Bilateral backstepping control of coupled linear parabolic PDEs with spatially varying coefficients

Simon Kerschbaum \textsuperscript{a} and Joachim Deutscher \textsuperscript{b}

\textsuperscript{a}Lehrstuhl für Regelungstechnik, Universität Erlangen-Nürnberg, Cauerstraße 7, D-91058 Erlangen, Germany
\textsuperscript{b}Institut für Mess-, Regel- und Mikrotechnik, Universität Ulm, Albert-Einstein-Allee 41, D-89081 Ulm, Germany

Abstract

This paper considers the backstepping state feedback control of coupled linear parabolic PDEs with spatially varying coefficients and bilateral actuation. By making use of the folding technique, a system representation with unilateral actuation is obtained, allowing to apply the standard backstepping transformation. To ensure the regularity of the solution, the folded system is subject to unusual folding boundary conditions, which lead to additional boundary couplings between the PDEs. Therefore, the solution of the corresponding kernel equations determining the transformations is a very challenging problem. A systematic approach to derive the corresponding integral equations is proposed, allowing to solve them with the method of successive approximations. By making use of a Volterra and a Volterra-Fredholm transformation, the closed-loop system is mapped into a cascade of stable parabolic systems. This allows a simple proof of exponential stability in the $L^2$-norm with the decay rate as design parameter. The bilateral state feedback stabilization of an unstable system of two coupled parabolic PDEs and the comparison to the application of an unilateral controller demonstrates the results of the paper.

Key words: Distributed-parameter systems, parabolic systems, bilateral control, backstepping, boundary control.

1 Introduction

1.1 Background and motivation

In the last decades, the backstepping method has emerged as a powerful tool for the boundary control of distributed parameter systems (for an overview, see [16,21]). In the scope of parabolic systems, current research is focused on the control of coupled PDEs, where results exist for both constant [2,17] and spatially varying coefficients [20,8] as well as for space and time dependent parameters [15]. This system class is of great interest for applications, modelling technological processes in chemical and biochemical engineering (see [1,13]) and crystallisation processes (see [6]). Most existing results for the control of parabolic systems consider the unilateral case, which means that the inputs of the system are all located at one boundary.

In practice, however, there may be the possibility to place actuators at all boundaries of the domain, achieving bilateral actuation. Using this additional degree of freedom in the controller design, it is to be expected that the control effort at one input for achieving comparable results as in the unilateral case, can be significantly reduced. Furthermore, the additional actuation has shown to be useful in the deployment of agents, when modelling multi-agent systems by continuum models (see [12]). In addition, bilateral control of traffic flow models is considered in [3].

In [22], the backstepping method was used for the controller and observer design of parabolic PDEs on balls of arbitrary dimension with actuation on the whole boundary. Since in one dimension, the ball actually is an interval, this included bilateral control of 1D parabolic systems. Therefore, [18] focused on this, included further system classes and described two basic concepts to deal with bilateral control. The first is to adjust the applied integral transformation leading to a symmetric backstepping transformation, which was applied in [22]. This approach is also utilized in [3] for the bilateral stabilization and observer design of a viscous Hamilton-Jacobi PDE. By the help of a Hopf-Cole transformation, this semilinear parabolic system is mapped into a linear representation so that the results of [22] can be applied. The other concept proposed in [18] is to reformulate the problem by suitable transformations to achieve unilateral actuation. To this end, the spatial domain is folded at a folding point inside the domain, with the aim to get a system representation which is similar to
the classic unilateral problem so that existing results can be utilized. This is applied in [11] for the observer design of a parabolic PDE with constant coefficients, where measurements are available at both boundaries and in-domain, which allows to design two independent observers for the different parts of the domain. For hyperbolic $2 \times 2$ systems with spatially varying coefficients, [23] considers the minimum time observer design.

As far as the controller design of parabolic systems is concerned, the folding approach leads to new challenges. In particular, folding a scalar parabolic system leads to a $2 \times 2$ system coupled via special folding boundary conditions (BCs), which need to be introduced to ensure the regularity of the solution. In [18] the folding point is chosen to be the centre of the spatial domain. With constant parameters, this leads to equal diffusion coefficients in the folded system. This was extended by [4,5] to the case where the folding point can be chosen arbitrarily as a design parameter to freely distribute the control effort between the available boundary inputs. However, this leads to different diffusion coefficients in the folded system, requiring the introduction of coupling terms, the so-called well-posedness terms, in the target system to ensure the well-posedness of the kernel equations. This challenge is tackled by introducing a second integral transformation of Fredholm-type to eliminate the couplings and to obtain a target system, whose stability is easy to show.

So far, however, there exist no results for the bilateral control of coupled parabolic PDEs with spatially varying coefficients and distinct diffusion coefficients. For this setup with $n$ states, the folding approach leads to the difficult problem to stabilize a set of $2n$ coupled parabolic PDEs subject to folding BCs.

1.2 Contribution

In this paper, a stabilizing state feedback controller is designed for linear coupled parabolic PDEs with bilateral actuation where all coefficients are spatially varying and the diffusion coefficients are distinct.

The design procedure is based on the folding approach, leading to a system representation with unilateral actuation. Hence, the usual backstepping transformation is applicable to determine the controller. The corresponding kernel equations, however, attain an additional coupling due to the folding BCs, which leads to new problems for their solution. This becomes very challenging for coupled parabolic PDEs.

In contrast to the existing results in [4,5], the solution of the kernel equations is based on the approach in [8,15]. In particular, integral equations are directly derived for the second-order kernel equations by mapping the appearing wave equation operator into its canonical form. Then, integral equations result from formal integrations. It is shown that despite the folding BCs, the kernel equations can be converted into proper integral equations by extending the method presented in [8] to the considered case of bilateral actuation. This requires new tricky ideas to setup the integral equations, in order to take the folding BCs into account. The resulting integral equations then have similar terms as in [8], and can thus be solved by the method of successive approximations.

Since the stability analysis of the target system resulting from the first backstepping transformation of the backstepping transformation is hindered by the coupling BCs, a Volterra-Fredholm transformation is applied to decouple the original target system into a cascade of stable parabolic systems, allowing a simple proof of stability relying on existing results. This allows the explicit specification of the stability margin for the closed-loop system. This transformation is the generalization of the approach in [5]. There, the Volterra part of the transformation can be significantly simplified because it involves only one state. In the general case of $n$ coupled states, the transformation leads to a novel, coupled set of boundary value problems (BVP) for the Volterra and Fredholm kernels of the decoupling transformation. With a similar approach as for the backstepping kernel, this BVP can be converted into integral equations and solved by the method of successive approximations. Hence, the paper provides a systematic approach to deal with the bilateral backstepping control problem for a large class of coupled parabolic PDEs.

1.3 Organisation and notation

The considered bilateral stabilization problem is introduced in the next section. Section 3 presents the design of the state feedback controller. Therein, the folding transformation is followed by the backstepping transformation and the decoupling transformation along with the resulting state feedback controller and the analysis of the closed-loop stability. In Section 4, the solution of the kernel equations for the transformations into the target system is determined by converting them into proper integral equations and applying a successive approximation. The comparison of the bilateral controller with an unilateral controller for an unstable system of two coupled parabolic PDEs confirms the effectiveness of the proposed design method in Section 5.

Notation: In the paper, the notations $f_\varepsilon(z,z) = f_\varepsilon(z,\zeta)|_{\zeta=z}, f_\zeta(z,z) = f_\zeta(z,\zeta)|_{\zeta=z}$ and $\lfloor \cdot \rfloor$, simplify the presentation of the results. The latter means that the expression in the square brackets has to be considered only if the condition $\ast$ is fulfilled. Finally, variables with index $\cdot_i$ or $\cdot_{ij}$ are represented without index but boldface, e.g., $K = K_{ij}$ if convenient.

