Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 52-2 (2019) 76–81

Folding Backstepping Approach to Parabolic PDE Bilateral Boundary Control

Stephen Chen * **Rafael Vazquez** ** **Miroslav Krstic** ***

* *Department of Mechanical and Aerospace Engineering, University of California, San Diego, La Jolla, CA 92093-0411 USA (e-mail: stc007@ucsd.edu)*
** *Departmento de Ingeniería Aeroespacial, Universidad de Sevilla, 41092 Sevilla, Spain (e-mail: rvazquez1@us.es)*
*** *Department of Mechanical and Aerospace Engineering, University of California, San Diego, La Jolla, CA 92093-0411 USA (e-mail: krstic@ucsd.edu)*

Abstract: We consider the stabilization problem for an unstable 1-D diffusion-reaction partial differential equation using a so-called folding transformation. The diffusion-reaction equation is transformed into a 2×2 system of coupled parabolic PDEs with exotic boundary conditions. A first backstepping transformation is designed to map the unstable system into a strict-feedback intermediate target system. A second backstepping transformation is designed to stabilize the intermediate target system. Interestingly, the companion gain kernel PDEs contain the folding boundary condition, exhibiting symmetry with the original system. The kernels posses a cascading structure that allows for sequential solution methods. Finally, the controller derived is shown to be exponentially stabilizing in the L^2 sense.

Keywords: Lyapunov-based and backstepping techniques, distributed parameter systems, boundary control

1. INTRODUCTION

More recently, boundary control of partial differential equations (PDEs) has been a focus of research. In particular, the backstepping approach has seen much popularity for one dimensional PDEs. The reference text in Krstic and Smyshlyaev (2008) disseminates the classical backstepping approach for various 1-D PDEs. It is important to note, however, that much of this theory is *unilateral* backstepping designs, that is, a single controller on one boundary.

Research into backstepping designs for arbitrarily high dimensional parabolic PDE has also been achieved, although not without restrictive conditions on the domain geometry and radial symmetry (Vazquez and Krstic (2017a)). In higher dimensional backstepping boundary control, the boundary hypersurface is actuated–for example, the unit disk in 2-D necessitates the unit circle as its boundary control. Aligning the higher dimensional perspective back into a 1-D context, one is led to believe the classical PDE backstepping results imposing only a single controller at one boundary should be able to be extended to a case where two controllers (albeit intrinsically linked) can be designed for actuation at both boundaries. This has motivated research into the so-called *bilateral* backstepping control for one dimensional parabolic PDE (Vazquez and Krstic (2016)). However, the existing approach is limited in using a symmetric backstepping transformation, which is potentially limiting if one is to begin to consider spatially-varying coefficients. Some results exist for bilateral control in different PDE classes, namely, systems of hyperbolic PDE (Auriol and Meglio (2017)) and viscous Hamilton-Jacobi PDE (Bekiaris-Liberis and Vazquez (2018)).

In hyperbolic PDEs (particularly first-order), the notion of "folding" the system state in space is easily achievable. In the first-order case, one can merely seperate the PDE into two spatial domains, and impose a first-order compatibility condition on the resulting internal boundary. The method extends to higher-order hyperbolic PDE, which can be decomposed to systems of first-order hyperbolic PDE, and folded. This is contrasted with parabolic PDE, which require second-order compatibility conditions on the resulting internal boundary. Moreover, the very nature of diffusion implies a bidirectional flow of information through these compatibility conditions (as opposed to unidirectional flow in hyperbolic PDE). It is for this reason that "folding" parabolic systems becomes more difficult than for their hyperbolic brethern. Nevertheless, folding a parabolic PDE will admit a coupled parabolic PDE system.

Coupled parabolic PDE systems have been studied prior, although perhaps not to the same extent that hyperbolic PDE have enjoyed. Several results exist for various coupling structures and in various problem contexts. In Vazquez and Krstic (2017b), boundary control for parabolic PDE systems with spatially varying diffusion and reaction coefficients are considered. In Deutscher and Kerschbaum (2018), the previous result was extended to include non-local coupling terms as well as extending boundary conditions. In Orlov et al. (2017), constant coefficient parabolic PDE systems are considered in the output-feedback scenario. In Tsubakino et al. (2013), cascaded parabolic PDE systems are studied, which is differentiated from the previous results by including boundary coupling rather than merely in-domain (interior) coupling.

2405-8963 © 2019, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2019.08.014

This paper achieves an initial result in bilateral control design for a parabolic PDE by imposing a "folding" transformation and designing a set of controllers for the resulting coupled parabolic PDE system with exotic boundary conditions coupling the systems.

2. MODEL AND FOLDING TRANSFORMATION

Consider the following unstable reaction-diffusion system with a spatially-varying reaction coefficient:

$$\partial_t u(y,t) = \varepsilon \partial_y^2 u(y,t) + \lambda(y) u(y,t) \tag{1}$$
$$u(-1,t) = \mathcal{U}_1(t) \tag{2}$$
$$u(1,t) = \mathcal{U}_2(t) \tag{3}$$

Where $y \in (-1,1)$.

