Boundary Control of PDEs:

A Course on Backstepping Designs

*class slides*

Scott Moura
Miroslav Krstic
Introduction
Fluid flows in aerodynamics and propulsion applications;

plasmas in lasers, fusion reactors, and hypersonic vehicles;

liquid metals in cooling systems for tokamaks and computers, as well as in welding and metal casting processes;

acoustic waves, water waves in irrigation systems...

Flexible structures in civil engineering, aircraft wings and helicopter rotors, astronomical telescopes, and in nanotechnology devices like the atomic force microscope...

Electromagnetic waves and quantum mechanical systems...

Waves and “ripple” instabilities in thin film manufacturing and in flame dynamics...

Chemical processes in process industries and in internal combustion engines...
Unfortunately, even “toy” PDE control problems like heat and wave equations (neither of which is unstable) require some background in functional analysis.

Courses in control of PDEs rare in engineering programs.

This course: methods which are easy to understand, minimal background beyond calculus.
Boundary Control

Two PDE control settings:

- “in domain” control (actuation penetrates inside the domain of the PDE system or is evenly distributed everywhere in the domain, likewise with sensing);

- “boundary” control (actuation and sensing are only through the boundary conditions).

Boundary control physically more realistic because actuation and sensing are non-intrusive (think, fluid flow where actuation is from the walls).*

*“Body force” actuation of electromagnetic type is also possible but it has low control authority and its spatial distribution typically has a pattern that favors the near-wall region.
Classes of PDEs and Benchmark PDEs Dealt With in the Course

In contrast to ODEs, no general methodology for PDEs.

Two basic categories of PDEs studied in textbooks: *parabolic* and *hyperbolic* PDEs, with standard examples being heat and wave equations.

Many more categories.
Categorization of PDEs studied in the course

<table>
<thead>
<tr>
<th></th>
<th>$\partial_t$</th>
<th>$\partial_{tt}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\partial_x$</td>
<td>transport PDEs, delays</td>
<td></td>
</tr>
</tbody>
</table>
| $\partial_{xx}$ | **parabolic PDEs,**  
reaction-advection-diffusion systems | hyperbolic PDEs,  
wave equations |
| $\partial_{xxx}$ | Korteweg-de Vries                 |                                       |
| $\partial_{xxxx}$ | Kuramoto-Sivashinsky  
and Navier-Stokes  
(Orr-Sommerfeld form) | Euler-Bernoulli  
and shear beams,  
Schroedinger, Ginzburg-Landau |

Timoshenko beam model has four derivatives in both time and space.

Also, complex-valued PDEs (with complex coefficients): Schrodinger and Ginzburg-Landau eqns. They “look” like parabolic PDEs, but behave like oscillatory, hyperbolic PDEs. Schrodinger equivalent to the Euler-Bernoulli beam PDE.
Choices of Boundary Controls

Thermal: actuate heat flux or temperature.

Structural: actuate beam’s boundary position, or force, or angle, or moment.

Mathematical choices of boundary control:

Dirichlet control $u(1,t)$—actuate value of a function at boundary

Neumann control $u_x(1,t)$—actuate slope of a function at boundary
The following are the basic types of boundary conditions for PDEs in dimension one:

- **Dirichlet**: \( w(0) = 0 \) (fixed temperature at \( x = 0 \))
- **Neumann**: \( w_x(0) = 0 \) (fixed heat flux at \( x = 0 \))
- **Robin (mixed)**: \( w_x(0) + qw(0) = 0 \)
Lyapunov Stability
Recall some basics of stability analysis for linear ODEs.

An ODE

\[ \dot{z} = Az, \quad z \in \mathbb{R}^n \]  

is exponentially stable (e.s.) at \( z = 0 \) if \( \exists M > 0 \) (overshoot coeff.) and \( \alpha > 0 \) (decay rate) s.t.

\[ \|z(t)\| \leq Me^{-\alpha t} \|z(0)\|, \quad \text{for all } t \geq 0 \]  

\( \| \cdot \| \) denotes one of the equivalent vector norms, e.g., the 2-norm.
Lyapunov Analysis for a Heat Equation in Terms of ‘$L_2$ Energy’

\[ \frac{w_t}{w_{xx}} \quad (25) \]
\[ w(0) = 0 \quad (26) \]
\[ w(1) = 0. \quad (27) \]

Obviously stable for physical reasons and stability can also be shown by finding explic. soln.

But we want to learn a method for analyzing stability.

Lyapunov function candidate ("energy")

\[ V(t) = \frac{1}{2} \int_0^1 w^2(x, t) \, dx = \frac{1}{2} \| w(t) \|^2 \quad (28) \]

where \( \| \cdot \| \) denotes the \( L_2 \) norm of a function of \( x \):

\[ \| w(t) \| = \left( \int_0^1 w(x, t)^2 \, dx \right)^{1/2}. \]

\[ ^{\text{Strictly speaking, this is a functional, but we refer to it simply as a “Lyapunov function.”}} \]
Time derivative of $V$:

$$\dot{V} = \frac{dV}{dt} = \int_0^1 w(x,t)w_t(x,t)dx \quad \text{(applying the chain rule)}$$

$$= \int_0^1 ww_{xx}dx \quad \text{(from (25))}$$

$$= \left. ww_x \right|_0^1 - \int_0^1 w_x^2dx \quad \text{(integration by parts)}$$

$$= -\int_0^1 w_x^2dx. \quad \text{(29)}$$

Since $\dot{V} \leq 0$, $V$ is bounded. However, it is not clear if $V$ goes to zero because (29) depends on $w_x$ and not on $w$, so one cannot express the right hand side of (29) in terms of $V$. 
Recall two useful inequalities:

**Young's Inequality (special case)**

\[ ab \leq \frac{\gamma}{2} a^2 + \frac{1}{2\gamma} b^2 \]  

(30)

**Cauchy-Schwartz Inequality**

\[ \int_0^1 uw \, dx \leq \left( \int_0^1 u^2 \, dx \right)^{1/2} \left( \int_0^1 w^2 \, dx \right)^{1/2} \]  

(31)
The following lemma establishes the relationship between the $L_2$ norms of $w$ and $w_x$.