2 Problem formulation

Consider the system

$$w_\varepsilon(\hat{z}, t) = \hat{A}(\hat{z})w_{z\hat{z}}(\hat{z}, t) + \hat{A}(\hat{z})w(\hat{z}, t) \quad (1a)$$
$$w_\varepsilon(0, t) = B_0w_0(0, t) + u_0(t), \quad t > 0 \quad (1b)$$
$$w_\varepsilon(1, t) = B_1w_1(1, t) + u_1(t), \quad t > 0 \quad (1c)$$

consisting of $n$ coupled linear parabolic PDEs (1a) for the state $w(\hat{z}, t) \in \mathbb{R}^n$, $n \geq 1$, defined on $(\hat{z}, t) \in (0, 1) \times \mathbb{R}^+$ with the initial condition (IC) $w(\hat{z}, 0) = w_0(\hat{z})$. The boundary conditions (BCs)
(1b), (1c) with $B_0, B_1 \in \mathbb{R}^{n \times n}$ contain the inputs $u_0(t), u_1(t) \in \mathbb{R}^n$. The matrix $\Lambda(\hat{z})$ is considered as a diffusion matrix and assumed to have a diagonal structure, i.e., $\Lambda(\hat{z}) = \text{diag}(\lambda_1(\hat{z}), \ldots, \lambda_n(\hat{z}))$ with the diffusion coefficients $\lambda_i \in C^2[0,1], i = 1, \ldots, n$. For a simpler presentation, they are assumed to be distinct and sorted, i.e., $\lambda \geq \lambda_1(\hat{z}) > \ldots > \lambda_n(\hat{z}) \geq \lambda$, $\hat{z} \in [0,1]$, with positive constants $\lambda$ and $\lambda$. The reaction matrix $A = [A_{ij}] \in (C[0,1])^{n \times n}$ describes the coupling between the states and is the main source of possible instability.

The system (1) may result from the more general PDE

$w_t(z, t) = (\Lambda(z)w_t(z, t)) + \Phi(z)w_t(z, t) + A(z)w_t(z, t)$

with a diagonal advection matrix $\Phi \in (C[0,1])^{n \times n}$ after a Hopf-Cole-transformation (see [8]).

In this paper, a static state feedback controller

$$\begin{bmatrix} u_0(t) \\ u_1(t) \end{bmatrix} = K[w(t)]$$

(2)

with the formal feedback operator $K$ is determined by making use of the backstepping method. This controller has to ensure that the closed-loop system is exponentially stable with an prescribed rate of convergence.

3 State feedback controller design

3.1 Folding transformation

As suggested by [18,4,5], a folding transformation is applied to get a system representation with one-sided actuation to be able to apply backstepping with the usual Volterra-type integral transformation. The spatial domain of the system is folded at the folding point $\hat{z}_0 \in (0,1)$, which is a design parameter. This leads to the new spatial domain $z \in [0,1]$ with

$$z = \hat{f}(\hat{z}) = \begin{cases} (\hat{z}_0 - \hat{z})/\hat{z}_0, & \hat{z} < \hat{z}_0 \\ (\hat{z} - \hat{z}_0)/(1 - \hat{z}_0), & \hat{z} \geq \hat{z}_0 \end{cases}$$

(3)

and the new state $x(z, t) = \text{col}(x^l(z, t), x^r(z, t)) \in \mathbb{R}^{2n}$, describing the left part $x^l(z, t) \in \mathbb{R}^n$ and the right part $x^r(z, t) \in \mathbb{R}^n$ w.r.t. $\hat{z}_0$, where

$$x(z, t) = \begin{bmatrix} x_1(z, t) \\ \vdots \\ x_n(z, t) \\ x_{n+1}(z, t) \\ \vdots \\ x_{2n}(z, t) \end{bmatrix} = \begin{bmatrix} w_1(\hat{z}_0 - \hat{z}_0 z, t) \\ \vdots \\ w_n(\hat{z}_0 - \hat{z}_0 z, t) \\ w_1(\hat{z}_0 + (1 - \hat{z}_0) z, t) \\ \vdots \\ w_n(\hat{z}_0 + (1 - \hat{z}_0) z, t) \end{bmatrix}.$$ (4)

The dynamics of the folded state reads

$$\begin{aligned}
x_1(z, t) &= \Lambda(z)x_1(z, t) + A(z)x(z, t) \\
x^r(0, t) &= x^l(0, t) \quad (5a) \\
x^l_1(z, t) &= -\frac{\hat{z}_0}{1 - \hat{z}_0} x^r_1(0, t) \quad (5b) \\
x^l(z, t) &= \begin{bmatrix} -\hat{z}_0 (B_0 w(0, t) + u_0(t)) \\ (1 - \hat{z}_0) (B_1 w(1, t) + u_1(t)) \end{bmatrix}, \\
&= \begin{bmatrix} \hat{A}(\hat{z}_0 - \hat{z}_0 z) \\ \hat{A}(\hat{z}_0 + (1 - \hat{z}_0) z) \end{bmatrix} \end{aligned}$$

(5d)

in which $\Lambda(z) = \text{bdiag}([\lambda_1(z), \ldots, \lambda_{2n}(z)])$ with

$$\lambda_i(z) = \begin{cases} \hat{\lambda}_i(\hat{z}_0 - \hat{z}_0 z)/\hat{z}_0^2, & i \leq n \\ \hat{\lambda}_i(\hat{z}_0 + (1 - \hat{z}_0) z)/(1 - \hat{z}_0)^2, & i > n \end{cases}.$$ (5e)

$A(z) = [A_{ij}(z)] = \text{bdiag}([A_l(z), A_r(z)]) \in \mathbb{R}^{2n \times 2n}$, $A_l(z), A_r(z) \in \mathbb{R}^{n \times n}$ and

$$\begin{aligned}
A_l(z) &= \hat{A}(\hat{z}_0 - \hat{z}_0 z) \\
A_r(z) &= \hat{A}(\hat{z}_0 + (1 - \hat{z}_0) z) \end{aligned}$$

(5f)

results from inserting $\hat{z} = f^{-1}(z)$ (see (3)) in $\hat{A}(\hat{z})$ and $\hat{A}(\hat{z})$. Obviously, the folded system (5) is subject to the usual unilateral control but contains the folding BCs (5b) and (5c). They ensure that the solution of the folded system has the same regularity properties as the original system, which requires continuity of the state and its first spatial derivative at the folding point. The appearance of these BCs is the main reason for new challenges in the backstepping control design for (1).

To be able to apply the backstepping method presented in [8,15], the diffusion coefficients $\lambda_i(z)$ resulting from the folding transformation must differ at each point $z$. While the restriction to distinct diffusion coefficients is considered for the sake of a simpler presentation and can be extended to allow equal coefficients without obstacles, the case of intersecting coefficients, i.e., being equal only on a finite subset of the spatial domain, has not yet been considered. Consequently, depending on the shape of the diffusion coefficients, the folding point must be chosen appropriately to ensure $\lambda \geq \lambda_1(z) > \ldots > \lambda_{2n}(z) \geq \lambda > 0, z \in [0,1]$. However, this is no strong restriction, since due to the different scaling with $\frac{1}{\hat{z}_0}$ and $\frac{1}{1 - \hat{z}_0}$ according to (6), appropriate folding points $\hat{z}_0$ can always be found to ensure this property. This result is valid for both small enough and large enough folding points, where the latter simply requires a reordering of the folded states. Moreover, in the included special case of constant diffusion coefficients, the folding point can be chosen without any restriction.

In the next steps, the state feedback controller (2) is calculated by designing an intermediate feedback law for
the new input \( u(t) \in \mathbb{R}^{2n} \) in (5d) using the backstepping method.