We select an arbiitary point $y_0 \in (-1,1)$ to "fold" the parabolic equation (1) around. We first use the following piecewise definition of $u(y,t)$ as:

$$u(y,t) = \begin{cases} u_1(y,t) & y \in (-1, y_0) \\ u_2(y,t) & y \in (y_0, 1) \end{cases} \tag{4}$$

and define the following piecewise spatial transformation in y:

$$x_1 = (y_0 - y)/(1+y_0) \qquad y \in (-1, y_0) \tag{5}$$
$$x_2 = (y - y_0)/(1-y_0) \qquad y \in (y_0, 1) \tag{6}$$

Note that $x_1, x_2 \in [0,1]$. This set of scaling and folding transformations will allow us to map the system (1)-(3) into the following system in matrix form:

$$\partial_t U(x,t) = E\partial_x^2 U(x,t) + \Lambda(x) U(x,t) \tag{7}$$
$$\alpha U_x(0,t) = -\beta U(0,t) \tag{8}$$
$$U(1,t) = \mathcal{U}(t) \tag{9}$$

We have dropped the subscript indexing on x for simplicity. The state $U : [0,1] \times [0,\infty) \to \mathbb{R}^2$ is defined as

$$U(x,t) := \begin{pmatrix} u_1(y_0 - (1+y_0)x, t) \\ u_2(y_0 + (1-y_0)x, t) \end{pmatrix} \tag{10}$$

while the control $\mathcal{U} : [0,\infty) \to \mathbb{R}^2$ is defined as

$$\mathcal{U} := \begin{pmatrix} \mathcal{U}_1(t) \\ \mathcal{U}_2(t) \end{pmatrix} \tag{11}$$

The parameter matrices are defined by

$$E := \mathrm{diag}(\varepsilon_1, \varepsilon_2) \tag{12}$$
$$\Lambda(x) := \mathrm{diag}(\lambda_1(x), \lambda_2(x)) \tag{13}$$
$$\alpha := \begin{pmatrix} 1 & a \\ 0 & 0 \end{pmatrix} \tag{14}$$
$$\beta := \begin{pmatrix} 0 & 0 \\ 1 & -1 \end{pmatrix} \tag{15}$$
$$\varepsilon_1 = \frac{\varepsilon}{(1+y_0)^2} \tag{16}$$
$$\varepsilon_2 = \frac{\varepsilon}{(1-y_0)^2} \tag{17}$$
$$\lambda_1(x) = \lambda(y_0 - (1+y_0)x) \tag{18}$$
$$\lambda_2(x) = \lambda(y_0 + (1-y_0)x) \tag{19}$$
$$a = (1+y_0)/(1-y_0) \tag{20}$$

The conditions encoded in the α, β matrices arise from imposing compatibility conditions between u_1 and u_2 at $x = 0$.

Assumption 1. We will assume that $\varepsilon_1 > \varepsilon_2$ (i.e., $-1 < y_0 < 0$) without loss of generality. To consider the opposing case, one may merely define $\bar{y} = -y$ initially and proceed with the same design outlined.

3. BACKSTEPPING CONTROL DESIGN

We use two sets of backstepping transformations. The first is modified from the transformation found in Vazquez and Krstic (2017b) to account for the folding conditions encapsulated in (8).

The first transformation is

$$V(x,t) = U(x,t) - \int_0^x K(x,y) U(y,t) dy \tag{21}$$

where $K(x,y) : \mathcal{T}_K \to \mathbb{R}^{2\times 2}$ are the kernels of the transformation on the domain $\mathcal{T}_K := \{(x,y) \in \mathbb{R}^2 | 0 \leq y \leq x \leq 1\}$. The kernels componentwise (with i-th row and j-th column of K notated with k_{ij}) satisfy a 2×2 coupled system of second-order hyperbolic PDE. We employ an approach like Vazquez and Krstic (2017b), where we use the following definition:

$$L(x,y) = \sqrt{E}\partial_x K(x,y) + \partial_y K(x,y)\sqrt{E} \tag{22}$$

allowing us to transform the K kernel into a system of coupled first-order hyperbolic PDE (K, L):