**Lemma 1 (Poincare Inequality)** For any $w$, continuously differentiable on $[0, 1]$, 

\[
\begin{align*}
\int_0^1 w^2 \, dx &\leq 2w^2(1) + 4 \int_0^1 w_x^2 \, dx \\
\int_0^1 w^2 \, dx &\leq 2w^2(0) + 4 \int_0^1 w_x^2 \, dx
\end{align*}
\]  

(32)

**Remark 1** The inequalities (32) are conservative. A tighter version of (32) is

\[
\int_0^1 w^2 \, dx \leq w^2(0) + \frac{8}{\pi^2} \int_0^1 w_x^2 \, dx,
\]

which is called “a variation of Wirtinger’s inequality.” The proof of (33) is far more complicated than the proof of (32) and is given in the classical book on inequalities by Hardy, Littlewood, and Polya. When $w(0) = 0$ or $w(1) = 0$, one can even get $\|w\| \leq \frac{2}{\pi} \|w_x\|$.  


Proof.

\[
\int_0^1 w^2 dx = xw^2|_0^1 - 2 \int_0^1 xww_x dx \quad \text{(integration by parts)}
\]
\[
= w^2(1) - 2 \int_0^1 xww_x dx
\]
\[
\leq w^2(1) + \frac{1}{2} \int_0^1 w^2 dx + 2 \int_0^1 x^2 w_x^2 dx.
\]

Subtracting the second term from both sides we get the first inequality in (32):

\[
\frac{1}{2} \int_0^1 w^2 dx \leq w^2(1) + 2 \int_0^1 x^2 w_x^2 dx
\]
\[
\leq w^2(1) + 2 \int_0^1 w_x^2 dx.
\]  

(34)

The second inequality in (32) is obtained in a similar fashion. QED
We now return to
\[ \dot{V} = -\int_0^1 w_x^2 dx. \]

Using Poincare inequality along with boundary conditions \( w(0) = w(1) = 0 \), we get
\[ \dot{V} = -\int_0^1 w_x^2 dx \leq -\frac{1}{4} \int_0^1 w^2 \leq -\frac{1}{2} V \]  \hspace{1cm} (35)

which, by the basic comparison principle for first order differential inequalities, implies that
\[ V(t) \leq V(0) e^{-t/2}, \]  \hspace{1cm} (36)

or
\[ \| w(t) \| \leq e^{-t/4} \| w_0 \| \]  \hspace{1cm} (37)

Thus, the system (25)–(27) is exponentially stable in \( L_2 \).
Response of a heat equation to a non-smooth initial condition.

The “instant smoothing” effect is the characteristic feature of the diffusion operator that dominates the heat equation.
Example 1 Consider the diffusion-advection equation

\[
\begin{align*}
    w_t &= w_{xx} + w_x \quad (49) \\
    w_x(0) &= 0 \quad (50) \\
    w(1) &= 0. \quad (51)
\end{align*}
\]

Using the Lyapunov function (28) we get

\[
\begin{align*}
    \dot{V} &= \int_0^1 w w_t \, dx \\
    &= \int_0^1 w w_{xx} \, dx + \int_0^1 w w_x \, dx \\
    &= w w_x \Big|_0^1 - \int_0^1 w_x^2 \, dx + \int_0^1 w w_x \, dx \\
    &= -\int_0^1 w_x^2 \, dx + \frac{1}{2} w^2 \Big|_0^1 \\
    &= -\int_0^1 w_x^2 \, dx + \frac{1}{2} w^2(1) - \frac{1}{2} w^2(0) \\
    &= -\int_0^1 w_x^2 \, dx - \frac{1}{2} w^2(0).
\end{align*}
\]
Finally, using the Poincare inequality (32) we get

$$\dot{V} \leq -\frac{1}{4}\|w\|^2 \leq -\frac{1}{2}V,$$

proving the exponential stability in $L_2$ norm,

$$\|w(t)\| \leq e^{-t/4}\|w_0\|.$$
Backstepping for Parabolic PDEs

(Reaction-Advection-Diffusion and Other Equations)
The most important part of this course.

We introduce the method of *backstepping*, using the class of parabolic PDEs.

Later we extend backstepping to 1st and 2nd-order hyperbolic PDEs and to other classes.

Parabolic PDEs are first order in time and, while they can have a large number of unstable eigenvalues, this number is finite, which makes them more easily accessible to a reader with background in ODEs.
Backstepping is capable of eliminating destabilizing forces/terms acting in the domain’s interior, using control that acts only on the boundary.

We build a state transformation, which involves a Volterra integral operator that ‘absorbs’ the destabilizing terms acting in the domain and brings them to the boundary, where control can eliminate them.

The Volterra operator has a lower triangular structure.
ODE Backstepping

The *backstepping* method and its name originated in the early 1990’s for stabilization of nonlinear ODE systems

Backstepping for PDEs—the Main Idea

Start with one of the simplest unstable PDEs, the reaction-diffusion equation:

\[ u_t(x,t) = u_{xx}(x,t) + \lambda u(x,t) \quad (104) \]
\[ u(0,t) = 0 \quad (105) \]
\[ u(1,t) = U(t) = \text{control} \quad (106) \]

The open-loop system (104), (105) (with \( u(1,t) = 0 \)) is unstable with arbitrarily many unstable eigenvalues for sufficiently large \( \lambda > 0 \).

Since the term \( \lambda u \) is the source of instability, the natural objective for a boundary feedback is to “eliminate” this term.
State transformation

\[ w(x,t) = u(x,t) - \int_0^x k(x,y)u(y,t) \, dy \]  \hspace{1cm} (107)

Feedback control

\[ u(1,t) = \int_0^1 k(1,y)u(y,t) \, dy \]  \hspace{1cm} (108)

*Target* system (exp. stable)

\[ \begin{align*}
    w_t(x,t) &= w_{xx}(x,t) \hspace{1cm} (109) \\
    w(0,t) &= 0 \hspace{1cm} (110) \\
    w(1,t) &= 0 \hspace{1cm} (111)
\end{align*} \]

Task: find kernel \( k(x,y) \).
The Volterra integral transformation in (107) has the following features:

The limits of integral are from 0 to \( x \), not from 0 to 1.