3.2 Backstepping transformation

Following the procedure known from the unilateral case, the boundedly invertible backstepping transformation

\[
\tilde{x}(z, t) = x(z, t) - \int_0^z K(z, \zeta)x(\zeta, t) d\zeta,
\]

(8)

into the new state \( \tilde{x}(z, t) = \text{col}(\tilde{x}^1(z, t), \tilde{x}^r(z, t)) \) and the state feedback

\[
u(t) = \tilde{u}(t) + K(1, 1)x(1, t) + \int_0^1 K_x(1, \zeta)x(\zeta, t) d\zeta
\]

(9)

with the kernel \( K(z, \zeta) \in \mathbb{R}^{2n \times 2n} \) and the new input \( \tilde{u}(t) = \text{col}(0, \tilde{u}^r(t)) \in \mathbb{R}^{2n} \) with \( \tilde{u}^r(t) \in \mathbb{R}^n \) are applied to map the system (5) into the intermediate target system

\[
\tilde{x}_t(z, t) = \Lambda(z)\tilde{x}_{zz}(z, t) - \mu \tilde{x}(z, t)
\]

(10a)

\[
\tilde{x}_r(0, t) = \tilde{x}^1(0, t)
\]

(10b)

\[
\tilde{x}^1_z(0, t) = -\tilde{z}_0\tilde{x}^1_z(0, t)
\]

(10c)

\[
\tilde{x}_z(1, t) = \tilde{u}(t),
\]

(10d)

with (10a) defined on \((z, t) \in (0, 1) \times \mathbb{R}^+\), where \( \mu \in \mathbb{R} \) is the design parameter to specify the stability margin of the system. The matrices \( \tilde{A}_0(z) = [\tilde{A}_{0,ij}(z)] \in \mathbb{R}^{2n \times 2n} \), \( \tilde{A}_1(z) = [\tilde{A}_{1,ij}(z)] \in \mathbb{R}^{2n \times 2n} \) are strictly lower triangular and are introduced to ensure well-posedness of the kernel equations for \( K(z, \zeta) \) (cf. [20,8]). Representing them by

\[
\tilde{A}_i(z) = \begin{bmatrix} \tilde{A}^i_1(z) & 0 \\ \tilde{A}^i_n(z) & -\tilde{A}^i_1(z) \end{bmatrix},
\]

(11)

\( i = 0, 1 \), shows that they lead to a coupling between the left and right subsystem via \( \tilde{A}^r_1(z) \), which will be eliminated in the next step. To simplify the presentation in the following chapters, note that by inserting (10b) and (10c) into the coupling terms in (10a), the PDE (10a) can be rewritten in the form \( \tilde{x}_t(z, t) = \Lambda(z)\tilde{x}_{zz}(z, t) - \mu \tilde{x}(z, t) - \tilde{A}_0(z)\tilde{x}(0, t) - \tilde{A}_1(z)\tilde{x}_z(0, t) \) with

\[
\tilde{A}_i(z) = \begin{bmatrix} \tilde{A}^i_1(z) & 0 \\ \tilde{A}^i_n(z) & -\tilde{A}^i_1(z) \end{bmatrix}, \quad i = 0, 1.
\]

(12)

3.3 Decoupling transformation

Due to the coupling BCs and the well-posed matrices \( \tilde{A}_0(z), \tilde{A}_1(z) \), the intermediate target system (10) has an involved structure, which hinders the stability analysis. To facilitate this problem, the boundedly invertible Volterra-Fredholm type decoupling transformation

\[
\tilde{x}^r(z, t) = \tilde{x}^r(z, t) - \int_0^z Q(z, \zeta)\tilde{x}^r(\zeta, t) d\zeta
\]

\[
- \int_0^1 P(z, \zeta)\tilde{x}^1(\zeta, t) d\zeta
\]

(13)

with the new state \( \tilde{x}^r(z, t) \in \mathbb{R}^n \) and the kernels \( P(z, \zeta), Q(z, \zeta) \in \mathbb{R}^{n \times n} \) is applied to the \( \tilde{x}^r \)-system and the additional state feedback

\[
u^r(t) = \int_0^1 \tilde{R}_f(\zeta)\tilde{x}(\zeta, t) d\zeta
\]

(14)

with \( \tilde{R}_f(\zeta) = [P_z(1, \zeta) Q_z(1, \zeta)] \in \mathbb{R}^{n \times 2n} \) is utilized. They map (10) into the final target system

\[
\tilde{x}_1(z, t) = \Lambda_1(z)\tilde{x}^1(z, t) - \mu \tilde{x}^1(z, t)
\]

(15a)

\[
- \tilde{A}_0^1(z)\tilde{x}^1(0, t) - \tilde{A}_1^1(z)\tilde{x}^1_z(0, t)
\]

(15b)

\[
\tilde{x}^r(0, t) = \tilde{x}^r(0, t)
\]

(15c)

\[
\tilde{x}^1(0, t) = -\tilde{z}_0\tilde{x}^1(0, t)
\]

(15d)

\[
\tilde{x}^1_z(1, t) = \tilde{x}^1_z(1, t) = 0,
\]

(15e)

with the strictly lower triangular coupling matrices \( \tilde{A}^1_i(z) \in \mathbb{R}^{n \times n}, i = 0, 1 \), in which the coupling of the right and left states is removed in the PDEs. This will result in a cascade of parabolic PDEs after unfolding.

To show that (13) is boundedly invertible, write it as

\[
\tilde{x}^r(z, t) = T_v[\tilde{x}^r(t)](z) - \int_0^1 P(z, \zeta)\tilde{x}^1(\zeta, t) d\zeta,
\]

(16)

where \( T_v[\tilde{x}^r(t)](z) = \tilde{x}^r(z, t) - \int_0^z Q(z, \zeta)\tilde{x}^r(\zeta, t) d\zeta \) is a Volterra-type integral operator. Hence, the inverse transformation reads

\[
\tilde{x}^r(z, t) = T_v^{-1} [\tilde{x}^r(\cdot, t) + \int_0^1 P(\cdot, \zeta)\tilde{x}^1(\zeta, t) d\zeta] (z),
\]

(17)

which requires solely the inversion of the Volterra-type integral operator \( T_v \). This inverse inherently exists and is bounded with the same reasoning as for (8). Therefore, the inverse transformation (17) is bounded.

To be able to realize (14), the backstepping transformation (8) needs to be inserted for the target system state \( \tilde{x}(z, t) \). After changing the order of integration, the additional state feedback then reads

\[
u^r(t) = \int_0^1 \left( \tilde{R}_f(\zeta) - \int_\zeta^1 \tilde{R}_f(z)K(z, \zeta) dz \right) x(\zeta, t) d\zeta
\]

(18)
and after inserting into (9)

\[ u(t) = K(1,1)x(1,t) + \int_0^1 \left( K_z(1,\zeta) + \text{col}(0, \tilde{R}_f(\zeta)) \right) x(\zeta,t) d\zeta \]  

(19)

is the final state feedback in terms of the folded state \( x(z,t) \).