$$\sqrt{\varepsilon_1}\partial_x k_{11}(x,y) + \sqrt{\varepsilon_1}\partial_y k_{11}(x,y) = l_{11}(x,y) \tag{23}$$
$$\sqrt{\varepsilon_1}\partial_x k_{12}(x,y) + \sqrt{\varepsilon_2}\partial_y k_{12}(x,y) = l_{12}(x,y) \tag{24}$$
$$\sqrt{\varepsilon_2}\partial_x k_{21}(x,y) + \sqrt{\varepsilon_1}\partial_y k_{21}(x,y) = l_{21}(x,y) \tag{25}$$
$$\sqrt{\varepsilon_2}\partial_x k_{22}(x,y) + \sqrt{\varepsilon_2}\partial_y k_{22}(x,y) = l_{22}(x,y) \tag{26}$$
$$\sqrt{\varepsilon_1}\partial_x l_{11}(x,y) - \sqrt{\varepsilon_1}\partial_y l_{11}(x,y) = (\lambda_1(y) + c_1) k_{11}(x,y) \tag{27}$$
$$\sqrt{\varepsilon_1}\partial_x l_{12}(x,y) - \sqrt{\varepsilon_2}\partial_y l_{12}(x,y) = (\lambda_2(y) + c_1) k_{12}(x,y) \tag{28}$$
$$\sqrt{\varepsilon_2}\partial_x l_{21}(x,y) - \sqrt{\varepsilon_1}\partial_y l_{21}(x,y) = (\lambda_1(y) + c_2) k_{21}(x,y) - g(x) k_{11}(x,y) \tag{29}$$
$$\sqrt{\varepsilon_2}\partial_x l_{22}(x,y) - \sqrt{\varepsilon_2}\partial_y l_{22}(x,y) = (\lambda_2(y) + c_2) k_{22}(x,y) - g(x) k_{12}(x,y) \tag{30}$$

subject to the following boundary conditions:

$$l_{11}(x,x) = -\frac{\lambda_1(x) + c_1}{2\sqrt{\varepsilon_1}} \tag{31}$$
$$k_{11}(x,0) = \frac{a\varepsilon_2}{\varepsilon_1(a\varepsilon_2 + \sqrt{\varepsilon_1\varepsilon_2})} \times \int_0^x \sqrt{\varepsilon_1} l_{11}(y,0) + \sqrt{\varepsilon_2} l_{12}(y,0) dy \tag{32}$$
$$l_{12}(x,x) = 0 \tag{33}$$
$$k_{12}(x,x) = 0 \tag{34}$$
$$k_{12}(x,0) = \frac{1}{a\varepsilon_2 + \sqrt{\varepsilon_1\varepsilon_2}} \times \int_0^x \sqrt{\varepsilon_1} l_{11}(y,0) + \sqrt{\varepsilon_2} l_{12}(y,0) dy \tag{35}$$
$$l_{21}(x,x) = (\sqrt{\varepsilon_1} + \sqrt{\varepsilon_2})^{-1} \Big((\varepsilon_2 - \sqrt{\varepsilon_1\varepsilon_2}) \frac{\lambda_1(x) + c_1}{2\varepsilon_1(\varepsilon_2 - \varepsilon_1)} + (\varepsilon_1 - \varepsilon_2) k_{11}(x,x) k_{21}(x,x) \Big) \tag{36}$$
$$k_{21}(x,0) = \frac{a\varepsilon_2}{\varepsilon_1} \frac{\lambda_2'(0)}{\lambda_2(0) + c_2} + \left(\sqrt{\varepsilon_1\varepsilon_2} + \frac{\varepsilon_1}{a}\right)^{-1} \times \int_0^x \sqrt{\varepsilon_1} l_{21}(y,0) + \sqrt{\varepsilon_2} l_{22}(y,0) dy \tag{37}$$
$$l_{22}(x,x) = -\frac{\lambda_2(x) + c_2}{2\sqrt{\varepsilon_2}} \tag{38}$$

$$k_{22}(x,0) = \frac{\lambda_2'(0)}{\lambda_2(0)+c_2} + \frac{\varepsilon_1}{a\varepsilon_2}\left(\sqrt{\varepsilon_1\varepsilon_2} + \frac{\varepsilon_1}{a}\right)^{-1} \times \int_0^x \sqrt{\varepsilon_1} l_{21}(y,0) + \sqrt{\varepsilon_2} l_{22}(y,0)dy \quad (39)$$

where g is a function that is related to the gain kernel component k_{21}:

$$g(x) = (\varepsilon_2 - \varepsilon_1)k_{21}(x,x). \quad (40)$$

These kernel equations are very similar to those found in Vazquez and Krstic (2017b), albeit with a mirrored folding condition in arising in (32),(35),(37),(39). The way the folding condition arises will be elucidated in Section 3.2.

Following this, an intermediate target system is imposed:

$$\partial_t v_1 = \varepsilon_1 \partial_x^2 v_1 - c_1 v_1 \quad (41)$$
$$\partial_t v_2 = \varepsilon_2 \partial_x^2 v_2 - c_2 v_2 + g(x)\partial_x v_1 \quad (42)$$
$$\partial_x v_1(0,t) = -a\partial_x v_2(0,t) \quad (43)$$
$$v_1(0,t) = v_2(0,t) \quad (44)$$
$$v_1(1,t) = 0 \quad (45)$$
$$v_2(1,t) = \mathcal{V}_2(t) \quad (46)$$

with $c_1, c_2 > 0$ being designed coefficients. Furthermore, $\mathcal{V}_2$ is an auxiliary controller that is to be designed with another backstepping transformation, and is related to the original controller $\mathcal{U}_2$:

$$\mathcal{V}_2(t) = \mathcal{U}_2(t) - \int_0^1 [k_{21}(1,y)u_1(y,t) + k_{22}(1,y)u_2(y,t)]dy \quad (47)$$

