PDE Backstepping: The First 25 Years

Miroslav Krstic

• CSS Control History Forum 2024 •

Purpose of lecture

- Not a tutorial
- Not a survey

Purpose of lecture

- Not a tutorial
- Not a survey

- **Historical** progression (birth of new ideas)

Purpose of lecture

- Not a tutorial
- Not a survey

- **Historical** progression (birth of new ideas)
- **Chronological**, with small deviations for topical flow

Purpose of lecture

- Not a tutorial
- Not a survey

- **Historical** progression (birth of new ideas)
- **Chronological**, with small deviations for topical flow

- Not comprehensive. Mention papers that initiate topics; books that integrate.

Purpose of lecture

- Not a tutorial
- Not a survey

- **Historical** progression (birth of new ideas)
- **Chronological**, with small deviations for topical flow

- Not comprehensive. Mention papers that initiate topics; books that integrate.
- Historical account $\longrightarrow$ attribution is important

Purpose of lecture

- Not a tutorial
- Not a survey

- **Historical** progression (birth of new ideas)
- **Chronological**, with small deviations for topical flow

- Not comprehensive. Mention papers that initiate topics; books that integrate.
- Historical account $\longrightarrow$ attribution is important
- Community expansion only over the last ten years

Feedback design for PDEs in the first four decades: **1960s-1990's**

- LQ/optimal (operator Riccati eqns)
- pole placement for reduced models of PDEs

Feedback design for PDEs in the first four decades: **1960s-1990's**

- LQ/optimal (operator Riccati eqns)
- pole placement for reduced models of PDEs

Bensoussan, Da Prato, Delfour, Mitter, *Representation and Control of Inf-Dim Sys*, 1993.

Curtain, Zwart, *Introduction to Infinite-Dimensional Linear Systems Theory*, 1995.

Lasiecka, Triggiani, *Control Theory for PDEs*, 2000.

Feedback design for PDEs in the first four decades: **1960s-1990's**

- LQ/optimal (operator Riccati eqns)
- pole placement for reduced models of PDEs

Bensoussan, Da Prato, Delfour, Mitter, *Representation and Control of Inf-Dim Sys*, 1993.

Curtain, Zwart, *Introduction to Infinite-Dimensional Linear Systems Theory*, 1995.

Lasiecka, Triggiani, *Control Theory for PDEs*, 2000.

Hard to imagine (general) extensions to **nonlinear** and **adaptive**.

Feedback linearization (root of PDE bkst)

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned}$$

Output fbk linearization

$$\begin{aligned} \xi_1 &= h(x) \\ \xi_i &= L_f^{i-1} h(x), \quad i = 1, \ldots, r-1 \qquad \left(L_g L_f^{r-1} h(0) \neq 0 \right) \end{aligned}$$

Chain of integrators/Brunovsky

$$\boxed{\dot{\xi}_i = \xi_{i+1}} \qquad \dot{\xi}_r = L_f^{r-1} h(x) + L_g L_f^{r-1} h(x) u$$

Isidori, *Nonlinear Control Systems*, 1989.

Feedback linearization (root of PDE bkst)

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned}$$

Output fbk linearization

$$\begin{aligned} \xi_1 &= h(x) \\ \xi_i &= L_f^{i-1} h(x), \quad i = 1, \ldots, r-1 \qquad \left(L_g L_f^{r-1} h(0) \neq 0 \right) \end{aligned}$$

Chain of integrators/Brunovsky

$$\boxed{\dot{\xi}_i = \xi_{i+1}} \qquad \dot{\xi}_r = L_f^{r-1} h(x) + L_g L_f^{r-1} h(x) u$$

Isidori, *Nonlinear Control Systems*, 1989.

Feedback linearization (root of PDE bkst)

$$\begin{aligned}\dot{x} &= f(x) + g(x)u \\ y &= h(x)\end{aligned}$$

Output fbk linearization

$$\begin{aligned}\xi_1 &= h(x) \\ \xi_i &= L_f^{i-1} h(x)\,, \quad i = 1, \ldots, r-1 \qquad \left(L_g L_f^{r-1} h(0) \neq 0\right)\end{aligned}$$

Chain of integrators/Brunovsky

$$\boxed{\dot{\xi}_i = \xi_{i+1}} \qquad \dot{\xi}_r = L_f^{r-1} h(x) + L_g L_f^{r-1} h(x) u$$

Isidori, *Nonlinear Control Systems*, 1989.

ODE backstepping (adaptive and robust)

$$\dot{x}_i = x_{i+1} + f_i(x_1, \ldots, x_i)\theta, \qquad x_{n+1} = u$$

Krstic, Kanellakopoulos, Kokotovic, *Nonlinear and Adaptive Control Design*, 1995.

Freeman, Kokotovic, *Robust Nonlinear Control Design*, 1996.

ODE backstepping (adaptive and robust)

$$\dot{x}_i = x_{i+1} + f_i(x_1, \ldots, x_i)\theta\,, \qquad x_{n+1} = u$$

Krstic, Kanellakopoulos, Kokotovic, *Nonlinear and Adaptive Control Design*, 1995.

Freeman, Kokotovic, *Robust Nonlinear Control Design*, 1996.