“Spatially causal,” that is, for a given \( x \) the right hand side of (107) depends only on the values of \( u \) in the interval \([0, x]\).

Invertible because of the presence of the identity operator and the spatial causality of the Volterra operator. Because of invertibility, stability of the target system translates into stability of the closed loop system consisting of the plant plus boundary feedback.
Gain Kernel PDE

Task: find the function $k(x, y)$ (which we call “gain kernel”) that makes the plant (104), (105) with the controller (108) equivalent to the target system (109)–(111).

We introduce the following notation:

$$k_x(x, x) = \frac{\partial}{\partial x} k(x, y) |_{y=x}$$

$$k_y(x, x) = \frac{\partial}{\partial y} k(x, y) |_{y=x}$$

$$\frac{d}{dx} k(x, x) = k_x(x, x) + k_y(x, x).$$
Differentiate the transformation (107) with respect to $x$ and $t$ using Leibnitz’s rule

$$\frac{d}{dx} \int_{0}^{x} f(x, y) \, dy = f(x, x) + \int_{0}^{x} f_x(x, y) \, dy.$$ 

Differentiating the transformation (107) with respect to $x$ gives

$$w_x(x) = u_x(x) - k(x, x)u(x) - \int_{0}^{x} k_x(x, y)u(y) \, dy$$

$$w_{xx}(x) = u_{xx}(x) - \frac{d}{dx}(k(x, x)u(x)) - k_x(x, x)u(x) - \int_{0}^{x} k_{xx}(x, y)u(y) \, dy$$

$$= u_{xx}(x) - u(x)\frac{d}{dx}k(x, x) - k(x, x)u_x(x) - k_x(x, x)u(x)$$

$$- \int_{0}^{x} k_{xx}(x, y)u(y) \, dy. \quad (112)$$
Next, we differentiate the transformation (107) with respect to time:

\[
wt(x) = ut(x) - \int_0^x k(x, y)u_t(y)dy
\]

\[
= u_{xx}(x) + \lambda u(x) - \int_0^x k(x, y) (u_{yy}(y) + \lambda u(y)) dy
\]

\[
= u_{xx}(x) + \lambda u(x) - k(x, x)u_x(x) + k(x, 0)u_x(0)
\]

\[
+ \int_0^x k(y, x)u_y(y)dy - \int_0^x \lambda k(x, y)u(y)dy \quad \text{(integration by parts)}
\]

\[
= u_{xx}(x) + \lambda u(x) - k(x, x)u_x(x) + k(x, 0)u_x(0) + k_y(x, x)u(x) - k_y(x, 0)u(0)
\]

\[
- \int_0^x k_{yy}(y, x)u(y)dy - \int_0^x \lambda k(x, y)u(y)dy. \quad \text{(integration by parts)}
\]  \hspace{1cm} (113)

Subtracting (112) from (113), we get

\[
wt - w_{xx} = \left[ \lambda + 2 \frac{d}{dx}k(x, x) \right] u(x) + k(x, 0)u_x(0)
\]

\[
+ \int_0^x \left( k_{xx}(x, y) - k_{yy}(x, y) - \lambda k(x, y) \right) u(y) dy
\]

\[
= 0
\]
For this to hold for all \( u \), three conditions have to be satisfied:

\[
\begin{align*}
    k_{xx}(x, y) - k_{yy}(x, y) - \lambda k(x, y) &= 0 \quad (114) \\
    k(x, 0) &= 0 \quad (115) \\
    \lambda + 2 \frac{d}{dx}k(x, x) &= 0. \quad (116)
\end{align*}
\]

We simplify (116) by integrating it with respect to \( x \) and noting from (115) that \( k(0, 0) = 0 \), which gives us

\[
\begin{align*}
    k_{xx}(x, y) - k_{yy}(x, y) &= \lambda k(x, y) \\
    k(x, 0) &= 0 \quad (117) \\
    k(x, x) &= -\frac{\lambda}{2} x
\end{align*}
\]
These three conditions form a well posed PDE of hyperbolic type in the “Goursat form.”

One can think of the $k$-PDE as a wave equation with an extra term $\lambda k$.

$x$ plays the role of time and $y$ of space.

In quantum physics such PDEs are called Klein-Gordon PDEs.
Domain of the PDE for gain kernel $k(x, y)$.

The boundary conditions are prescribed on hypotenuse and the lower cathetus of the triangle.

The value of $k(x, y)$ on the vertical cathetus gives us the control gain $k(1, y)$. 
Converting Gain Kernel PDE to an Integral Equation

To find a solution of the $k$-PDE (117) we first convert it into an integral equation.

Introducing the change of variables

$$
\xi = x + y, \quad \eta = x - y
$$

we have

$$
\begin{align*}
k(x, y) &= G(\xi, \eta) \\
k_x &= G_\xi + G_\eta \\
k_{xx} &= G_{\xi\xi} + 2G_{\xi\eta} + G_{\eta\eta} \\
k_y &= G_\xi - G_\eta \\
k_{yy} &= G_{\xi\xi} - 2G_{\xi\eta} + G_{\eta\eta}.
\end{align*}
$$
Thus, the gain kernel PDE becomes

\[
G_{\xi \eta}(\xi, \eta) = \frac{\lambda}{4} G(\xi, \eta) \tag{119}
\]

\[
G(\xi, \xi) = 0 \tag{120}
\]

\[
G(\xi, 0) = -\frac{\lambda}{4} \xi. \tag{121}
\]

Integrating (119) with respect to \( \eta \) from 0 to \( \eta \), we get

\[
G_{\xi}(\xi, \eta) = G_{\xi}(\xi, 0) + \int_{0}^{\eta} \frac{\lambda}{4} G(\xi, s) \, ds = -\frac{\lambda}{4} + \int_{0}^{\eta} \frac{\lambda}{4} G(\xi, s) \, ds. \tag{122}
\]