The second transformation takes the intermediate target system and transforms it into a stable target system. Note that it is a scalar transformation on *one* equation.

$$w_2(x,t) = v_2(x,t) - \int_0^x p(x,y)v_2(y,t)dy - \int_0^x q(x,y)v_1(y,t)dy \quad (48)$$

p, q are kernels that satisfy a coupled second-order hyperbolic PDE system. Much like the case with the k_{ij} kernels, we define two functions $r(x,y), s(x,y)$ to transform the gain kernels (p,q) from a 2×2 second-order hyperbolic PDEs into a 4×4 set of first-order hyperbolic PDEs:

$$r(x,y) = \partial_x p(x,y) + \partial_y p(x,y) \quad (49)$$
$$s(x,y) = \sqrt{\varepsilon_2}\partial_x q(x,y) + \sqrt{\varepsilon_1}\partial_y q(x,y) \quad (50)$$

Where the kernels p, q, r, s satisfy the following set of PDEs:

$$\partial_x p(x,y) + \partial_y p(x,y) = r(x,y) \quad (51)$$
$$\partial_x r(x,y) - \partial_y r(x,y) = 0 \quad (52)$$
$$\sqrt{\varepsilon_2}\partial_x q(x,y) + \sqrt{\varepsilon_1}\partial_y q(x,y) = s(x,y) \quad (53)$$
$$\sqrt{\varepsilon_2}\partial_x s(x,y) - \sqrt{\varepsilon_1}\partial_y s(x,y) = (c_1 - c_2)q(x,y) + \partial_y p(x,y)g(y) - p(x,y)g'(y) \quad (54)$$

subject to the following boundary conditions:

$$p(x,0) = a\frac{\varepsilon_1}{\varepsilon_2}\left(\frac{g(0)}{\varepsilon_1 - \varepsilon_2} - \int_0^x \frac{\varepsilon_2 r(y,0) + \sqrt{\varepsilon_1}s(y,0) - g(0)}{a\varepsilon_1 + \sqrt{\varepsilon_1\varepsilon_2}}\right)dy \quad (55)$$
$$r(x,x) = 0 \quad (56)$$
$$q(x,0) = \frac{g(0)}{\varepsilon_1 - \varepsilon_2} - \int_0^x \frac{\varepsilon_2 r(y,0) + \sqrt{\varepsilon_1}s(y,0) - g(0)}{a\varepsilon_1 + \sqrt{\varepsilon_1\varepsilon_2}}dy \quad (57)$$
$$s(x,x) = (\sqrt{\varepsilon_1} + \sqrt{\varepsilon_2})^{-1}\left(g(x) + (\sqrt{\varepsilon_1\varepsilon_2} - \varepsilon_2)\frac{1}{\varepsilon_1 - \varepsilon_2}g'(x)\right) \quad (58)$$

These transformations will admit the following target system:

$$\partial_t v_1(x,t) = \varepsilon_1\partial_x^2 v_1(x,t) - c_1 v_1(x,t) \quad (59)$$
$$\partial_t w_2(x,t) = \varepsilon_2\partial_x^2 w_2(x.t) - c_2 w_2(x,t) \quad (60)$$
$$\partial_x v_1(0,t) = -a\partial_x w_2(0,t) \quad (61)$$
$$v_1(0,t) = w_2(0,t) \quad (62)$$
$$v_1(1,t) = 0 \quad (63)$$
$$w_2(1,t) = 0 \quad (64)$$

3.1 Lyapunov stability

Lemma 2. The target system (59)-(64) is exponentially stable in the L^2-sense, that is, there exist constants $M_1, \gamma_1 > 0$ such that

$$||(v_1, w_2)||_{L^2}(t) \leq M_1 e^{-\gamma_1 t}||(v_1, w_2)||_{L^2}(0), \forall t \geq 0 \quad (65)$$

We omit the proof but give the following Lyapunov function.

$$V(t) = \frac{1}{2}\int_0^1 v_1(x,t)^2 + \frac{a\varepsilon_1}{\varepsilon_2}w_2(x,t)^2 dx \quad (66)$$

Note that $a\varepsilon_1\varepsilon_2^{-1} > 1$ by definition and Assumption 1. The constants M_1, γ_1 can be found:

$$M_1 = \sqrt{\frac{a\varepsilon_1}{\varepsilon_2}} \quad (67)$$
$$\gamma_1 = \min\left\{c_1 + \frac{\varepsilon_1}{4}, \frac{\varepsilon_2}{a\varepsilon_1}\left(c_2 + \frac{\varepsilon_2}{4}\right)\right\} \quad (68)$$

We will now state the contribution of our paper below:

Theorem 3. Consider the system (1)-(3). The pair of feedback controllers

$$\mathcal{U}_1(t) = \int_{-1}^1 F_1(y)u(y,t)dy \quad (69)$$
$$\mathcal{U}_2(t) = \int_{-1}^1 F_2(y)u(y,t)dy \quad (70)$$

where the gain functions F_1, F_2 are defined by

$$F_1(y) = \begin{cases} (1+y_0)^{-1}k_{11}\left(1, \frac{y_0 - y}{1+y_0}\right) & y \leq y_0 \\ (1-y_0)^{-1}k_{12}\left(1, \frac{y - y_0}{1-y_0}\right) & y > y_0 \end{cases} \quad (71)$$

$$F_2(y) = \begin{cases} (1+y_0)^{-1}h_1\left(\frac{y_0 - y}{1+y_0}\right) & y \leq y_0 \\ (1-y_0)^{-1}h_2\left(\frac{y - y_0}{1-y_0}\right) & y > y_0 \end{cases} \quad (72)$$

$$h_1(y) = k_{21}(1,y) + q(1,y) - \int_y^1 \big[p(1,z)k_{21}(z,y) + q(1,z)k_{11}(z,y)\big]dz \quad (73)$$

$$h_2(y) = k_{22}(1,y) + p(1,y) - \int_y^1 \big[p(1,z)k_{22}(z,y)$$

$$+ q(1,z)k_{12}(z,y)\big]dz \tag{74}$$

where k_{ij}, p, q are kernels which satisfy (23)-(30),(51)-(54). With a choice of the design parameter $y_0 \in (-1,1)$, the controller pair $\mathcal{U}_1, \mathcal{U}_2$ will exponentially stabilize the trivial solution $u \equiv 0$ in the L^2-sense, that is, there exist constants $M, \gamma > 0$ such that

$$||u(\cdot,t)||_{L^2} \le Me^{-\gamma t}\,||u(\cdot,0)||_{L^2}, \forall t \ge 0 \tag{75}$$

The proof of the theorem is relatively straightforward, and utilizes the invertibility of the Volterra transformations (21),(48) to establish an equivalence relation between the target system and the original system under feedback. The Lyapunov function (66) can be shown to be equivalent to a transformed Lyapunov function establishing stability in the original system with the nonlocal stabilizing feedback law.

3.2 Kernel equation derivation for K

Differentiating (21) once in time and twice in space and then imposing (41)-(46), one can arrive at the following set of conditions comprising a PDE for K:

$$E\partial_x^2 K(x,y) + \partial_y^2 K(x,y)E = K(x,y)\Lambda(y) + CK(x,y) - G(x)K(x,y) \tag{76}$$

$$\partial_y K(x,x)E + E\partial_x K(x,x) = -E\frac{d}{dx}K(x,x) - \Lambda(x) - C - G(x)K(x,x) \tag{77}$$

$$EK(x,x) - K(x,x)E = G(x) \tag{78}$$

$$K(x,0)E\partial_x U(0) = \partial_y K(x,0)EU(0) \tag{79}$$

where C and $G(x)$ are defined as follows for compact matrix notation:

$$C := \operatorname{diag}(c_1, c_2) \tag{80}$$

$$G(x) := \begin{pmatrix} 0 & 0 \\ g(x) & 0 \end{pmatrix} \tag{81}$$

One may see that (79) is quite curious and unusual for most backstepping designs. (79) comes about due to the exotic folding condition (8). Surprisingly enough, looking at the conditions componentwise will reveal that (79) exactly consitutes folding conditions on K. The folding conditions that arise from (79) are:

$$\varepsilon_1 k_{11}(x,0) - a\varepsilon_2 k_{12}(x,0) = 0 \tag{82}$$

$$\varepsilon_1 \partial_y k_{11}(x,0) + \varepsilon_2 \partial_y k_{12}(x,0) = 0 \tag{83}$$

$$\varepsilon_1 k_{21}(x,0) - a\varepsilon_2 k_{22}(x,0) = 0 \tag{84}$$

$$\varepsilon_1 \partial_y k_{21}(x,0) + \varepsilon_2 \partial_y k_{22}(x,0) = 0 \tag{85}$$

Then, applying (22) to the 2×2 second-order hyperbolic matrix PDE (76) will transform the problem into the $2 \times 2 \times 2$ first-order hyperbolic PDEs given by (23)-(30).

A discussion of well-posedness of k_{ij}, l_{ij} The method of solving the kernel equations is very similar to the approach found in Vazquez and Krstic (2017b). However, due to the differing boundary conditions, some care must be taken. The initial step is identical – one can establish integral operator equations:

$$\begin{pmatrix} k_{11} \\ k_{12} \end{pmatrix} = H_1\left[\begin{pmatrix} l_{11} \\ l_{12} \end{pmatrix}\right] \tag{86}$$

$$\begin{pmatrix} l_{11} \\ l_{12} \end{pmatrix} = H_2\left[\begin{pmatrix} k_{11} \\ k_{12} \end{pmatrix}\right]. \tag{87}$$

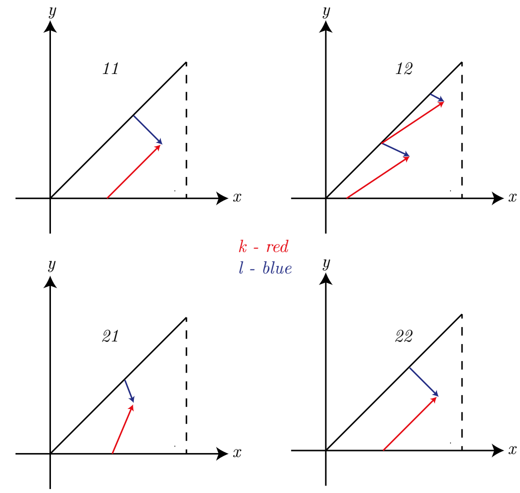

Fig. 1. Kernel characteristics for k, l equations. In particular, note 12 and 21 cases, and which boundary conditions are needed.