With (19) and the folding definitions (5d), (4), the state feedback controller (2) for the original system directly follows as

\[
\begin{bmatrix}
  u_0(t) \\
  u_1(t)
\end{bmatrix} = \begin{bmatrix} B_0 w(0,t) \\ B_1 w(1,t) \end{bmatrix} + \begin{bmatrix} -\frac{1}{z_0} I_n & 0 \\ 0 & -\frac{1}{z_0} I_n \end{bmatrix} \begin{bmatrix} \tilde{R}_w(0,\tilde{\zeta}) \\ \tilde{R}_w(1,\tilde{\zeta}) \end{bmatrix} \]  

\[ \cdot \left( R_0 w(0,t) + R_1 w(1,t) + \int_0^1 R(\tilde{\zeta}) w(\tilde{\zeta},t) d\tilde{\zeta} \right) \]  

\[ (20a) \]

with \( R_b = [R_{b,ij}] \in \mathbb{R}^{2n \times n}, b = 0,1, R(\tilde{\zeta}) = [R_{ij}(\tilde{\zeta})] \in \mathbb{R}^{2n \times n} \), and

\[ R_{i,j}(\tilde{\zeta}) = \begin{cases} \frac{1}{n_0} R_{f,ij}(\frac{\tilde{\zeta}_0 - \tilde{\zeta}}{n_0}), & \tilde{\zeta} < \tilde{\zeta}_0 \\ \frac{1}{1 - n_0} R_{f,ij+n}(\frac{\tilde{\zeta} - \tilde{\zeta}_0}{1 - n_0}), & \tilde{\zeta} \geq \tilde{\zeta}_0, \end{cases} \]  

\[ (20b) \]

\[ i = 1, \ldots, 2n, j = 1, \ldots, n. \]

3.4 Stability of the target system

To analyse the stability of the target system, (15) is unfolded back to the original representation with

\[ \tilde{w}(\tilde{z},t) = \begin{cases} \tilde{z}'((\tilde{z}_0 - \tilde{z})/\tilde{\zeta}_0, t), & \tilde{z} < \tilde{\zeta}_0 \\ \tilde{z}'((\tilde{z} - \tilde{\zeta}_0)/(1 - \tilde{z}_0), t), & \tilde{z} \geq \tilde{\zeta}_0 \end{cases} \]  

leading to

\[ \begin{align*}
\tilde{w}_t(\tilde{z},t) &= \tilde{A}(\tilde{z}) \tilde{w}_z(\tilde{z},t) - \mu \tilde{w}(\tilde{z},t) \\
- A_0(\tilde{z}) \tilde{w}_z(\tilde{z}_0,t) - A_1(\tilde{z}) \tilde{w}_z(\tilde{z}_0,t) & (22a) \\
\tilde{w}_z(0,t) &= 0 \quad (22b) \\
\tilde{w}_z(1,t) &= 0 \quad (22c)
\end{align*} \]

where \( A_b(\tilde{z}) = [A_{b,ij}(\tilde{z})], b = 0,1, i,j = 1, \ldots, n, \) with

\[ \begin{align*}
A_{0,ij}(\tilde{z}) &= \begin{cases} \tilde{A}_{0,ij}(\tilde{z}), & \tilde{z} < \tilde{\zeta}_0 \\ \tilde{A}_{0,ij}(\tilde{z}), & \tilde{z} \geq \tilde{\zeta}_0 \end{cases} \\
A_{1,ij}(\tilde{z}) &= \begin{cases} -\tilde{\zeta}_0 \tilde{A}_{1,ij}(\tilde{z}), & \tilde{z} < \tilde{\zeta}_0 \\ (1 - \tilde{\zeta}_0) \tilde{A}_{1,ij}(\tilde{z}), & \tilde{z} \geq \tilde{\zeta}_0 \end{cases}
\end{align*} \]  

\[ (23a, 23b) \]

are strictly lower triangular matrices when taking (3) into account. Therefore, the unfolded target system (22) has the well-known form, i.e., it is a cascade of exponentially stable parabolic PDEs (see [19,8]), except for the fact that the coupling due to \( A_1(\tilde{z}), i,j = 1, \) occurs at the folding point \( \tilde{\zeta}_0 \) rather than at the left boundary. This, however, does not change the reasoning in the corresponding stability proof of [9] so that the following lemma is valid.

**Lemma 1** (Stability of the target system). Assume that \( \mu > 0 \). Then, the initial value problem (IVP) for the unfolded target system (22) is well-posed and (22) is exponentially stable in the weighted \( L_2 \)-norm \( ||h|| = (\int_0^1 ||\tilde{A}^-\frac{1}{\tilde{z}}(\tilde{z}) h(\tilde{z})||_{\mathbb{C}^n} d\tilde{z})^{1/2} \), i.e.,

\[ ||\tilde{w}(t)|| \leq M e^{(-\mu + c)t} ||\tilde{w}(0)||, \quad t \geq 0 \]  

(24)

for all \( \tilde{w}(0) \in (L_2(0,1))^n \) an \( M \geq 1 \) and any \( c > 0 \) such that \( -\mu + c < 0 \).

**PROOF.** By noting that the point evaluation is a relatively bounded operator for all \( \tilde{\zeta}_0 \in (0,1) \) (see [14, Ch. IV, §1, Sec. 2]), the proof directly follows from [9]. Since the latter result ensures that the corresponding system operator is the infinitesimal generator of an analytic \( C_0 \)-semigroup, choosing the ICs in \( L_2(0,1) \) leads to a unique mild solution of (22) verifying well-posedness (see [7, Lem 3.1.5]). Furthermore, the decay rate is determined by the spectrum of the system operator (for details, see [9]).

Note that the decoupling of the left and right subsystems by (13) leads to a significant simplification of the target system structure so that the stability margin can be explicitly assigned, like in the unilateral case.

3.5 Closed-loop stability

Due to the bounded invertibility of (8) and (13), the stability of the unfolded target system (22) according to Lemma 1 implies the stability of the closed-loop folded system (5) with (9). After unfolding back to the system representation of (1), this directly leads to the following theorem.

**Theorem 2** (Closed-loop stability). Assume that \( \mu > 0 \). Then, the IVP for the closed-loop system (1) with (20) is well-posed and the system is exponentially stable in the weighted \( L_2 \)-norm \( ||h|| = (\int_0^1 ||\tilde{A}^{-\frac{1}{\tilde{z}}}(\tilde{z}) h(\tilde{z})||_{\mathbb{C}^n} d\tilde{z})^{1/2} \), i.e.,

\[ ||w(t)|| \leq M e^{(-\mu + c)t} ||w(0)||, \quad t \geq 0 \]  

(25)

for all \( w(0) \in (L_2(0,1))^n \), an \( M \geq 1 \) and any \( c > 0 \) such that \( -\mu + c < 0 \).

4 Solution of the kernel equations

To allow the calculation of the feedback gains (20b)–(20d), the kernels \( K(z,\zeta), Q(z,\zeta) \) and \( P(z,\zeta) \) of both the backstepping transformation (8) and the decoupling transformation (13) need to be determined.
4.1 Backstepping transformation

In order to map (5) into the intermediate target system (10), $K(z, \zeta)$ must be the solution of the kernel equations

$$
\Lambda(z) (K_{zz}(z, \zeta) - (K(z, \zeta))_{\zeta} = K(z, \zeta)(A(\zeta) + \mu I) \tag{26a}
$$

$$
\Lambda(z)(K'(z, z) + \Lambda(z)K_z(z, z) + K_\zeta(z, \zeta)\Lambda(z) + K(z, z)\Lambda'(z) = -(A(z) + \mu I) \tag{26b}
$$

$$
K(0,0) = 0 \tag{26d}
$$

$$
K(z, 0)\Lambda(0)S_1 + \tilde{A}_1(z)S_1 = 0 \tag{26e}
$$

$$
K_\zeta(z, 0)\Lambda(0)S_2 + K(z, 0)\Lambda'(0)S_2 - \tilde{A}_0(z)S_2 = 0 \tag{26f}
$$

with (26a) defined on $0 < \zeta < z < 1$ and

$$
\begin{bmatrix}
-\tilde{z}_0 I_n \\
I_n
\end{bmatrix}, \quad S_2 = \begin{bmatrix}
I_n \\
I_n
\end{bmatrix}, \tag{27}
$$

which follow from the same calculations as in [20,8]. The kernel equations (26) are similar to the ones found in [8] except for the new folding BCs (26e), (26f), requiring significant modifications of the solution procedure. In the remainder of this section, the following theorem will be proved.