Early lumping 1: PDE $\longrightarrow$ Brunovsky

1997

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

Early lumping 1: PDE → Brunovsky

1997

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

$$\downarrow \quad \text{spatial discretiz.}$$

$$\dot{u}_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + \lambda_i u_i$$

Early lumping 1: PDE $\longrightarrow$ Brunovsky

1997

$$
\begin{aligned}
u_t &= u_{xx} + \lambda(x)u \\
&\downarrow \qquad \text{spatial discretiz.} \\
\dot{u}_i &= \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + \lambda_i u_i \\
&\downarrow \qquad \text{transform: } u \longrightarrow w = T(h)u \\
\dot{w}_i &= \frac{1}{h^2} w_{i+1} \qquad \text{Brunovsky}
\end{aligned}
$$

Early lumping 1: PDE $\longrightarrow$ Brunovsky

1997

$$
\begin{aligned}
u_t &= u_{xx} + \lambda(x)u \\
&\downarrow \qquad \text{spatial discretiz.} \\
\dot{u}_i &= \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + \lambda_i u_i \\
&\downarrow \qquad \text{transform: } u \longrightarrow w = T(h)u \\
\dot{w}_i &= \frac{1}{h^2} w_{i+1} \qquad \text{Brunovsky}
\end{aligned}
$$

elements of matrix $T(h)$ grow unbounded as $h \longrightarrow 0$

Early lumping 2: Bkst. on space-discretized PDE

$$\dot{u}_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + \lambda_i u_i$$

$$\downarrow$$

$$\dot{w}_i = \frac{w_{i+1} - 2w_i + w_{i-1}}{h^2} - c w_i$$

target sys. in same class

Early lumping 2: Bkst. on space-discretized PDE

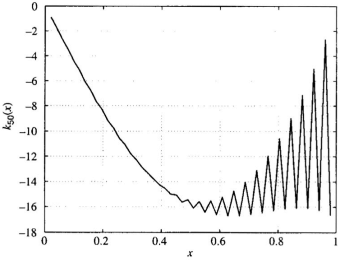
$$\dot{u}_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + \lambda_i u_i$$

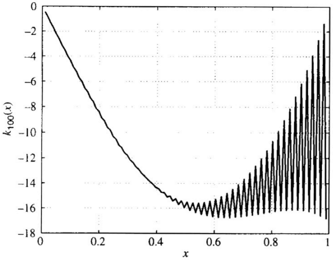
$$\downarrow$$

$$\dot{w}_i = \frac{w_{i+1} - 2w_i + w_{i-1}}{h^2} - c w_i$$

target sys. in same class

Feedback **gain** converges only WEAKLY.





Balogh, Krstic (EJC 2002)

Early lumping 2: Bkst. on space-discretized PDE

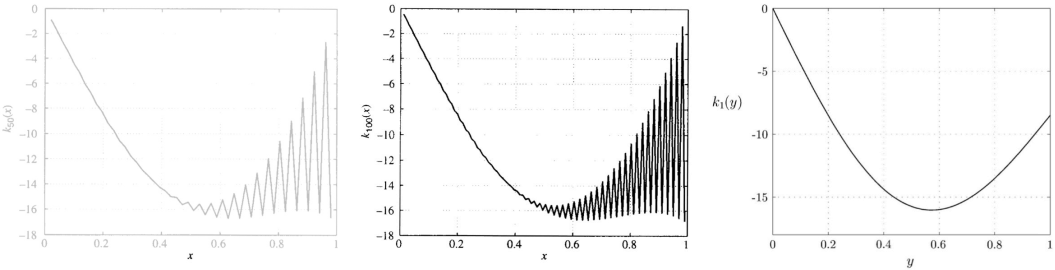
$$\dot{u}_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} + \lambda_i u_i$$

$$\downarrow$$

$$\dot{w}_i = \frac{w_{i+1} - 2w_i + w_{i-1}}{h^2} - c w_i$$

target sys. in same class

Feedback **gain** converges only WEAKLY.



Balogh, Krstic (EJC 2002)

Continuum backstepping for parabolic PIDEs

$$u_t = u_{xx} + \lambda(x)u + b(x)u_x + \int_0^x f(x,y)u(y,t)dy$$

Continuum backstepping for parabolic PIDEs

$$u_t = u_{xx} + \lambda(x)u + b(x)u_x + \int_0^x f(x,y)u(y,t)dy$$

Backstepping transform

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

Continuum backstepping for parabolic PIDEs

$$u_t = u_{xx} + \lambda(x)u + b(x)u_x + \int_0^x f(x,y)u(y,t)dy$$

Backstepping transform

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

to ***target system***

$$w_t = w_{xx} \qquad \text{(exp. stable)}$$

Continuum backstepping for parabolic PIDEs

$$u_t = u_{xx} + \lambda(x)u + b(x)u_x + \int_0^x f(x,y)u(y,t)dy$$

Backstepping transform

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

to ***target system***

$$w_t = w_{xx}$$ (exp. stable)

under kernel PDE

$$\boxed{k_{xx} - k_{yy} = \lambda(x)k}$$ + more terms...