One can directly apply the method of successive approximations to obtain a solution $k_{11}, k_{12}, l_{11}, l_{12}$. This process is repeated for $k_{21}, k_{22}, l_{21}, l_{22}$.

$$\begin{pmatrix} k_{21} \\ k_{22} \end{pmatrix} = H_3\left[\begin{pmatrix} l_{21} \\ l_{22} \end{pmatrix}\right] \tag{88}$$

$$\begin{pmatrix} l_{21} \\ l_{22} \end{pmatrix} = H_4\left[\begin{pmatrix} k_{21} \\ k_{22} \end{pmatrix}\right] \tag{89}$$

Since the integral operators found in $H_i, i \in \{1,2,3,4\}$ are all of Volterra type with L^2 kernels, by Section 1.5 in Tricomi (1985), one can conclude that a unique L^2 solution exists for these integral equations.

3.3 Kernel derivation for p, q

We differentiate (48) once in time, and twice in space, and then impose (59)-(64). We can recover the following coupled hyperbolic kernel PDEs for p, q:

$$\partial_x^2 p(x,y) - \partial_y^2 p(x,y) = 0 \tag{90}$$

$$\varepsilon_2 \partial_x^2 q(x,y) - \varepsilon_1 \partial_y^2 q(x,y) = (c_2 - c_1)q(x,y) + \partial_y p(x,y)g(y) - p(x,y)g'(y) \tag{91}$$

$$\frac{d}{dx}p(x,x) = 0 \tag{92}$$

$$q(x,x) = \frac{1}{\varepsilon_1 - \varepsilon_2} g(x) \tag{93}$$

$$(\varepsilon_1 - \varepsilon_2)\partial_y q(x,x) + \frac{2\varepsilon_2}{\varepsilon_1 - \varepsilon_2} g'(x) = g(x) \tag{94}$$

$$-\varepsilon_2 \partial_y p(x,0) + g(0) - \varepsilon_1 \partial_y q(x,0) = 0 \tag{95}$$

$$p(x,0) = a\frac{\varepsilon_1}{\varepsilon_2} q(x,0) \tag{96}$$

Again, applying (49),(50) will admit the gain kernels (51)-(54).

A discussion of well-posedness for p, q, r, s It is not immediately obvious that the system p, q, r, s is even a well-posed system. One can apply method of characteristcs selectively, and prune the system into one where the well-posedness is more straightforward to prove. Straight away, one can note that (52),(56) will trivially define the solution $r(x, y) \equiv 0$.

(51) will then be trivially solvable via method of characteristics, with $p(x,y) = p(x-y,0)$. One must note that $p(x,0)$ is an *integral* operator on $s(x,0)$, which is critical for establishing the well-posedness of the reduced (q,s) kernel system.

(54) is the primary point of contention, as it involves a term of the form $\partial_y p(x,y)$. However, noting our observation above,

$$\partial_y p(x,y) = \partial_y p(x-y,0) = \frac{\sqrt{\varepsilon_1}s(x-y,0) - g(0)}{a\varepsilon_1 + \sqrt{\varepsilon_1\varepsilon_2}} \quad (97)$$

By using this representation for $\partial_y p(x,y)$ in (54), it is quite clear to see that (q,s) becomes a coupled hyperbolic PDE system with a nonlocal term (being $s(x-y,0)$) in the right-hand side. However, this nonlocality does not affect the well-posedness of the system, which becomes more clear if one directly applies the method of characteristics.

4. SIMULATION STUDIES

A natural question arises with the design parameter y_0. The choice of the folding point intuitively will affect the distribution of control effort between $\mathcal{U}_1$ and $\mathcal{U}_2$. Since explicit solutions to the gain kernels $(23)-(30)$, $(51)-(54)$ are difficult (and probably impossible) to find, we turn to numerical studies to gain a basic understanding of how the choice of y_0 affects the controllers.

Recalling Assumption 1, we select a folding point $y_0 = -0.001$ to observe the case in which the folding point is close to the center, and another folding point at $y_0 = -0.3$ which is further inside the interval $(-1,0)$.

Parameter	Value
ε	1
$\lambda(x)$	$5e^{2x}$
y_0	$-0.001, -0.3$
c_1, c_2	2

Table 1. Simulation parameters

The reaction coefficient $\lambda(x)$ is chosen to be $\lambda(x) = 5e^{2x}$, which is heavily biased for $x \in (0,1)$. This in particular helps illustrate the effectiveness of choosing a non-midway folding point. Since the instability is nonuniform, the choice of the folding point can help mitigate the unbalanced control effort, or in certain operating scenarios, shift the control effort from one controller the other.