**Theorem 3** (Kernel equations of the Volterra transformation). The kernel equations (26) have a piecewise continuous solution on the spatial domain $0 \leq \zeta \leq z \leq 1$.

This result, in particular, means that (8) exists and maps $L_2$-functions into $L_2$-functions, which is sufficient for the shown closed-loop stability property. Depending on the regularity of the system parameters, it is also possible to verify a higher regularity of the kernel.

4.1.1 Canonical kernel equations

The solution of the kernel equations (26) relies on a transformation into integral equations and their solution with the method of successive approximations using the approach in [15]. In contrast to [8], it is shown in [15] that the step of eliminating the first-order derivatives in the canonical kernel equations is not necessary, simplifying the derivation of the integral equations.

As first step, the kernel equations (26) are considered for each matrix element $K(z, \zeta) = K_{ij}(z, \zeta)$, $i, j = 1, \ldots, 2n$. For this, it is convenient to first evaluate the component form of the BC (26e), resulting in

$$
\tilde{z}_0 K(z, 0)\lambda_j(0) + \tilde{z}_0 \tilde{A}_{1,ij}(z) = K_{ij+n}(z, 0)\lambda_{j+n}(0) + \tilde{A}_{1,ij+n}(z) \tag{28a}
$$

for $j \leq n$ which is equivalent to

$$
\tilde{z}_0 K_{ij-n}(z, 0)\lambda_{j-n}(0) + \tilde{z}_0 \tilde{A}_{1,ij-n}(z) = K(z, 0)\lambda_j(0) + \tilde{A}_{1,ij}(z) \tag{28b}
$$

with $j > n$. This shows that the respective left ($j \leq n$) and right ($j > n$) elements of the kernel $K(z, \zeta)$ and the matrix $\tilde{A}_1(z)$ are coupled. Similarly, the component form of (26f) reads

$$
K_\zeta(z, 0)\lambda_j(0) + K(z, 0)\lambda_j(0) - \tilde{A}_{1,ij}(z) - \tilde{A}_{1,ij+j+n}(z) = -K_{ij+n,\zeta}(z, 0)\lambda_{j+n}(0) - K_{ij+j+n}(z, 0)\lambda_{j+n}(0). \tag{29}
$$

In the following, (28b) is considered as a BC for the right elements ($j > n$), whereas (29) is a BC for the left elements ($j \leq n$), which is indicated by the orange areas in Figure 1.

The coupling matrices $\tilde{A}_0(z)$ and $\tilde{A}_1(z)$ are needed to ensure well-posedness of the kernel equations (see [8]), naming them well-posedness terms. Their task is to fulfill the BCs (28) and (29) for some indices so that no condition on the respective kernel element results. Yet, they can only cover so much conditions that they attain a strictly lower triangular structure, i.e., $\tilde{A}_{k,ij}(z) \neq 0$, $k = 0, 1$, only for $i > j$ which is important for the stability of the target system. In the non-folding case, they need to remove the BC at the lower boundary ($z, 0$) for all kernel elements with $i > j$ (see [8]). In the folding case, (28a) shows that removing the BC for elements
with \( j < i \leq n \) by choosing
\[
\tilde{A}_{1,ij+n}(z) = 0 \quad \text{(30a)}
\]
\[
\tilde{A}_{1,ij}(z) = -\mathbf{K}(z,0)\lambda_j(0) + \frac{1}{\tilde{z}_0}K_{ij+n}(z,0)\lambda_{j+n}(0) \quad \text{(30b)}
\]
for \( j \leq n \) also removes the BC for the element \( K_{ij+n}(z,\zeta) \). Covering all index combinations this way and performing the same considerations for (29) finally leads to the component form of the kernel equations
\[
\lambda_i(z)\mathbf{K}_{zz}(z,\zeta) - (\lambda_j(\zeta)\mathbf{K}(z,\zeta))\zeta\zeta
= \sum_{k=1}^{2n} \left( K_{ik}(z,\zeta)A_{kj}(\zeta) \right) + \mu\mathbf{K}(z,\zeta) \tag{31}
\]
defined on \( 0 < \zeta < z < 1 \), with the BCs
\[
i = j : \\
\mathbf{K}(z,z) = -\int_0^z \frac{A_{ii}(\zeta) + \mu}{2\sqrt{\lambda_i(\zeta)\lambda_i(z)}} d\zeta \tag{32a}
\]
\[
i \leq n : \\
\mathbf{K}_{\zeta}(z,0)\lambda_i(0) + \mathbf{K}(z,0)\lambda'_i(0) = -K_{ii+n,\zeta}(z,0)\lambda_i+n(0) - K_{ii+n}(z,0)\lambda'_i+n(0) \tag{32b}
\]
and
\[
i \neq j : \\
\mathbf{K}(z,z) = 0 \quad \text{(33a)}
\]
\[
\mathbf{K}_{\zeta}(z,0)\lambda_j(0) = K_{ij-n}(z,0)\lambda_j-n(0)\tilde{z}_0 \tag{33b}
\]
\[
j > n, \ i \leq j - n : \\
\mathbf{K}(z,0)\lambda_j(0) = K_{ij-n}(z,0)\lambda_j-n(0)\tilde{z}_0 \tag{33c}
\]
\[
j \leq n, \ i \neq j : \\
\mathbf{K}_{\zeta}(z,0)\lambda_j(0) + \mathbf{K}(z,0)\lambda'_j(0) = -K_{ij+n,\zeta}(z,0)\lambda_j+n(0) - K_{ij+n}(z,0)\lambda'_j+n(0) \tag{33d}
\]
when the non-zero elements of \( \tilde{A}_0(z) \) and \( \tilde{A}_1(z) \) are chosen to be
\[
i > j, \ i - n \leq j \leq n : \\
\tilde{A}_{0,ij}(z) = \lambda_j(0)\mathbf{K}_{\zeta}(z,0) + \lambda'_j(0)\mathbf{K}(z,0) \tag{34a}
\]
\[
= \lambda_j+n(0)K_{ij+n,\zeta}(z,0) + \lambda'_j+n(0)K_{ij+n}(z,0) \tag{34b}
\]
\[
i > j > n : \\
\tilde{A}_{0,ij}(z) = +\lambda_j(0)\mathbf{K}_{\zeta}(z,0) + \lambda'_j(0)\mathbf{K}(z,0) \tag{34c}
\]
\[
+ \lambda_j-n(0)K_{ij-n,\zeta}(z,0) + \lambda'_j-n(0)K_{ij-n}(z,0) \tag{34d}
\]
\[
\tilde{A}_{1,ij}(z) = -\lambda_j(0)\mathbf{K}(z,0) + \tilde{z}_0\lambda_{j-n}(0)K_{ij-n}(z,0). \tag{34d}
\]

Figure 1 provides a graphical overview of the related index combinations for the kernel BCs and the coupling matrix elements needed to fulfill (28) and (29). Note that with the choice (34), the matrices \( A_0(z) \) and \( \tilde{A}_1(z) \) have the structure
\[
\tilde{A}_k(z) = \begin{bmatrix} \tilde{A}_{k1}(z) & 0 \\ \tilde{A}_{k2}(z) & \tilde{A}_{k2}(z) \end{bmatrix}, \quad k = 0, 1, \tag{35}
\]
where \( \tilde{A}_{k1}(z) \) and \( \tilde{A}_{k2}(z) \) are strictly lower triangular matrices and \( \tilde{A}_{k2}(z) \) is an upper triangular matrix so that in total, \( A_0(z) \) and \( \tilde{A}_1(z) \) are strictly lower triangular (blue regions in Figure 1).