Continuum backstepping for parabolic PIDEs

PDE $\boxed{k_{xx} - k_{yy} = \lambda(x)k}$ is equivalent to integral equation

$$4\, G(\xi,\eta) = -\int_\eta^\xi \lambda\left(\frac{\tau}{2}\right)d\tau + \int_\eta^\xi \int_0^\eta \lambda\left(\frac{\tau - s}{2}\right)G(\tau,s)\,ds\,d\tau$$

Continuum backstepping for parabolic PIDEs

PDE $k_{xx} - k_{yy} = \lambda(x)k$ is equivalent to integral equation

$$4\, G(\xi,\eta) = -\int_\eta^\xi \lambda\left(\frac{\tau}{2}\right)d\tau + \int_\eta^\xi \int_0^\eta \lambda\left(\frac{\tau - s}{2}\right)G(\tau,s)\,ds\,d\tau$$

Smyshlyaev's (TAC, 2004) explicit solution for λ = const:

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

I_1 = Bessell function

Smyshlyaev, Krstic (TAC 2004)

(with contributions from Boskovic, 2001; Liu, 2003)

Earlier (independent) **encounters in mathematics**

Transform $w(x,t) = u(x,t) - \int_0^{x} k(x,y,t)u(y,t)dy$

- Colton (JDE 1977): to solve $u_t = u_{xx} + \lambda(x,t)u$ by solving the heat equation $w_t = w_{xx}$

  Analogous to using (in limited cases) a diffeomorphism only to solve a nonlinear ODE using a linear ODE.

- Seidman (Appl. Math. Optim. 1984):

  Proves that cost of null controllability of $u_t = u_{xx} + \lambda(x,t)u$ in time T scales as $e^{O(1/T)}$.

No feedback of $u(x,t)$

Earlier (independent) **encounters in mathematics**

Transform $w(x,t) = u(x,t) - \int_0^{x} k(x,y,t)u(y,t)dy$

- Colton (JDE 1977): to solve $u_t = u_{xx} + \lambda(x,t)u$ by solving the heat equation $w_t = w_{xx}$

  Analogous to using (in limited cases) a diffeomorphism only to solve a nonlinear ODE using a linear ODE.

- Seidman (Appl. Math. Optim. 1984):

  Proves that cost of null controllability of $u_t = u_{xx} + \lambda(x,t)u$ in time T scales as $e^{O(1/T)}$.

No feedback of $u(x,t)$

Earlier (independent) encounters in mathematics

Transform $w(x,t) = u(x,t) - \int_0^{x} k(x,y,t)u(y,t)dy$

- Colton (JDE 1977): to solve $u_t = u_{xx} + \lambda(x,t)u$ by solving the heat equation $w_t = w_{xx}$

  Analogous to using (in limited cases) a diffeomorphism only to solve a nonlinear ODE using a linear ODE.

- Seidman (Appl. Math. Optim. 1984):

  Proves that cost of null controllability of $u_t = u_{xx} + \lambda(x,t)u$ in time T scales as $e^{\mathcal{O}(1/T)}$.

No feedback of $u(x,t)$

Nonlinear bkst transform (true FL)

For nonlinear $u_t = u_{xx} + F[u]$, infinite nonlinear Volterra series:

$$w(x,t) = u(x,t) - \sum_{n=1}^{\infty} \int_0^x \int_0^{\xi_1} \cdots \int_0^{\xi_{n-1}} k_n(x, \xi_1, \ldots, \xi_n) \left(\prod_{j=1}^{n} u(\xi_j, t) \right) d\xi_n \cdots d\xi_1$$

Vazquez, Krstic (Automatica 2008)

Nonlinear bkst transform (true FL)

For nonlinear $u_t = u_{xx} + F[u]$, infinite nonlinear Volterra series:

$$w(x,t) = u(x,t) - \sum_{n=1}^{\infty} \int_0^x \int_0^{\xi_1} \cdots \int_0^{\xi_{n-1}} k_n(x, \xi_1, \ldots, \xi_n) \left(\prod_{j=1}^{n} u(\xi_j, t) \right) d\xi_n \cdots d\xi_1$$

Kernel PDEs on domains $\{0 \le \xi_n \le \cdots \le \xi_1 \le x\}$ of increasing dimensions:

$$\partial_{xx} k_n = \partial_{\xi_1 \xi_1} k_n + \cdots + \partial_{\xi_n \xi_n} k_n$$ + lin. nonlocal terms in $k_1, \ldots, k_n$

Vazquez, Krstic (Automatica 2008)

Backstepping **observers**

Luenberger-type

$$\underbrace{\hat{u}_t = \hat{u}_{xx} + \lambda(x)\hat{u}}_{\text{copy of plant}} + p(x)\underbrace{\left[u(0,t) - \hat{u}(0,t)\right]}_{\text{output estim. error}}$$

Backstepping **observers**

Luenberger-type

$$\underbrace{\hat{u}_t \;=\; \hat{u}_{xx} + \lambda(x)\hat{u}}_{\text{copy of plant}} \;+\; p(x)\underbrace{\Big[u(0,t) - \hat{u}(0,t)\Big]}_{\text{output estim. error}}$$

Bkst-transformable to error system

$$\tilde{w}_t = \tilde{w}_{xx}$$

Backstepping **observers**

Luenberger-type

$$\underbrace{\hat{u}_t = \hat{u}_{xx} + \lambda(x)\hat{u}}_{\text{copy of plant}} + p(x) \underbrace{\left[u(0,t) - \hat{u}(0,t) \right]}_{\text{output estim. error}}$$

Bkst-transformable to error system

$$\tilde{w}_t = \tilde{w}_{xx}$$

Inspired by bkst observers of Krener-Kang (2003) and earlier Krener-Isidori (1983).