Next, we integrate (122) with respect to \( \xi \) from \( \eta \) to \( \xi \) to get the integral equation

\[
G(\xi, \eta) = -\frac{\lambda}{4}(\xi - \eta) + \frac{\lambda}{4} \int_{\eta}^{\xi} \int_{0}^{\eta} G(\tau, s) \, ds \, d\tau \tag{123}
\]

The \( G \)-integral eqn is easier to analyze than the \( k \)-PDE.
Method of Successive Approximations

Start with an initial guess

\[ G^0(\xi, \eta) = 0 \]  \hspace{1cm} (124)

and set up the recursive formula for (123) as follows:

\[
G^{n+1}(\xi, \eta) = -\frac{\lambda}{4}(\xi - \eta) + \frac{\lambda}{4} \int_{\eta}^{\xi} \int_{0}^{\eta} G^n(\tau, s) ds d\tau
\]  \hspace{1cm} (125)

If this functional iteration converges, we can write the solution \( G(\xi, \eta) \) as

\[
G(\xi, \eta) = \lim_{n \to \infty} G^n(\xi, \eta).
\]  \hspace{1cm} (126)
Let us denote the difference between two consecutive terms as

\[ \Delta G^n(\xi, \eta) = G^{n+1}(\xi, \eta) - G^n(\xi, \eta). \]  

(127)

Then

\[ \Delta G^{n+1}(\xi, \eta) = \frac{\lambda}{4} \int_{\eta}^{\xi} \int_{0}^{\eta} \Delta G^n(\tau, s) \, ds \, d\tau \]  

(128)

and (126) can be alternatively written as

\[ G(\xi, \eta) = \sum_{n=0}^{\infty} \Delta G^n(\xi, \eta). \]  

(129)

Computing \( \Delta G^n \) from (128) starting with

\[ \Delta G^0 = G^1(\xi, \eta) = -\frac{\lambda}{4}(\xi - \eta), \]  

(130)

we can observe the pattern which leads to the following formula:

\[ \Delta G^n(\xi, \eta) = -\frac{(\xi - \eta)\xi^n\eta^n}{n! (n+1)!} \left( \frac{\lambda}{4} \right)^{n+1} \]  

(131)

This formula can be verified by induction.
The solution to the integral equation is given by

\[ G(\xi, \eta) = - \sum_{n=0}^{\infty} \frac{(\xi - \eta)\xi^n\eta^n}{n!(n+1)!} \left( \frac{\lambda}{4} \right)^{n+1}. \]  

(132)

To compute the series (132), note that a first order modified Bessel function of the first kind can be represented as

\[ I_1(x) = \sum_{n=0}^{\infty} \frac{(x/2)^{2n+1}}{n!(n+1)!}. \]  

(133)
ASIDE: Modified Bessel Functions $I_n$

The function $y(x) = I_n(x)$ is a solution to the following ODE

$$x^2y'' + xy' - (x^2 + n^2)y = 0$$  \hspace{1cm} (134)$$

Series representation

$$I_n(x) = \sum_{m=0}^{\infty} \frac{(x/2)^{n+2m}}{m!(m+n)!}$$  \hspace{1cm} (135)$$

Properties

$$2nI_n(x) = x(I_{n-1}(x) - I_{n+1}(x))$$  \hspace{1cm} (136)$$

$$I_n(-x) = (-1)^nI_n(x)$$  \hspace{1cm} (137)$$
Differentiation

\[ \frac{d}{dx} I_n(x) = \frac{1}{2}(I_{n-1}(x) + I_{n+1}(x)) = \frac{n}{x} I_n(x) + I_{n+1}(x) \tag{138} \]

\[ \frac{d}{dx}(x^n I_n(x)) = x^n I_{n-1}, \quad \frac{d}{dx}(x^{-n} I_n(x)) = x^{-n} I_{n+1} \tag{139} \]

Asymptotic properties

\[ I_n(x) \approx \frac{1}{n!} \left( \frac{x}{2} \right)^n, \quad x \to 0 \tag{140} \]

\[ I_n(x) \approx \frac{e^x}{\sqrt{2\pi x}}, \quad x \to \infty \tag{141} \]
Modified Bessel functions $I_n$. 
Comparing (135) with (132) we obtain

\[ G(\xi, \eta) = -\frac{\lambda}{2} (\xi - \eta) \frac{I_1(\sqrt{\lambda\xi\eta})}{\sqrt{\lambda\xi\eta}} \]  

(142)

or, returning to the original \( x, y \) variables,

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

(143)
As $\lambda$ gets larger, the plant becomes more unstable which requires more control effort.

Low gain near the boundaries is logical: near $x = 0$ the state is small even without control because of the boundary condition $u(0) = 0$; near $x = 1$ the control has the most impact.
Inverse Transformation

We need to establish that stability of the $w$-target system (109)–(111) implies stability of the $u$-closed-loop system (104), (105), (108), by showing that the transformation $u \mapsto w$ is invertible.

Postulate an inverse transformation in the form

$$u(x) = w(x) + \int_0^x l(x, y)w(y) \, dy,$$

where $l(x, y)$ is the transformation kernel.

Given the direct transformation (107) and the inverse transformation (144), the kernels $k(x, y)$ and $l(x, y)$ satisfy

$$l(x, y) = k(x, y) + \int_y^x k(x, \xi)l(\xi, y) \, d\xi$$

(145)
Proof of (145). First recall from calculus the following formula for changing the order of integration:

\[
\int_0^x \int_0^y f(x, y, \xi) d\xi dy = \int_0^x \int_\xi^y f(x, y, \xi) dy d\xi
\] (146)

Substituting (144) into (107), we get

\[
w(x) = w(x) + \int_0^x l(x, y)w(y)dy - \int_0^x k(x, y) \left[ w(y) + \int_0^y l(y, \xi)w(\xi)d\xi \right] dy
\]

\[
= w(x) + \int_0^x l(x, y)w(y)dy - \int_0^x k(x, y)w(y)dy - \int_0^x \int_0^y k(x, y)l(y, \xi)w(\xi)d\xi dy
\]

\[
0 = \int_0^x w(y) \left[ l(x, y) - k(x, y) - \int_y^x k(x, \xi)l(\xi, y) d\xi \right] dy.
\]

Since the last line has to hold for all \( w(y) \), we get the relationship (145).
The formula (145) is general (it does not depend on the plant and the target system) but is not very helpful in actually finding \( l(x, y) \) from \( k(x, y) \).