The closed-loop responses show the effect of choosing the folding point on the state response. In particular, the symmetry of the solution is heavily affected, as one may naturally be inclinded to believe. In Figure 3, the folding point is close to the midpoint, and therefore the response looks roughly symmetric. However, this is contrasted with Figure 4 (the choice of $y_0 = -0.3$, further from the midpoint), where the response is quicker in $x \in (-1,0)$ than in $x \in (0,1)$.

Finally, the control effort exerted by each controller reveals more. By shifting the folding point further from the spatial midpoint, we can see that the control effort exerted by each controller can change drastically. In particular, observing Figure 5, when the folding point is chosen to be close to the midpoint, the control effort exerted by either controller is roughly the same. However, by biasing the controllers via selecting the folding point further away, the control effort of $\mathcal{U}_1$ diminishes, while the control effort of $\mathcal{U}_2$ increases. However, from these plots, the overall control

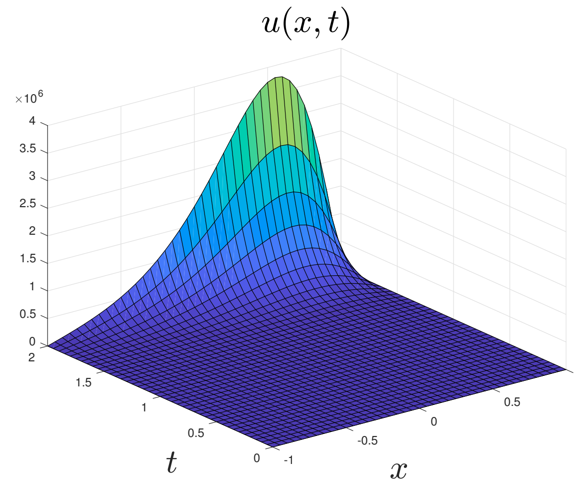

Fig. 2. Open-loop response $u(x,t)$ of the system. Note that the biasing of λ to half the domain is noticeble in the growth.

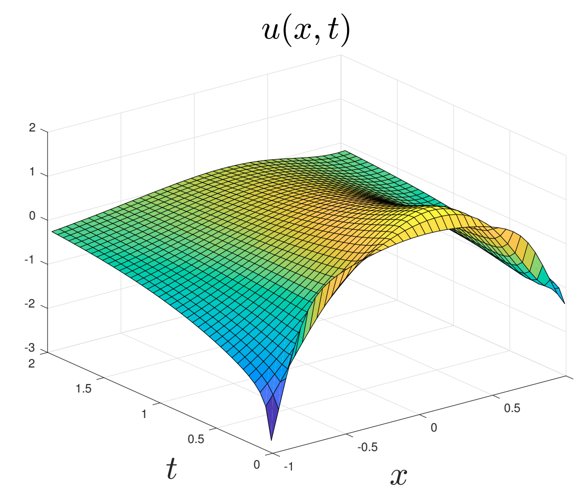

Fig. 3. Closed-loop response $u(x,t)$ with the folding point chosen to be $y_0 = -0.001$. The folding point is quite close to the midpoint of the spatial domain, and the response is close to symmetric.

Folding point y_0	Ψ_1	Ψ_2	$\Psi_1 + \Psi_2$
-0.001	0.1459	0.0854	0.2313
-0.3	0.0372	0.0643	0.1015

Table 2. Energy Ψ_i (L_2) consumed by control effort for varying folding point choices

effort appears to increase. To study this, we define the controller energy using an L_2 norm:

$$\Psi_i = \int_0^T |\mathcal{U}_i(t)|^2 dt \quad (98)$$

where T is the total simulation length. It turns out instead that the energy expended by either controller is in fact lower as the folding point is moved further away from the midpoint, as one may notice from Table 2.

It is interesting to note that depending on application, one design may be superior than another. With one choice, the l_2 signal norm of the controller is smaller (lower energy),

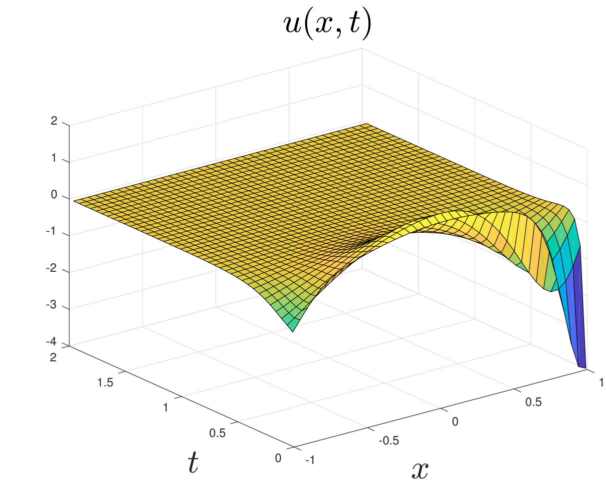

Fig. 4. Closed-loop response $u(x,t)$ with the folding point chosen to be $y_0 = -0.3$. The folding point is further from the midpoint of the spatial domain. One half of the domain converges quicker than the other.