Smyshlyaev, Krstic (SCL 2005)

Hyperbolic (1st-order) PDEs

$$u_t = u_x + \int_0^x f(x,y)u(y,t)dy$$

Smyshlyaev, Krstic (SCL 2008)

Hyperbolic (1st-order) PDEs

$$u_t = u_x + \int_0^x f(x,y)u(y,t)dy$$

transformed into transport PDE

$$w_t = w_x \,, \qquad w(1,t) = 0$$

Settles in $T = 1$ sec.

Smyshlyaev, Krstic (SCL 2008)

Coupled (multiple) **hyperbolic PDEs**

$$
\begin{aligned}
u_t &= -\epsilon_1(x)u_x + \varphi_1(u,v,x)u_x + \psi_1(u,v,x)v_x + f_1(u,v,x) \\
v_t &= \epsilon_2(x)v_x + \varphi_2(u,v,x)u_x + \psi_2(u,v,x)v_x + f_2(u,v,x)
\end{aligned}
$$

Coron, Vazquez, Krstic, Bastin (SICON 2013)

(380 GS citations)

Coupled (multiple) **hyperbolic PDEs**

$$
\begin{aligned}
u_t &= -\epsilon_1(x)u_x \quad + \varphi_1(u,v,x)u_x + \psi_1(u,v,x)v_x \ + f_1(u,v,x) \\
v_t &= \epsilon_2(x)v_x \quad + \varphi_2(u,v,x)u_x + \psi_2(u,v,x)v_x \ + f_2(u,v,x)
\end{aligned}
$$

Local stabilization in H^2.

Coron, Vazquez, Krstic, Bastin (SICON 2013)

(380 GS citations)

$(n+m) \times (n+m)$ counterconvecting PDEs

- $u \in \mathbb{R}^n$ rightward, unactuated
- $v \in \mathbb{R}^m$ leftward, actuated

$(n+m) \times (n+m)$ counterconvecting PDEs

- $u \in \mathbb{R}^n$ rightward, unactuated
- $v \in \mathbb{R}^m$ leftward, actuated

- $m = 1$: Di Meglio, Vazquez, Krstic (TAC, 2013)
- $m, n > 1$: Hu, Di Meglio, Vazquez, Krstic (2016, 2019)
- hyp. with zero characteristic speeds: de Andrade, Vazquez, Karafyllis, Krstic (TAC)

$(n+m) \times (n+m)$ counterconvecting PDEs

- $u \in \mathbb{R}^n$ rightward, unactuated
- $v \in \mathbb{R}^m$ leftward, actuated

- $m = 1$: Di Meglio, Vazquez, Krstic (TAC, 2013)
- $m, n > 1$: Hu, Di Meglio, Vazquez, Krstic (2016, 2019)
- hyp. with zero characteristic speeds: de Andrade, Vazquez, Karafyllis, Krstic (TAC)

Applications: water canals, oil drilling, laser chambers, traffic

Academic application 1: Navier-Stokes

$$
\begin{aligned}
u_t &= \frac{1}{\mathrm{Re}}\Delta u - uu_x - vu_y - wu_z - p_x \\
v_t &= \frac{1}{\mathrm{Re}}\Delta v - uv_x - vv_y - wv_z - p_y \\
w_t &= \frac{1}{\mathrm{Re}}\Delta w - uw_x - vw_y - ww_z - p_y
\end{aligned}
$$

Vazquez, Krstic, *Control of Turbulent and Magnetohydrodynamic Channel Flows*, 2007

Academic application 1: Navier-Stokes

$$
\begin{aligned}
u_t &= \frac{1}{\mathrm{Re}}\Delta u - u u_x - v u_y - w u_z - p_x \\
v_t &= \frac{1}{\mathrm{Re}}\Delta v - u v_x - v v_y - w v_z - p_y \\
w_t &= \frac{1}{\mathrm{Re}}\Delta w - u w_x - v w_y - w w_z - p_y
\end{aligned}
$$

Convert to heat equation (with convection) for unstable wave numbers.

Vazquez, Krstic, *Control of Turbulent and Magnetohydrodynamic Channel Flows*, 2007

Academic application 2: unstable **Timoshenko beam**

$$\begin{aligned}(\varepsilon\partial_{tt} - \partial_{xx})\, u &= -\alpha_x \\ (\mu\partial_{tt} - \partial_{xx})\, \alpha &= \frac{a}{\varepsilon}(u_x - \alpha)\end{aligned}$$

with boundary anti-damping

Academic application 2: unstable **Timoshenko beam**

$$\begin{aligned} (\varepsilon \partial_{tt} - \partial_{xx})\, u &= -\alpha_x \\ (\mu \partial_{tt} - \partial_{xx})\, \alpha &= \frac{a}{\varepsilon}(u_x - \alpha) \end{aligned}$$

with boundary anti-damping

Converted to $n = m = 2$ coupled hyperbolic systems

Chen, Vazquez, Krstic (TAC 2024)

