquad \boldsymbol{T}=\{x,y:0<y<x<1\}$$



"boundary" conditions

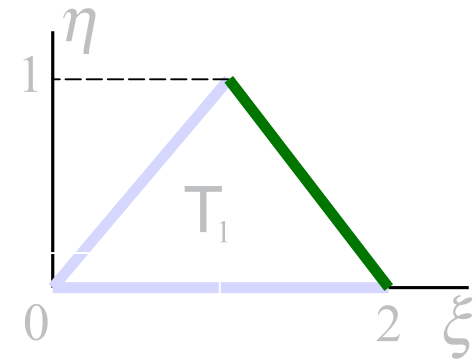
$$k(x,x)=-\frac{1}{2\varepsilon}\int_0^x(\lambda(y)+c)\,dy$$

$$\varepsilon k_y(x,0)=\varepsilon q k(x,0)+g(x)-\int_0^x k(x,y)g(y)\,dy$$

UCSD Jacobs | Mechanical and Aerospace Engineering

Conversion to Integral Equation

$$\xi = x + y \qquad \eta = x - y \qquad G(\xi,\eta) = k\left(\frac{\xi+\eta}{2}, \frac{\xi-\eta}{2}\right)$$



$$\begin{aligned}
G(\xi,\eta) = &-\frac{1}{4\varepsilon}\int_\eta^\xi a\left(\frac{\tau}{2}\right)d\tau - \frac{1}{2\varepsilon}\int_0^\eta e^{q(\tau-\eta)}\left[a\left(\frac{\tau}{2}\right)+2g(\tau)\right]d\tau \\
&-\frac{1}{4\varepsilon}\int_\eta^\xi\int_0^\eta f\left(\frac{\tau+s}{2},\frac{\tau-s}{2}\right)ds\,d\tau - \frac{1}{2\varepsilon}\int_0^\eta e^{q(\tau-\eta)}\int_0^\tau f\left(\frac{\tau+s}{2},\frac{\tau-s}{2}\right)ds\,d\tau \\
&+\frac{1}{4\varepsilon}\int_\eta^\xi\int_0^\eta a\left(\frac{\tau-s}{2}\right)G(\tau,s)\,ds\,d\tau + \frac{1}{4\varepsilon}\int_\eta^\xi e^{q(\tau-\eta)}\int_0^\eta a\left(\frac{\tau-s}{2}\right)G(\tau,s)\,ds\,d\tau \\
&+\frac{1}{4\varepsilon}\int_\eta^\xi\int_0^\eta\int_\mu^{\mu+\eta-s} f\left(\frac{\tau-s}{2},\mu-\frac{\tau+s}{2}\right)G(\tau,s)\,ds\,d\tau \\
&+\frac{1}{2\varepsilon}\int_0^\eta e^{q(\mu-\eta)}\int_0^\mu\int_\mu^{2\mu-s} f\left(\frac{\tau-s}{2},\mu-\frac{\tau+s}{2}\right)G(\tau,s)\,ds\,d\tau \\
&+\frac{1}{2\varepsilon}\int_0^\eta\int_s^{2\eta-s} e^{q((\tau+s)/2-\eta)}g\left(\frac{\tau-s}{2}\right)G(\tau,s)\,ds\,d\tau
\end{aligned}$$

UCSD Jacobs | Mechanical and Aerospace Engineering