Following the lines in [15], the component form (31) of the kernel equations is mapped into its canonical form with the new kernel elements
\[
\mathbf{G}(\xi,\eta) = G_{ij}(\xi_{ij},\eta_{ij}) = \lambda_j(\zeta(\xi,\eta))K_{ij}(z(\xi,\eta),\zeta(\xi,\eta)). \tag{36}
\]

For this, the transformation to normalize the coefficients of the highest derivative to 1 is combined with the transformation of a wave equation into its canonical form. In particular, the canonical coordinates
\[
\xi = \xi_{ij}(z,\zeta) = \frac{1}{2}(1-s)(\phi_1(1) + \phi_2(1)) + s(\phi_1(z) + \phi_2(\zeta)) \tag{37a}
\]
\[
\eta = \eta_{ij}(z,\zeta) = -\frac{1}{2}(1-s)(\phi_1(1) - \phi_2(1)) + \phi_i(z) - \phi_j(\zeta) \tag{37b}
\]
and their inverses
\[
z = z(\xi,\eta) = \phi_i^{-1}\left(\frac{s}{2}(s\xi + \eta) + \frac{1}{2}(1-s)\phi_i(1)\right) \tag{38a}
\]
\[
\zeta = \zeta(\xi,\eta) = \phi_j^{-1}\left(\frac{1}{2}(s\xi - \eta) + \frac{1}{2}(1-s)\phi_j(1)\right) \tag{38b}
\]
are considered, where
\[
s = s_{ij} = \begin{cases} 1, & i \leq j \\ -1, & i > j \end{cases} \tag{39}
\]
and
\[
\phi_i(z) = \int_0^z \frac{d\zeta}{\sqrt{\lambda_i(\zeta)}}, \quad i = 1, \ldots, 2n \tag{40}
\]
(see [15]). This leads to the canonical kernel equations
\[
G_{\xi\eta}(\xi,\eta) = \mathbf{F}[G_{\xi}, G_{\eta}](\xi,\eta)
= \frac{s}{4} \sum_{k=1}^{2n} \left( \frac{\lambda_j(\zeta)}{\lambda_k(\zeta)}G_{ik}(\xi,\eta)A_{kj}(\zeta) \right) \left| \zeta = \zeta(\xi,\eta) \right| + \frac{s\mu}{4} G_{\xi}(\xi,\eta)
\]
\[
- \frac{1}{4} \bar{a}_\Delta(\xi,\eta)G_{\xi}(\xi,\eta) + \frac{s}{4} \bar{a}_\Sigma(\xi,\eta)G_{\eta}(\xi,\eta) \tag{41a}
\]
with the BCs

\[
\begin{align*}
  i & = j: \\
  G(\xi, 0) &= -\sqrt{\lambda_i(z(\xi, 0))} \int_0^{z(\xi, 0)} \frac{A_{ii}(\bar{\zeta}) + \mu}{2 \sqrt{\lambda_i(\bar{\zeta})}} d\bar{\zeta} \\
  i & \leq n: (G_{\eta}(\eta, \eta) - G(\xi, \eta)) \bar{\zeta}_0 \\
  &= G_{ii+n, \xi}(\eta, \eta) - G_{ii+n, \eta}(\eta, \eta) \\
  i & \neq j: \\
  G(\xi, \eta(\xi)) &= 0 \\
  sG(\xi, \eta(\xi)) + G_\eta(\xi, \eta(\xi)) \\
  &= A_{ij}(z) \sqrt{\lambda_i(z)} \frac{\lambda_j(z) - \lambda_i(z)}{\lambda_j(z)} \\
  j & > n, i \leq j - n: G(\eta, \eta) = \bar{\zeta}_0 G_{ij-n}(\eta, \eta) \\
  j & \leq n, i < j: (G_{\eta}(\eta, \eta) - G_\xi(\xi, \eta)) \bar{\zeta}_0 \\
  &= G_{ij+n, \xi}(\eta, \eta) - G_{ij+n, \eta}(\eta, \eta) \\
  j & > n, i > j - n: G(\eta, \eta) = g_f(\eta) \\
  j & \leq n, i > j: G_{\eta}(\eta, \eta) = g_f(\eta)
\end{align*}
\]

and

\[
\begin{align*}
  \bar{\alpha}_\xi(\xi, \eta) &= \frac{\lambda_j(z)}{2 \sqrt{\lambda_i(z)}} - \frac{\lambda_i(z)}{2 \sqrt{\lambda_j(z)}} \\
  \bar{\alpha}_\xi(\xi, \eta) &= \frac{\lambda_j(z)}{2 \sqrt{\lambda_i(z)}} + \frac{\lambda_i(z)}{2 \sqrt{\lambda_j(z)}}
\end{align*}
\]

where \((z, \zeta)\) are substituted by \((38)\). In \((41c)\) and \((41g)\), the result \(\sqrt{\lambda_j(z)} = \bar{z}_0 = \bar{z}_0/(1 - \bar{z}_0)\) was applied, which can be derived from \((6)\). The fact that \(\xi_{ij+n}(\eta, \eta) = \eta_{ij+n}(\eta, \eta) = \eta\) for \(i \leq j \leq n\), respectively \(\xi_{ij-n}(\eta, \eta) = \eta_{ij-n}(\eta, \eta) = \eta\) for \(i \leq j - n\) is utilized in \((41c)\), \((41f)\) and \((41g)\). In \((41d)\) and \((41c)\) \(\eta(\xi)\) is the strictly monotonically decreasing lower boundary of the canonical spatial domain (see \([8]\) and Figure 2), which is also defined in the case \(i = j\) as \(\eta(\xi) = 0\). The artificial BCs \((41h)\), \((41i)\) have been introduced to fully determine the kernel and contain the degrees of freedom \(g_f = g_{ij} \in C[0, \bar{\eta}], i, j = 1, \ldots, 2n\), where \(\eta = \phi_1(1)\) for \(i \leq j\) and \(\eta = \phi_j(1)\) for \(i > j\). Their introduction as Dirichlet respectively Neumann BCs is defined by the way the equations will be converted into integral equations in the next step. The indices for which they are provided are depicted by the red dashed areas in Figure 1. Note that in contrast to the unilateral case, there also exist artificial BCs for \(i \leq j\).

### 4.1.2 Kernel integral equations

For the conversion into integral equations, \((41a)\) is formally integrated w.r.t. \(\xi\) and \(\eta\). Due to the different types of coupling BCs \((41f)\) for \(j > n\) and \((41c), (41g)\) for \(j \leq n\), a different sequence of the formal integration is needed for elements with \(j \leq n\) and \(j > n\). This is visualized in Figure 2. Starting with \(j > n\), \((41a)\) is integrated w.r.t. \(\eta\) first and then w.r.t. \(\xi\) as in \([15]\), leading to

\[
\begin{align*}
  G_\xi(\xi, \eta(\xi)) &= G_\xi(\xi, \eta(\xi)) + \int_{\eta(\xi)}^{\eta} G_{\xi}(\xi, \bar{\eta}) d\bar{\eta} \\
  \eta(\xi) &= \int_{\eta(\xi)}^{\xi} G(\xi, \bar{\eta}) d\xi \\
  \xi_i(\eta) &= \eta_i(\eta), \quad \eta_i(\eta) \\
  \eta_i(\eta) &\geq 0, \quad \eta_i(\eta) < 0
\end{align*}
\]

which is the left boundary of the spatial domain shown in Figure 2. Introducing the new variable

\[
H(\xi, \eta) := G_\xi(\xi, \eta)
\]

results in

\[
\begin{align*}
  H(\xi, \eta) &= H(\xi, \eta(\xi)) + \int_{\eta(\xi)}^{\eta} F[G, H, G_{\eta}](\xi, \bar{\eta}) d\bar{\eta} \\
  G(\xi, \eta) &= G(\xi, \eta(\xi)) + \int_{\eta(\xi)}^{\xi} H(\xi, \eta) d\xi
\end{align*}
\]

for \(j > n\) after \(G_{\xi\eta}\) has been substituted by the right side of the PDE \((41a)\). It can be seen that \((46)\) requires suitable BCs for \(H(\xi, \eta(\xi)) = G_\xi(\xi, \eta(\xi))\) and \(G(\xi(\eta), \eta)\).