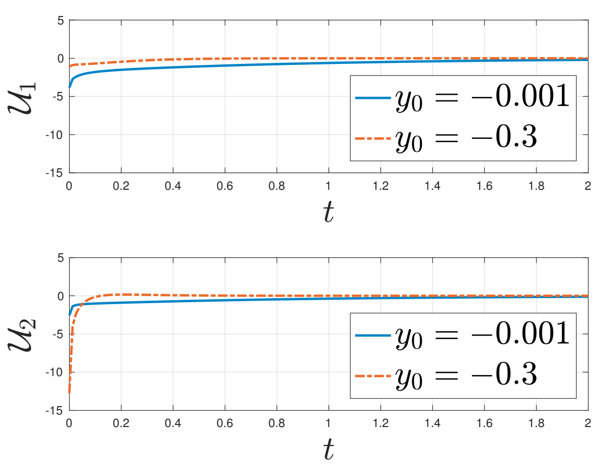

Fig. 5. Control effort exerted by controllers for both choices of folding point. Notice how as the folding point y_0 is shifted further into the negative half of the domain, the control effort exerted by $\mathcal{U}_2$ grows while $\mathcal{U}_1$ shrinks.

but pays the price of high peaking. The opposite case is certainly true as well – a choice that lowers the peaking (l_∞ signal norm) will reduce the peaking, but at the cost of energy.

5. CONCLUSION

A method for designing bilateral backstepping feedback controllers for an unstable parabolic equation is presented. The unstable diffusion-reaction PDE is transformed via folding to a system of coupled parabolic PDEs with an exotic boundary condition arising from imposing second-order compatibility conditions. The controllers are designed for the coupled PDE system through the use of two consecutive backstepping transformations. The controllers possess a design parameter in the choice of the folding point, which affects the response of both controllers. By choosing this design parameter, the designer can shift control effort from one controller to another, which is beneficial from a fault-tolerant perspective, as well as in efficiency.

This problem raises an interesting perspective from the dual problem in state estimation. In particular, if one follows the intuition behind the dual problem, it is possible that having a pair of pointwise sensors within the interior of the domain of the PDE may be sufficient to fully observe the system state. From an application standpoint, this is highly desirable, as it involves potentially two low-cost or otherwise simple sensors in the interior. For example, in chemical reactors, measuring concentration and flux at any arbitrary point within the domain could admit an accurate estimate of the state.

Ongoing and future work includes the dual problem in state estimation, output feedback, and applying extendable framework of results in mixed-type PDEs to solve the problem of bilateral boundary control with distinct input delays.

REFERENCES

Auriol, J. and Meglio, F.D. (2017). Two sided boundary stabilization of heterodirectional linear coupled hyperbolic pdes. *IEEE Transactions on Automatic Control*, PP(99), 1–1. doi:10.1109/TAC.2017.2763320.

Bekiaris-Liberis, N. and Vazquez, R. (2018). Nonlinear bilateral full-state feedback trajectory tracking for a class of viscous hamilton-jacobi pdes. In *2018 IEEE Conference on Decision and Control (CDC)*, 515–520. IEEE.

Deutscher, J. and Kerschbaum, S. (2018). Backstepping control of coupled linear parabolic pides with spatially varying coefficients. *IEEE Transactions on Automatic Control*, 63(12), 4218–4233.

Krstic, M. and Smyshlyaev, A. (2008). *Boundary Control of PDEs: A Course on Backstepping Designs*. SIAM.

Orlov, Y., Pisano, A., Pilloni, A., and Usai, E. (2017). Output feedback stabilization of coupled reaction-diffusion processes with constant parameters. *SIAM Journal on Control and Optimization*, 55(6), 4112–4155. doi:10.1137/15M1034325. URL `https://doi.org/10.1137/15M1034325`.

Tricomi, F.G. (1985). *Integral equations*, volume 5. Courier Corporation.

Tsubakino, D., Krstic, M., and Yamashita, Y. (2013). Boundary control of a cascade of two parabolic pdes with different diffusion coefficients. In *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, 3720–3725. IEEE.

Vazquez, R. and Krstic, M. (2016). Bilateral boundary control of one-dimensional first- and second-order pdes using infinite-dimensional backstepping. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, 537–542. doi:10.1109/CDC.2016.7798324.

Vazquez, R. and Krstic, M. (2017a). Boundary control and estimation of reaction–diffusion equations on the sphere under revolution symmetry conditions. *International Journal of Control*, 1–10.

Vazquez, R. and Krstic, M. (2017b). Boundary control of coupled reaction-advection-diffusion systems with spatially-varying coefficients. *IEEE Transactions on Automatic Control*, 62(4), 2026–2033. doi: 10.1109/TAC.2016.2590506.