Delays/predictors by (hyperbolic) backstepping

$$\dot{X} = AX + BU(t-D)$$

$$\dot{X} = f(X, U(t-D))$$

$$\dot{X} = f(X, U(t - D(X,t))), \qquad D \text{ depends on } \textit{current } X$$

$$\dot{X} = f\left(X, U\left((\mathrm{Id} + D \circ X)^{-1}(t)\right)\right), \qquad D \text{ depends on } \textit{past } X$$

Delays/predictors by (hyperbolic) backstepping

$$
\begin{aligned}
\dot{X} &= AX + BU(t-D) \\
\dot{X} &= f(X, U(t-D)) \\
\dot{X} &= f(X, U(t-D(X,t)))\,, \qquad D \text{ depends on } \textit{current } X \\
\dot{X} &= f\left(X, U\left((\mathrm{Id} + D \circ X)^{-1}(t)\right)\right)\,, \qquad D \text{ depends on } \textit{past } X
\end{aligned}
$$

Delays/predictors by (hyperbolic) **backstepping**

$$\dot{X} = AX + BU(t - D)$$

$$\dot{X} = f(X, U(t - D))$$

$$\dot{X} = f(X, U(t - D(X, t))), \qquad D \text{ depends on } \textit{current } X$$

$$\dot{X} = f\left(X, U\left((\mathrm{Id} + D \circ X)^{-1}(t)\right)\right), \qquad D \text{ depends on } \textit{past } X$$

Delays/predictors by (hyperbolic) **backstepping**

$$
\begin{aligned}
\dot{X} &= AX + BU(t-D) \\
\dot{X} &= f(X, U(t-D)) \\
\dot{X} &= f(X, U(t - D(X, t))), && D \text{ depends on } \textit{current } X \\
\dot{X} &= f\left(X, U\left((\mathrm{Id} + D \circ X)^{-1}(t)\right)\right), && D \text{ depends on } \textit{past } X
\end{aligned}
$$

Delays/predictors by (hyperbolic) backstepping

$$\dot{X} = AX + BU(t - D)$$

$$\dot{X} = f(X, U(t - D))$$

$$\dot{X} = f(X, U(t - D(X, t))) , \qquad D \text{ depends on } \textit{current } X$$

$$\dot{X} = f\left(X, U\left((\mathrm{Id} + D \circ X)^{-1}(t)\right)\right) , \qquad D \text{ depends on } \textit{past } X$$

Krstic, *Delay Compensation for Nonlinear, Adaptive, and PDE Systems*, 2009

Bekiaris-Liberis, Krstic, *Nonlinear Control Under Nonconstant Delays*, 2013

Karafyllis, Krstic, *Predictor Fbk for Delay Systems: Implementations and Approximations*, 2013

Delays/predictors by (hyperbolic) **backstepping**

$$\begin{aligned}
\dot{X} &= AX + BU(t-D) \\
\dot{X} &= f(X, U(t-D)) \\
\dot{X} &= f(X, U(t-D(X,t))), \qquad D \text{ depends on } \textit{current } X \\
\dot{X} &= f\left(X, U\left((\mathrm{Id} + D \circ X)^{-1}(t)\right)\right), \qquad D \text{ depends on } \textit{past } X
\end{aligned}$$

Krstic, *Delay Compensation for Nonlinear, Adaptive, and PDE Systems*, 2009

Bekiaris-Liberis, Krstic, *Nonlinear Control Under Nonconstant Delays*, 2013

Karafyllis, Krstic, *Predictor Fbk for Delay Systems: Implementations and Approximations*, 2013

Delays/predictors by (hyperbolic) backstepping

$$\dot{X} = AX + BU(t - D)$$

$$\dot{X} = f(X, U(t - D))$$

$$\dot{X} = f(X, U(t - D(X, t))), \qquad D \text{ depends on } \textit{current } X$$

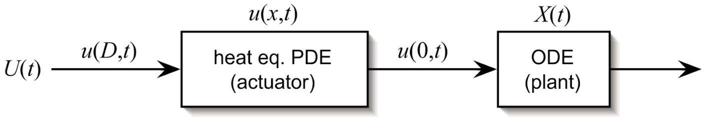
$$\dot{X} = f\left(X, U\left((\mathrm{Id} + D \circ X)^{-1}(t)\right)\right), \qquad D \text{ depends on } \textit{past } X$$

Krstic, *Delay Compensation for Nonlinear, Adaptive, and PDE Systems*, 2009

Bekiaris-Liberis, Krstic, *Nonlinear Control Under Nonconstant Delays*, 2013

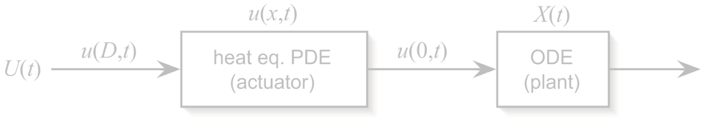
Karafyllis, Krstic, *Predictor Fbk for Delay Systems: Implementations and Approximations*, 2013