Instead, we follow the same approach that led us to the kernel PDE for \( k(x, y) \).

Differentiating (144) with respect to time we get

\[
\begin{align*}
  u_t(x) &= w_t(x) + \int_0^x l(x, y) w_t(y) \, dy \\
  &= w_{xx}(x) + l(x, x) w_x(x) - l(x, 0) w_x(0) - l_y(x, x) w(x) \\
  &\quad + \int_0^x l_{yy}(x, y) w(y) \, dy \\
\end{align*}
\]  

(147)

and differentiating twice with respect to \( x \) gives

\[
\begin{align*}
  u_{xx}(x) &= w_{xx}(x) + l_x(x, x) w(x) + w(x) \frac{d}{dx} l(x, x) + l(x, x) w_x(x) \\
  &\quad + \int_0^x l_{xx}(x, y) w(y) \, dy. \\
\end{align*}
\]  

(148)
Subtracting (148) from (147) we get

\[
\lambda w(x) + \lambda \int_0^x l(x, y) w(y) \, dy = -2w(x) \frac{d}{dx} l(x, x) - l(x, 0) w_x(0) \\
+ \int_0^x (l_{yy}(x, y) - l_{xx}(x, y)) w(y) \, dy
\]

which gives the following conditions on \( l(x, y) \):

\[
\begin{align*}
  l_{xx}(x, y) - l_{yy}(x, y) &= -\lambda l(x, y) \\
  l(x, 0) &= 0 \\
  l(x, x) &= -\frac{\lambda}{2} x
\end{align*}
\]

(149)

Comparing this PDE with the PDE (117) for \( k(x, y) \), we see that

\[
l(x, y; \lambda) = -k(x, y; -\lambda).
\]

(150)
From (143) we have

\[
l(x, y) = -\lambda y \frac{I_1 \left( \sqrt{-\lambda(x^2 - y^2)} \right)}{\sqrt{-\lambda(x^2 - y^2)}} = -\lambda y \frac{I_1 \left( j\sqrt{\lambda(x^2 - y^2)} \right)}{j\sqrt{\lambda(x^2 - y^2)}},
\]

or, using the properties of \( I_1 \),

\[
l(x, y) = -\lambda y \frac{J_1 \left( \sqrt{\lambda(x^2 - y^2)} \right)}{\sqrt{\lambda(x^2 - y^2)}}
\]

(151)
ASIDE: Bessel Functions $J_n$

The function $y(x) = J_n(x)$ is a solution to the following ODE

$$x^2 y'' + xy' + (x^2 - n^2)y = 0$$  \hspace{1cm} (152)

Series representation

$$J_n(x) = \sum_{m=0}^{\infty} \frac{(-1)^m (x/2)^{n+2m}}{m! (m+n)!}$$  \hspace{1cm} (153)

Relationship with $J_n(x)$

$$I_n(x) = i^{-n}J_n(ix), \quad I_n(ix) = i^n J_n(x)$$  \hspace{1cm} (154)

Properties

$$2nJ_n(x) = x(J_{n-1}(x) + J_{n+1}(x))$$  \hspace{1cm} (155)

$$J_n(-x) = (-1)^n J_n(x)$$  \hspace{1cm} (156)
Differentiation

\[
\frac{d}{dx} J_n(x) = \frac{1}{2} (J_{n-1}(x) - J_{n+1}(x)) = \frac{n}{x} J_n(x) - J_{n+1}(x)
\]

(157)

\[
\frac{d}{dx} (x^n J_n(x)) = x^n J_{n-1}, \quad \frac{d}{dx} (x^{-n} J_n(x)) = -x^{-n} J_{n+1}
\]

(158)

Asymptotic properties

\[
J_n(x) \approx \frac{1}{n!} \left(\frac{x}{2}\right)^n, \quad x \to 0
\]

(159)

\[
J_n(x) \approx \sqrt{\frac{2}{\pi x}} \cos \left(x - \frac{\pi n}{2} - \frac{\pi}{4}\right), \quad x \to \infty
\]

(160)
Bessel functions $J_n$. 
Summary of control design for the reaction-diffusion equation

**Plant**
\[ u_t = u_{xx} + \lambda u \quad (161) \]
\[ u(0) = 0 \quad (162) \]

**Controller**
\[ u(1) = - \int_0^1 y \lambda \frac{I_1 \left( \sqrt{\lambda(1 - y^2)} \right)}{\sqrt{\lambda(1 - y^2)}} u(y) \, dy \quad (163) \]

**Transformation**
\[ w(x) = u(x) + \int_0^x \lambda y \frac{I_1 \left( \sqrt{\lambda(x^2 - y^2)} \right)}{\sqrt{\lambda(x^2 - y^2)}} u(y) \, dy \quad (164) \]
\[ u(x) = w(x) - \int_0^x \lambda y \frac{J_1 \left( \sqrt{\lambda(x^2 - y^2)} \right)}{\sqrt{\lambda(x^2 - y^2)}} w(y) \, dy \quad (165) \]

**Target system**
\[ w_t = w_{xx} \quad (166) \]
\[ w(0) = 0 \quad (167) \]
\[ w(1) = 0 \quad (168) \]
Open-loop response for reaction-diffusion plant (161), (162) for the case $\lambda = 20$. The plant has one unstable eigenvalue $20 - \pi^2 \approx 10$. 
Closed-loop response with controller (163) implemented.
The control (163) for reaction-diffusion plant (161)–(162).
Example 3  Consider the plant with a Neumann boundary cond. on the uncontrolled end,

\[
\begin{align*}
    u_t &= u_{xx} + \lambda u \quad (169) \\
    u_x(0) &= 0 \quad (170) \\
    u(1) &= U(t) \quad (171)
\end{align*}
\]