For \(j \leq n\), \((41a)\) is integrated in the reverse order to get

\[
\begin{align*}
  G_\eta(\xi, \eta) &= G_\eta(\xi, \eta) + \int_{\eta(\xi)}^{\eta} G_{\eta\xi}(\xi, \bar{\eta}) d\bar{\eta} \\
  G(\xi, \eta) &= G(\xi, \eta(\xi)) + \int_{\eta(\xi)}^{\eta} G_\eta(\xi, \bar{\eta}) d\bar{\eta},
\end{align*}
\]

where the introduction of

\[
J(\xi, \eta) := G_\eta(\xi, \eta)
\]
provides

\[
J(\xi, \eta) = J(\xi_l(\eta), \eta) + \int_{\xi_l(\eta)}^{\xi} F[G, G_\xi, J(\bar{\bar{\xi}}, \eta)]d\bar{\bar{\eta}}
\] (49a)

\[
G(\xi, \eta) = G(\xi, \eta_l(\xi)) + \int_{\eta_l(\xi)}^{\eta} J(\xi, \bar{\bar{\eta}})d\bar{\bar{\eta}},
\] (49b)

which need BCs for \( J(\xi_l(\eta), \eta) = G_\eta(\xi_l(\eta), \eta) \) and \( G(\xi, \eta_l(\xi)) \). In the following, the appearing boundary terms in (46) and (49) are substituted by suitable BCs.

As first step, with \( \eta_l(\xi) = 0, i = j \), the BCs (41b) and (41d) can directly be inserted for \( G(\xi, \eta_l(\xi)) \) in (49b). To substitute \( H(\xi, \eta_l(\xi)) = G_\xi(\xi, \eta_l(\xi)) \) in (46a), differentiate (41d) w.r.t. \( \xi \) to obtain

\[
G_\xi(\xi, \eta_l(\xi)) + G_{\eta_l}(\xi, \eta_l(\xi))\eta_l(\xi) = 0, \quad i \neq j.
\] (50)

Now, solving (41e) for \( G_{\eta_l}(\xi, \eta_l(\xi)) \) and inserting the result into (50) yields

\[
H(\xi, \eta_l(\xi)) = \frac{c^2(\xi)\eta_l^2(\xi)}{s\eta_l(\xi) - 1}, \quad i \neq j,
\] (51)

where the denominator cannot be zero (see [8]). Moreover, differentiating (41b) w.r.t. \( \xi \) gives

\[
H(\xi, \eta_l(\xi)) = H(\xi, 0) = \frac{\partial \xi c^1(\xi)}{c^1(\xi)}, \quad i = j.
\] (52)

The remaining boundary terms incorporate the coupling BCs (41c), (41f) and (41g) and thus need special attention.

For the formulation of \( G(\xi_l(\eta), \eta) \) in (46b), first note that the lower boundary \( \eta_l(\xi) \) of the domain \( (\xi, \eta) \) is the left boundary of the domain for \( \eta < 0 \), too (see Figure 2), so that inserting \( \xi = \xi_l(\eta) = \eta^{-1}(\xi) \) in (41d) yields

\[
G(\xi_l(\eta), \eta) = 0, \quad \eta < 0.
\] (53)

With (44), this leads to

\[
G(\xi_l(\eta), \eta) = \begin{cases}
G(\xi_l(\eta), \eta), & \eta \geq 0 \\
0, & \eta < 0,
\end{cases}
\] (54)

where the BCs (41f) and (41h) can be inserted for \( \eta \geq 0 \). Then, (46b) reads

\[
G(\xi, \eta) = [\bar{\bar{\delta}} G_{ij-n}(\eta, \eta)] i \leq j - n + [g_f(\eta)] i > j - n
\]

\[
+ \int_{\xi_l(\eta)}^{\xi} H(\xi, \eta)d\xi
\] (55)

for \( j > n \) (see right picture of Figure 2).

For \( G_{ij-n}(\eta, \eta) \), the respective result (49b) with (41b) and (41d) is now inserted to obtain

\[
G(\xi, \eta) = [\bar{\bar{\delta}} g_f(\eta)] i \geq j - n + \int_{\xi_l(\eta)}^{\xi} H(\xi, \eta)d\xi
\]

\[
+ \int_{\eta_l(\eta)}^{\eta} J_{ij-n}(\eta, \bar{\bar{\eta}})d\bar{\bar{\eta}}
\] (56)

for \( i \leq j - n \).

To substitute \( J(\xi_l(\eta), \eta) \) in (49a), the BCs (41e), (41c) and (41g) need to be formulated as conditions for \( J = G_\eta \). Solving (41g) for \( G_\xi(\xi, \eta_l(\xi)) \) and inserting
the result into (50) yields

\[ J(\xi(\eta), \eta) = \frac{c^2(\xi(\eta))}{1 - n\eta(\xi(\eta))} =: c^2(\eta), \quad i \neq j \quad (57) \]

in the case \( \eta < 0 \) after the substitution \( \xi = \xi_1(\eta) \). Since the denominator in (51) cannot be zero, the same holds for the one in (57).

For \( \eta \geq 0 \), (41f) is differentiated w.r.t. \( \eta \) yielding

\[ G_\xi(\eta, \eta) + G_\eta(\eta, \eta) = \tilde{z}G_{ij-n, \xi}(\eta, \eta) + \tilde{z}G_{ij-n, \eta}(\eta, \eta) \quad (58) \]

for \( j > n \), \( i \leq j - n \). Now note that for an element with column index \( j > n \) that is subject to (58), the coupled element with column \( j - n \) must fulfill one of the BCs (41c), (41g). This can be exploited by considering an index-shift \( j - n \rightarrow j \) for (58), leading to

\[ G_{ij+n, \xi}(\eta, \eta) + G_{ij+n, \eta}(\eta, \eta) = \tilde{z}G_{ij-n, \xi}(\eta, \eta) + \tilde{z}G_{ij-n, \eta}(\eta, \eta) \quad (59) \]

for \( j \leq n \), \( i \leq j \). Solving (59) for \( G_{ij+n, \eta}(\eta, \eta) \) and inserting the result into (41c) and (41g) leads to

\[ J(\eta, \eta) = \frac{1}{\tilde{z}}G_{ij-n, \xi}(\eta, \eta) \quad (60) \]

for \( i \leq j \leq n \), \( \eta \geq 0 \) after some rearrangements, where \( G_{ij+n, \xi}(\eta, \eta) = H_{ij+n}(\eta, \eta) \) can be replaced by (46a).

Together with (41f) and (57), the result (60) can now be inserted for \( J(\xi(\eta), \eta) \) in (49a).

In (46a) and (49a) both \( G_\xi = H \) and \( G_\eta = J \) appear under the integrals. Though (46a) is only required for \( j > n \) to determine \( G \) according to (46b), it can also be utilized for \( j \leq n \), because (51) and (52) are valid for all \( j = 1, \ldots, 2n \). Therefore, \( G_\xi \) can be replaced by \( H \) in (49a). On the contrary, the equation (49a) may only be used for \( j \leq n \), since the utilized folding BC (60) is only valid there. Thus, integration by parts needs to be applied in (46a) for \( j > n \) to eliminate \( G_\eta \), like shown in [15].