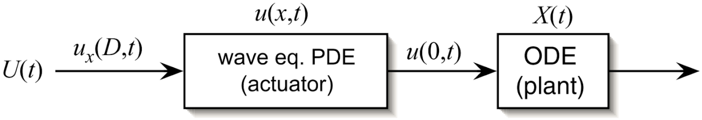
Other PDE-ODE and PDE-PDE cascades



Krstic, *Delay Compensation for Nonlinear, Adaptive, and PDE Systems*, 2009

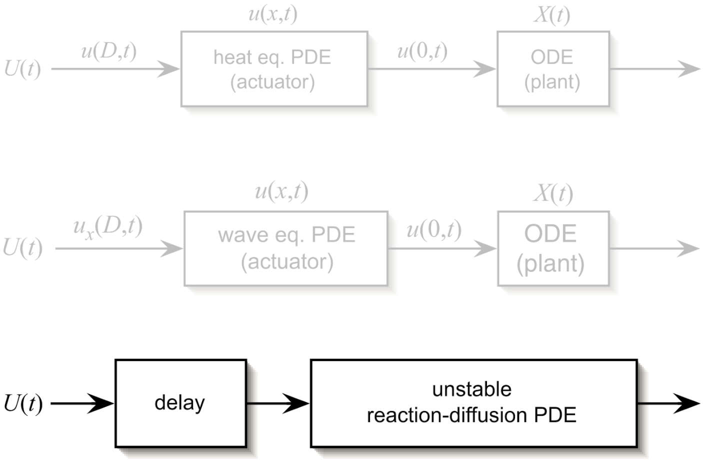
Other PDE-ODE and PDE-PDE cascades





Krstic, *Delay Compensation for Nonlinear, Adaptive, and PDE Systems*, 2009

Other PDE-ODE and PDE-PDE cascades



Krstic, *Delay Compensation for Nonlinear, Adaptive, and PDE Systems*, 2009

Adaptive control of parabolic PDEs

$$u_t = u_{xx} + \lambda(x)u + \int_0^x f(x,\xi)u(\xi,t)d\xi$$

λ, f – unknown, u – unmeasured

Smyshlyaev, Krstic, *Adaptive Control of Parabolic PDEs*, 2010

Adaptive control of hyperbolic PDEs

$$\begin{aligned} u_t &= -\epsilon_1 u_x + c_1(x) v \\ v_t &= \epsilon_2 v_x + c_2(x) u \end{aligned}$$

c_1, c_2 – unknown, u, v – unmeasured

Anfinsen, Aamo, *Adaptive Control of Hyperbolic PDEs*, 2020

Delay-adaptive control

$$\dot{X} = AX + BU(t - D)$$

D – unknown, A, B – also unknown

Bresch-Pietri, Krstic (2009)

Delay-adaptive control

$$\dot{X} = AX + BU(t - D)$$

D – unknown, A, B – also unknown

Bresch-Pietri, Krstic (2009)

Extensions: unmeasured X, multiple inputs/delays, distributed delays

Zhu, Krstic, *Delay-Adaptive Linear Control*, 2020

Output regulator by PDE bkst

Inspiration from Byrnes-Isidori.

Parabolic

$$u_t = u_{xx} + \lambda(x) + g(x)\delta(t)\,, \qquad \dot{\delta} = S\delta$$

Output regulator by PDE bkst

Inspiration from Byrnes-Isidori.

Parabolic

$$u_t = u_{xx} + \lambda(x) + g(x)\delta(t)\,, \qquad \dot{\delta} = S\delta$$

Output regulator by PDE bkst

Inspiration from Byrnes-Isidori.

Parabolic

$$u_t = u_{xx} + \lambda(x) + g(x)\delta(t), \qquad \dot{\delta} = S\delta$$

Regulator equation:

$$m'' = S^T m + g(x) - \int_0^x k(x,y) g(y) dy$$
$$m(0) = m'(0) = 0$$

Deutscher (Automatica, 2015) **(parabolic)**

Deutscher (Automatica, 2017) **(hyperbolic)**

Output regulator by PDE bkst

Inspiration from Byrnes-Isidori.

Parabolic

$$u_t = u_{xx} + \lambda(x) + g(x)\delta(t)\,, \qquad \dot{\delta} = S\delta$$

Regulator equation:

$$m'' = S^T m + g(x) - \int_0^x k(x,y)g(y)dy$$
$$m(0) = m'(0) = 0$$

Deutscher (Automatica, 2015) **(parabolic)**

Deutscher (Automatica, 2017) **(hyperbolic)**

Prescribed-time stabilization

$$u_t = u_{xx}$$

Coron, Nguyen (ARMA 2017)

Null-controllability by bkst time-varying fbk with PW-const gains that $\nearrow \infty$ as $t \to T$.

Prescribed-time stabilization

$$u_t = u_{xx}$$

Coron, Nguyen (ARMA 2017)

Null-controllability by bkst time-varying fbk with PW-const gains that $\nearrow \infty$ as $t \to T$.

Continuous backstepping gain:

$$k(x,y,t) = -y\mu(t)\, \mathrm{e}^{\sqrt{\mu(t)}(x^2-y^2)/4} \frac{I_1\left(\sqrt{\mu(t)(x^2-y^2)}\right)}{\sqrt{\mu(t)(x^2-y^2)}}$$

$$\mu(0) = 1/T^2, \qquad \lim_{t\to T} \mu(t) = \infty$$

Espitia, Polyakov, Efimov, Perruquetti (Automatica 2019)

Fredholm backstepping: Basic

$$w(x,t) = u(x,t) - \int_0^1 k(x,y)u(y,t)dy$$

Not automatically invertible.