We use the transformation

\[
    w(x) = u(x) - \int_0^x k(x, y) u(y) \, dy \quad (172)
\]

to map this plant into the target system

\[
\begin{align*}
    w_t &= w_{xx} \quad (173) \\
    w_x(0) &= 0 \quad (174) \\
    w(1) &= 0 \quad (175)
\end{align*}
\]
Differentiation of the transformation (172) with respect to $x$ gives (112) (it does not depend on the particular plant). Differentiating (172) with respect to time, we get

$$w_t(x) = u_t(x) - \int_0^x k(x, y)u_t(y) \, dy$$

$$= u_{xx}(x) + \lambda u(x) - \int_0^x k(x, y)[u_{yy}(y) + \lambda u(y)] \, dy$$

$$= u_{xx}(x) + \lambda u(x) - k(x, x)u_x(x) + k(x, 0)u_x(0)$$

$$+ \int_0^x k_y(x, y)u_y(y) \, dy - \int_0^x \lambda k(x, y)u(y) \, dy \quad \text{(integration by parts)}$$

$$= u_{xx}(x) + \lambda u(x) - k(x, x)u_x(x) + k_y(x, x)u(x) - k_y(x, 0)u(0)$$

$$- \int_0^x k_{yy}(x, y)u(y) \, dy - \int_0^x \lambda k(x, y)u(y) \, dy \quad \text{(integration by parts)}$$

(176)
Subtracting (112) from (176), we get

\[ w_t - w_{xx} = \left[ \lambda + 2 \frac{d}{dx} k(x, x) \right] u(x) - k_y(x, 0) u(0) \]

\[ + \int_0^x \left( k_{xx}(x, y) - k_{yy}(x, y) - \lambda k(x, y) \right) u(y) dy. \]  \hspace{1cm} (177)

For the right hand side of this equation to be zero for all \( u(x) \), three conditions must be satisfied:

\[ k_{xx}(x, y) - k_{yy}(x, y) - \lambda k(x, y) = 0 \]  \hspace{1cm} (178)

\[ k_y(x, 0) = 0 \]  \hspace{1cm} (179)

\[ \lambda + 2 \frac{d}{dx} k(x, x) = 0. \]  \hspace{1cm} (180)

Integrating (180) with respect to \( x \) gives \( k(x, x) = -\lambda / 2x + k(0, 0) \), where \( k(0, 0) \) is obtained using the boundary condition (174):

\[ w_x(0) = u_x(0) + k(0, 0) u(0) = 0, \]

so that \( k(0, 0) = 0 \). The gain kernel PDE is thus

\[ k_{xx}(x, y) - k_{yy}(x, y) = \lambda k(x, y) \]  \hspace{1cm} (181)

\[ k_y(x, 0) = 0 \]  \hspace{1cm} (182)

\[ k(x, x) = -\frac{\lambda}{2} x. \]  \hspace{1cm} (183)
Note that this PDE is very similar to (117). The only difference is in the boundary condition at \( y = 0 \). The solution to the PDE (181)–(183) is obtained through a summation of successive approximation series, similarly to the way it was obtained for the PDE (117):

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

(184)

Thus, the controller is given by

\[
u(1) = -\int_0^1 \frac{I_1 \left( \sqrt{\lambda(1 - y^2)} \right)}{\sqrt{\lambda(1 - y^2)}} u(y) dy.
\]  

(185)
Neumann Actuation

Consider the plant (104), (105) but with the heat flux $u_x(1)$ actuated:

$$
\begin{align*}
  u_t &= u_{xx} + \lambda u \\
  u(0) &= 0 \\
  u_x(1) &= U(t).
\end{align*}
$$

(186) (187) (188)

We use the same transformation (107), (143) as we used in the case of Dirichlet actuation. To obtain the control $u_x(1)$, we need to differentiate (107) with respect to $x$:

$$
w_x(x) = u_x(x) - k(x,x)u(x) - \int_{0}^{x} k_x(x,y)u(y) \, dy
$$

and set $x = 1$. It is clear now that the target system has to have the Neumann boundary condition at $x = 1$:

$$
\begin{align*}
  w_t &= w_{xx} \\
  w(0) &= 0 \\
  w_x(1) &= 0,
\end{align*}
$$

(189) (190) (191)

which gives the controller

$$
u_x(1) = k(1,1)u(1) + \int_{0}^{1} k_x(1,y)u(y) \, dy.
$$

(192)
All that remains is to derive the expression for $k_x$ from (143) using the properties of Bessel functions:

$$k_x(x, y) = -\lambda_y x \frac{I_2\left(\sqrt{\lambda(x^2 - y^2)}\right)}{x^2 - y^2}.$$  

Finally, the controller is

$$u_x(1) = -\frac{\lambda}{2} u(1) - \int_0^1 \lambda y \frac{I_2\left(\sqrt{\lambda(1 - y^2)}\right)}{1 - y^2} u(y) \, dy. \quad (193)$$
Reaction-Advection-Diffusion Equation

\[ u_t = \varepsilon u_{xx} + bu_x + \lambda u \]  
(194)

\[ u(0) = 0 \]  
(195)

\[ u(1) = U(t) \]  
(196)

First, we eliminate the advection term \( u_x \) with the following change of variable:

\[ v(x) = u(x)e^{\frac{b^2}{2\varepsilon}x} \]  
(197)

Taking the temporal and spatial derivatives, we get

\[ u_t(x) = v_t(x)e^{-\frac{b^2}{2\varepsilon}x} \]

\[ u_x(x) = v_x(x)e^{-\frac{b^2}{2\varepsilon}x} - \frac{b}{2\varepsilon}v(x)e^{-\frac{b^2}{2\varepsilon}x} \]

\[ u_{xx}(x) = v_{xx}(x)e^{-\frac{b^2}{2\varepsilon}x} - \frac{b}{2\varepsilon}v_x(x)e^{-\frac{b^2}{2\varepsilon}x} + \frac{b^2}{4\varepsilon^2}v(x)e^{-\frac{b^2}{2\varepsilon}x} \]
In the \( \nu \)-variable we get a reaction-diffusion system

\[
\begin{align*}
\nu_t &= \epsilon \nu_{xx} + \left( \lambda - \frac{b^2}{4\epsilon} \right) \nu \\
\nu(0) &= 0 \\
\nu(1) &= u(1)e^{\frac{b}{2\epsilon}} = \text{control.}
\end{align*}
\] (198, 199, 200)

Now the transformation

\[
w(x) = \nu(x) - \int_0^x k(x, y) \nu(y) dy
\] (201)

leads to the target system

\[
\begin{align*}
w_t &= \epsilon w_{xx} - cw \\
w(0) &= 0 \\
w(1) &= 0.
\end{align*}
\] (202, 203, 204)
Here the constant $c$ is a design parameter that sets the decay rate of the closed loop system. It should satisfy the following stability condition:

$$c \geq \max \left\{ \frac{b^2}{4\epsilon} - \lambda, 0 \right\}.$$  

The max is used to prevent spending unnecessary control effort when the plant is stable.
The gain kernel \( k(x, y) \) can be shown to satisfy the following PDE:

\[
\varepsilon k_{xx}(x, y) - \varepsilon k_{yy}(x, y) = \left( \lambda - \frac{b^2}{4\varepsilon} + c \right) k(x, y)
\]  

(205)

\[
k(x, 0) = 0
\]  

(206)

\[
k(x, x) = -\frac{x}{2\varepsilon} \left( \lambda - \frac{b^2}{4\varepsilon} + c \right).
\]  

(207)

This equation is exactly the same as (117), just with a different constant instead of \( \lambda \),

\[
\lambda_0 = \frac{1}{\varepsilon} \left( \lambda - \frac{b^2}{4\varepsilon} + c \right).
\]  

(208)

Therefore the solution to (205)–(207) is given by

\[
k(x, y) = -\lambda_0 y \frac{I_1 \left( \sqrt{\lambda_0 (x^2 - y^2)} \right)}{\sqrt{\lambda_0 (x^2 - y^2)}}.
\]  

(209)
The controller is

\[ u(1) = \int_0^1 e^{-\frac{b}{2\varepsilon}(1-y)\lambda_0 y} \frac{I_1 \left( \sqrt{\lambda_0 (1-y^2)} \right)}{\sqrt{\lambda_0 (1-y^2)}} u(y) dy. \]  

(210)

Let us examine the effect of the advection term \( bu_x \) in (194) on open-loop stability and on the size of the control gain. From (198) we see that the advection term has a beneficial effect on open-loop stability, irrespective of the sign of the advection coefficient \( b \). However, the effect of \( b \) on the gain function in the control law in (210) is ‘sign-sensitive.’ Negative values of \( b \) demand much higher control effort than positive values of \( b \). Interestingly, negative values of \( b \) refer to the situation where the state disturbances advect towards the actuator at \( x = 1 \), whereas the ‘easier’ case of positive \( b \) refers to the case where the state disturbances advect away from the actuator at \( x = 1 \) and towards the Dirichlet boundary condition (195) at \( x = 0 \).
Observer Design
Sensors placed at the boundaries.

Motivation: fluid flows (aerodynamics, acoustics, chemical process control, etc.).
Observer Design for PDEs with Boundary Sensing

\[ u_t = u_{xx} + \lambda u \]  \hspace{1cm} (249)

\[ u_x(0) = 0 \]  \hspace{1cm} (250)

\[ u(1) = U(t) \hspace{0.5cm} \text{(open-loop or feedback signal)} \]  \hspace{1cm} (251)

\[ \text{meas. output} = u(0) \hspace{0.5cm} \text{(at the boundary w/ Neumann b.c.)} \]  \hspace{1cm} (252)

Observer:

\[ \hat{u}_t = \hat{u}_{xx} + \lambda \hat{u} + p_1(x)[u(0) - \hat{u}(0)] \]  \hspace{1cm} (253)

\[ \hat{u}_x(0) = p_{10}[u(0) - \hat{u}(0)] \]  \hspace{1cm} (254)

\[ \hat{u}(1) = U(t) \]  \hspace{1cm} (255)

The function \( p_1(x) \) and the constant \( p_{10} \) are observer gains to be determined.
Mimics the finite-dimensional observer format of “copy of the plant plus output injection.”

Finite-dim plant

\[
\begin{align*}
\dot{x} &= Ax + Bu \\
y &= Cx
\end{align*}
\]

Observer

\[
\dot{\hat{x}} = A\hat{x} + Bu + L(y - C\hat{x})
\]

$L = \text{observer gain}$

$L(y - C\hat{x}) = \text{“output error injection”}$

In (253), (254) the obs. gains $p_1(x)$ and $p_{10}$ form an inf-dim “vector” like $L$. 
Objective: find $p_1(x)$ and $p_{10}$ such that $\hat{u}$ converges to $u$.