Fredholm backstepping: Basic

$$w(x,t) = u(x,t) - \int_0^{1} k(x,y)u(y,t)dy$$

Not automatically invertible.

$$u_t = u_{x(x)} + \int_0^{1} f(x,y)u(y,t)dy$$

Parabolic: Tsubakino, Bribiesca Argomedo, Krstic (CDC 2014)

Hyperbolic: Bribiesca Argomedo, Krstic (TAC 2015). Generalized in Coron, Hu, Olive (J. Func. Anal. 2016).

Fredholm backstepping: Basic

$$w(x,t) = u(x,t) - \int_0^{1} k(x,y)u(y,t)dy$$

Not automatically invertible.

$$u_t = u_{x(x)} + \int_0^{1} f(x,y)u(y,t)dy$$

Parabolic: Tsubakino, Bribiesca Argomedo, Krstic (CDC 2014)

Hyperbolic: Bribiesca Argomedo, Krstic (TAC 2015). Generalized in Coron, Hu, Olive (J. Func. Anal. 2016).

Fredholm backstepping: Basic

$$w(x,t) = u(x,t) - \int_0^1 k(x,y)u(y,t)dy$$

Not automatically invertible.

$$u_t = u_{x(x)} + \int_0^1 f(x,y)u(y,t)dy$$

Parabolic: Tsubakino, Bribiesca Argomedo, Krstic (CDC 2014)

Hyperbolic: Bribiesca Argomedo, Krstic (TAC 2015). Generalized in Coron, Hu, Olive (J. Func. Anal. 2016).

Fredholm backstepping: Advanced

Korteweg-de Vries: $u_t + u_{xxx} + u_x + uu_x = 0$ Coron, Lü (JMPA 2014)

Kuramoto-Sivashinsky: $u_t + u_{xxxx} + \lambda u_{xx} + uu_x = 0$ Coron, Lü (JDE 2015)

Schrödinger (bilinear): $i\psi_t + \Delta\psi + u(t)\mu(x)\psi = 0$ Coron, Gagnon, Morancey (JMPA 2018)

Fredholm backstepping: Advanced

Korteweg-de Vries: $u_t + u_{xxx} + u_x + uu_x = 0$ Coron, Lü (JMPA 2014)

Kuramoto-Sivashinsky: $u_t + u_{xxxx} + \lambda u_{xx} + uu_x = 0$ Coron, Lü (JDE 2015)

Schrödinger (bilinear): $i\psi_t + \Delta\psi + u(t)\mu(x)\psi = 0$ Coron, Gagnon, Morancey (JMPA 2018)

Fredholm backstepping: Advanced

Korteweg-de Vries: $u_t + u_{xxx} + u_x + uu_x = 0$ Coron, Lü (JMPA 2014)

Kuramoto-Sivashinsky: $u_t + u_{xxxx} + \lambda u_{xx} + uu_x = 0$ Coron, Lü (JDE 2015)

Schrödinger (bilinear): $i\psi_t + \Delta\psi + u(t)\mu(x)\psi = 0$ Coron, Gagnon, Morancey (JMPA 2018)

Applications: 1 Additive manufacturing

Stefan model of phase change

$$T_t = \alpha T_{xx} \qquad T(s,t) = T_{\mathrm{m}} \quad \text{melting temp. at interface}$$

$$\dot{s} = -\beta T_x(s,t) \qquad \text{nonlin. ODE for motion of liquid-solid interface}$$

Koga, Krstic, *Materials Phase Change PDE Control and Estimation*, 2020

(experiment)

Applications: 1 Additive manufacturing

Stefan model of phase change

$$T_t = \alpha T_{xx} \qquad T(s,t) = T_{\mathrm{m}} \qquad \text{melting temp. at interface}$$

$$\dot{s} = -\beta T_x(s,t) \qquad \text{nonlin. ODE for motion of liquid-solid interface}$$

Koga, Krstic, *Materials Phase Change PDE Control and Estimation*, 2020

(experiment)

Applications: 2 Traffic flow

Stop-and-go instability:

$$\partial_t \rho + \partial_x (\rho v) = 0 \qquad \text{LWR}$$

$$\partial_t (v + p(\rho)) + v \partial_x (v + p(\rho)) = \frac{V(\rho) - v}{\tau} \qquad \text{Aw-Rascle}$$

Yu, Krstic, *Traffic Congestion Control by PDE Backstepping*, 2022

Applications: More

- **Water canals**: Coron, Bastin; Diagne
- **Oil drilling**: Di Meglio, Arsnes, Aamo, Hasan, Sagert, Auriol, Bresch-Pietri, Krstic
  Experiments STATOIL, Total
- **Li-ion batteries**: Moura, Chaturvedi, Camacho, Koga, Krstic
  Experiments: Robert Bosch

Applications: More

- **Water canals**: Coron, Bastin; Diagne

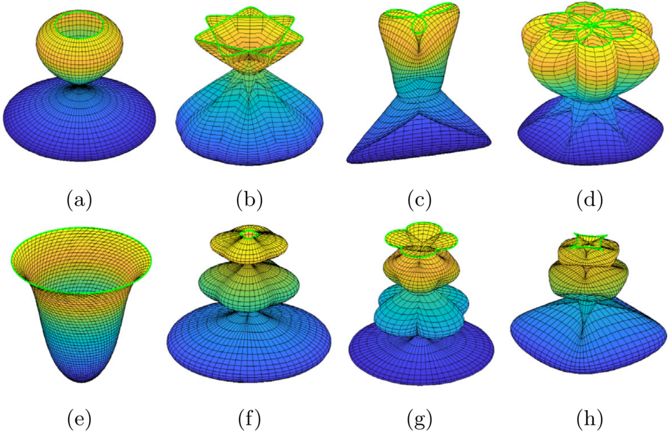
- **Oil drilling**: Di Meglio, Arsnes, Aamo, Hasan, Sagert, Auriol, Bresch-Pietri, Krstic
  Experiments STATOIL, Total

- **Li-ion batteries**: Moura, Chaturvedi, Camacho, Koga, Krstic
  Experiments: Robert Bosch

Applications: More

- **Water canals**: Coron, Bastin; Diagne
- **Oil drilling**: Di Meglio, Arsnes, Aamo, Hasan, Sagert, Auriol, Bresch-Pietri, Krstic
  Experiments STATOIL, Total
- **Li-ion batteries**: Moura, Chaturvedi, Camacho, Koga, Krstic
  Experiments: Robert Bosch

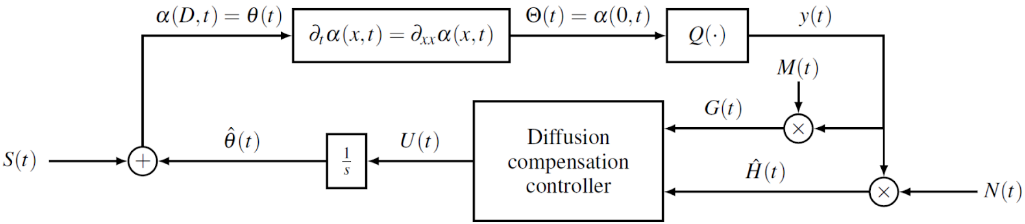
Vehicle swarm deployment by PDE bkst



Open-loop instability → rich deployment geometries

Extremum seeking for PDEs

Delays, parabolic, wave, games



Oliveira, Krstic, *Extremum Seeking through Delays and PDEs*, 2022

Machine learning for PDE backstepping

Kernel operator

$$\mathcal{K}: \ \lambda(x) \ \mapsto \ k(x,y)$$

Neural operator approximation

$$\lambda \longrightarrow \boxed{\widehat{\mathcal{K}}} \longrightarrow k$$

Learn the kernel mapping $\lambda \mapsto k$ “once-and-for-all,” for all λ.

Machine learning for PDE backstepping

Kernel operator

$$\mathcal{K}: \ \lambda(x) \ \mapsto \ k(x,y)$$

Neural operator approximation

$$\lambda \longrightarrow \boxed{\widehat{\mathcal{K}}} \longrightarrow k$$

Learn the kernel mapping $\lambda \mapsto k$ "once-and-for-all," for all λ.

Machine learning for PDE backstepping

Kernel operator

$$\mathcal{K}: \ \lambda(x) \ \mapsto \ k(x,y)$$

Neural operator approximation

$$\lambda \longrightarrow \boxed{\widehat{\mathcal{K}}} \longrightarrow k$$

Learn the kernel mapping $\lambda \mapsto k$ "once-and-for-all," for all λ.

- Hyperbolic: Bhan, Shi, Krstic (TAC 2024)
- Parabolic: Krstic, Bhan, Shi (Automatica 2024)
- Delay-compensated: Diagne, Qi, Wang (2024)
- Adaptive, gain scheduling: Lamarque, Krstic, Bhan, etc. (2024)

Machine learning for PDE backstepping

Kernel operator

$$\mathcal{K} : \ \lambda(x) \ \mapsto \ k(x,y)$$

Neural operator approximation

$$\lambda \longrightarrow \boxed{\widehat{\mathcal{K}}} \longrightarrow k$$

Learn the kernel mapping $\lambda \mapsto k$ "once-and-for-all," for all λ.

- Hyperbolic: Bhan, Shi, Krstic (TAC 2024)
- Parabolic: Krstic, Bhan, Shi (Automatica 2024)
- Delay-compensated: Diagne, Qi, Wang (2024)
- Adaptive, gain scheduling: Lamarque, Krstic, Bhan, etc. (2024)

Machine learning for PDE backstepping

Kernel operator

$$\mathcal{K}: \ \lambda(x) \ \mapsto \ k(x,y)$$

Neural operator approximation

$$\lambda \longrightarrow \boxed{\widehat{\mathcal{K}}} \longrightarrow k$$

Learn the kernel mapping $\lambda \mapsto k$ "once-and-for-all," for all λ.

- Hyperbolic: Bhan, Shi, Krstic (TAC 2024)
- Parabolic: Krstic, Bhan, Shi (Automatica 2024)
- Delay-compensated: Diagne, Qi, Wang (2024)
- Adaptive, gain scheduling: Lamarque, Krstic, Bhan, etc. (2024)

How to begin learning PDE backstepping?

Boundary Control of PDEs:
A Course on Backstepping Designs

SIAM 2008