Error variable

$$\tilde{u} = u - \hat{u}$$ \hfill (259)

Error system

$$\tilde{u}_t = \tilde{u}_{xx} + \lambda \tilde{u} - p_1(x)\tilde{u}(0)$$ \hfill (260)

$$\tilde{u}_x(0) = -p_{10}\tilde{u}(0)$$ \hfill (261)

$$\tilde{u}(1) = 0$$ \hfill (262)

Magic needed: remove the destabilizing term $\lambda \tilde{u}(x)$ using feedback of boundary term $\tilde{u}(0)$
Backstepping transformation

\[ \tilde{u}(x) = \tilde{w}(x) - \int_{0}^{x} p(x, y) \tilde{w}(y) dy \]  

(263)

Target system

\[ \tilde{w}_t = \tilde{w}_{xx} \]  

(264)

\[ \tilde{w}_x(0) = 0 \]  

(265)

\[ \tilde{w}(1) = 0 \]  

(266)
Differentiating the transformation (303), we get

\[ \tilde{u}_t(x) = \tilde{w}_t(x) - \int_0^x p(x, y)\tilde{w}_{yy}(y) \, dy \]
\[ = \tilde{w}_t(x) - p(x, x)\tilde{w}_x(x) + p(x, 0)\tilde{w}_x(0) + p_y(x, x)\tilde{w}(x) \]
\[ - p_y(x, 0)\tilde{w}(0) - \int_0^x p_{yy}(x, y)\tilde{w}(y) \, dy, \]

(267)

\[ \tilde{u}_{xx}(x) = \tilde{w}_{xx}(x) - \tilde{w}(x)\frac{d}{dx}p(x, x) - p(x, x)\tilde{w}_x(x) \]
\[ - p_x(x, x)\tilde{w}(x) - \int_0^x p_{xx}(x, y)\tilde{w}(y) \, dy. \]

(268)
Subtracting (268) from (267), we obtain:

\[
\tilde{u}_t - \tilde{u}_{xx} = \lambda \left( \tilde{w}(x) - \int_0^x p(x,y) \tilde{w}(y) \, dy \right) - p_1(x) \tilde{w}(0) - \tilde{u}(0)
\]

\[
= 2\tilde{w}(x) \frac{d}{dx} p(x,x) - p_y(x,0) \tilde{w}(0) + \int_0^x (p_{xx}(x,y) - p_{yy}(x,y)) \tilde{w}(y) \, dy
\]

want this to = 0

For the last equality to hold, three conditions must be satisfied:

\[
p_{xx}(x,y) - p_{yy}(x,y) = -\lambda p(x,y)
\]

\[
\frac{d}{dx} p(x,x) = \frac{\lambda}{2}
\]

\[
p_1(x) = \epsilon p_y(x,0)
\]
Recall the backstepping transform

\[ \tilde{u}(x) = \tilde{w}(x) - \int_0^x p(x, y) \tilde{w}(y) dy \]  
(273)

\[ \tilde{u}_x(x) = \tilde{w}_x(x) - p(x, x) \tilde{w}(x) - \int_0^x p_x(x, y) \tilde{w}(y) dy \]  
(274)

and set \( x = 1 \) and \( x = 0 \):

\[ \tilde{u}(0) = \tilde{w}(0) \]  
(275)

\[ \tilde{u}(1) = \tilde{w}(1) - \int_0^1 p(1, y) \tilde{w}(y) dy \]  
(276)

\[ \tilde{u}_x(0) = \tilde{w}_x(0) - p(0, 0) \tilde{w}(0) \]  
(277)

Recall that the target system requires that

\[ \tilde{w}_x(0) = 0 \]  
(278)

\[ \tilde{w}(1) = 0 \]  
(279)
It follows that

\[ \tilde{u}(1) = - \int_0^1 p(1,y)\tilde{w}(y)dy \]  
\[ \tilde{u}_x(0) = -p(0,0)\tilde{u}(0) \]  
(280)  
(281)

Recall now the boundary conditions (261), (262)

\[ \tilde{u}_x(0) = -p_{10}\tilde{u}(0) \]  
\[ \tilde{u}(1) = 0 \]  
(282)  
(283)

This provides the conditions:

\[ p_{10} = p(0,0) \]  
\[ p(1,y) = 0 \]  
(284)  
(285)
Let us solve (271) and (285) for $p(x, x)$ and combine the result with the equations (270) and (285):

\[
\begin{bmatrix}
  p_{xx}(x, y) - p_{yy}(x, y) &= -\lambda p(x, y) \\
  p(1, y) &= 0 \\
  p(x, x) &= \frac{\lambda}{2}(x - 1)
\end{bmatrix}
\] (286)

To solve, make a change of variables

\[
\tilde{x} = 1 - y, \quad \tilde{y} = 1 - x, \quad \tilde{p}(\tilde{x}, \tilde{y}) = p(x, y)
\] (287)

which gives the following PDE:

\[
\tilde{p}_{\tilde{x}\tilde{x}}(\tilde{x}, \tilde{y}) - \tilde{p}_{\tilde{y}\tilde{y}}(\tilde{x}, \tilde{y}) = \lambda \tilde{p}(\tilde{x}, \tilde{y})
\] (288)

\[
\tilde{p}(\tilde{x}, 0) = 0,
\] (289)

\[
\tilde{p}(\tilde{x}, \tilde{x}) = -\frac{\lambda}{2}x.
\] (290)
The solution is

$$\bar{p}(\bar{x}, \bar{y}) = -\lambda \bar{y} \frac{I_1(\sqrt{\lambda (\bar{x}^2 - \bar{y}^2)})}{\sqrt{\lambda (\bar{x}^2 - \bar{y}^2)}}.$$

(291)

or, in the original variables,

$$p(x, y) = -\lambda (1-x) \frac{I_1(\sqrt{\lambda (2-x-y)(x-y)})}{\sqrt{\lambda (2-x-y)(x-y)}}.$$

(292)

The observer gains, obtained using (272) and (284) are

$$p_1(x) = p_y(x, 0) = \frac{\lambda (1-x)}{x(2-x)} I_2 \left( \sqrt{\lambda x(2-x)} \right)$$

(293)

$$p_{10} = p(0, 0) = -\frac{\lambda}{2}.$$

(294)
### Summary of the plant and observer

**Plant**

\[ u_t = u_{xx} + \lambda u \]  
\[ u_x(0) = 0 \]  
\[ u(1) = U \]  

**Observer**

\[ \hat{u}_t = \hat{u}_{xx} + \lambda \hat{u} + \frac{\lambda (1-x)}{x(2-x)} I_2 \left( \sqrt{\lambda x(2-x)} \right) \left[ u(0) - \hat{u}(0) \right] \]  
\[ \hat{u}_x(0) = -\frac{\lambda}{2} \left[ u(0) - \hat{u}(0) \right] \]  
\[ \hat{u}(1) = U \]