Then, substituting all considered boundary terms finally leads to the kernel integral equations

\[ G = G_0 + F_G[H, J] \quad (61a) \]

\[ H = H_0 + F_H[G, H, J] \quad (61b) \]

\[ j \leq n : \quad J = J_0 + F_J[G, H, J] \quad (61c) \]

with

\[ G_0 = \left[ c^1(\xi) \right]_{i = j} + \left[ g_f(\eta) \right]_{j > \eta} + \left[ \tilde{z}c^1(\eta) \right]_{i = j - n} \quad (62a) \]

\[ H_0 = H_{0, ij} = \left[ \frac{1}{4} \tilde{a}_\Sigma(\xi, \eta) c^1(\xi) \right]_{i \leq j} + \left[ c^4(\xi) \right]_{i = j} \]

\[ - \left[ \frac{1}{4} \tilde{a}_\Sigma(\xi, 0) c^1(\xi) \right]_{i = j} + \left[ \frac{1}{4} \tilde{a}_\Sigma(\xi, \eta) g_f(\eta) \right]_{j > \eta} + \left[ c^3(\xi) \right]_{i \neq j} + \left[ \frac{1}{4} \tilde{a}_\Sigma(\xi, \eta) \tilde{z}c^1(\eta) \right]_{i = j - n} \quad (62b) \]

\[ J_0 = \left[ c^3(\eta) \right]_{i \neq j} + \frac{1}{\tilde{z}} H_{0, ij+n}(\eta, \eta) \left[ \begin{array}{c} i \leq j \end{array} \right]_{\eta \geq 0} + \left[ g_f(\eta) \right]_{i \geq j} \quad (62c) \]

as well as

\[ F_G[H, J] = \left[ \int_{\eta(\xi)}^{\eta} J(\xi, \eta) d\eta \right]_{j \leq n} + \left[ H(\xi, \eta) d\xi \right]_{j > n} \]

\[ + \left[ \int_{\eta(\xi)}^{\eta} J_{ij-n}(\eta, \eta) d\eta \right]_{i \leq j - n} \quad (63a) \]

\[ F_H[G, H, J] = F_{H, ij}[G, H, J] = \left[ \frac{\tilde{a}_\Sigma(\xi, \eta) J(\xi, \eta)}{4 \tilde{z}} + \left[ \tilde{a}_\Sigma(\xi, \eta) \right]_{ij} \right]_{\xi(\eta)}^{\eta} \]

\[ + \left[ \tilde{a}_\Sigma(\xi, \eta) \tilde{z} \int_{\eta(\xi)}^{\eta} J_{ij-n}(\eta, \eta) d\eta \right]_{i \leq j - n} \quad (63b) \]

\[ F_J[G, H, J] = \left[ \int_{\eta(\xi)}^{\eta} \frac{\tilde{a}_\Sigma(\xi, \eta) J(\xi, \eta)}{4 \tilde{z}} + \left[ \tilde{a}_\Sigma(\xi, \eta) \right]_{ij} \right]_{\xi(\eta)}^{\eta} \]

\[ - \left[ \frac{1}{\tilde{z}} F_{H, ij+n}[G, H_{ij+n}, J](\eta, \eta) \right]_{i \leq j} \quad (63c) \]

where \( \tilde{a}(\xi, \eta) = \partial_\eta \tilde{a}_\Sigma(\xi, \eta) \). Note that \( F_G \) and \( F_H \) only contain \( J_{ij} \) with \( j \leq n \), which is determined by (61c).

4.2 Decoupling transformation

After differentiating (13) w.r.t. time as well as using (10) and (13), similar calculations as in [20,8] show that (10) is mapped into the final target system (15) if the kernels \( P(z, \zeta) \) and \( Q(z, \zeta) \) are the solution of the kernel equations

\[ \Lambda_r(z) P_{zz}(z, \zeta) - (P(z, \zeta) \Lambda_r(z)) \zeta \zeta = 0 \quad (64a) \]

\[ P(z, 1) \Lambda_r(1) + P(z, 1) \Lambda_r'(1) = 0 \quad (64b) \]

\[ P(0, \zeta) = 0 \quad (64c) \]

\[ P_z(0, \zeta) = 0 \quad (64d) \]
\[ \Lambda_r(z)Q_{zz}(z, \zeta) - (Q(z, \zeta) \Lambda_r(\zeta))_{\zeta \zeta} = 0 \]  
(64e)  
\[ Q_r(z, z)\Lambda_r(z) + Q(z, z)\Lambda_r(z) + \Lambda_r(z)Q'(z, z) \]  
+ \Lambda_r(z)Q_r(z, z) = 0 \]  
(64f)  
\[ \Lambda_r(z)Q(z, z) - Q(z, z)\Lambda_r(z) = 0 \]  
(64g)  
\[ Q(0, 0) = 0 \]  
(64h)

with the coupling BCs

\[
P(z, 0)\Lambda_r(0) - \frac{1}{\tau_0} Q(z, 0)\Lambda_r(0) = -\bar{A}^r_1(z) - \int_0^z Q(z, \zeta)\bar{A}^r_1(\zeta)d\zeta - \int_0^1 P(z, \zeta)\bar{A}^1_1(\zeta)d\zeta + \bar{A}^1_0(z), \]  
(64i)

\[ Q(z, 0)\Lambda_r(0) + Q(z, 0)\Lambda_r(0) + P_z(z, 0)\Lambda_r(0) + P(z, 0)\Lambda_r(0) = -\bar{A}^0_1(z) + \int_0^z Q(z, \zeta)\bar{A}^0_1(\zeta)d\zeta + \int_0^1 P(z, \zeta)\bar{A}^0_1(\zeta)d\zeta + \bar{A}^1_0(z), \]  
(64j)

in which \( P(z, \zeta) \) is defined on the rectangular domain \( 0 \leq z \leq 1, 0 \leq \zeta \leq 1 \) and \( Q(z, \zeta) \) is defined on the triangular domain \( 0 \leq z, \zeta \leq 1 \). Since the kernel equations for \( P(z, \zeta) \) and \( Q(z, \zeta) \) are coupled via the BCs (64i) and (64j), they cannot be solved independently, but a solution method needs to be derived allowing a simultaneous determination of both kernels. This challenging problem of solving coupled Fredholm-Volterra kernel equations is new in the backstepping framework and provides a general extension of the corresponding transformation presented in [5]. In the sequel, the following well-posedness result for these kernel equations will be shown.

**Theorem 4** (Kernel equations of the Volterra-Fredholm transformation). The kernel equations (64) have a piecewise continuous solution \( P(z, \zeta) \) on the spatial domain \( 0 \leq z \leq 1, 0 \leq \zeta \leq 1 \) and \( Q(z, \zeta) \) on the spatial domain \( 0 \leq z, \zeta \leq 1 \).

Since both PDEs (64a) and (64e) have the same spatial differential operator as (26a), the approach to solve (64) is the same as for the solution of (26). The main difference is that (64a) is now defined on a rectangular domain and that the coupling between the kernel elements appears solely in the coupling BCs. Therefore, the component forms of the equations for both kernels are transformed into canonical coordinates in the next step, which can then be converted into integral equations.

### 4.2.1 Canonical kernel equations

The equivalent component form of (64) for the matrix elements \( P(z, \zeta) = P_{ij}(z, \zeta), i, j = 1, \ldots, n, \) reads

\[ \lambda^r_{ij}(z)P_{zz}(z, \zeta) - (\lambda^r_{ij}(z)P(z, \zeta))_{\zeta \zeta} = 0 \]  
(65a)  
\[ \lambda^r_{ij}(1)P_{z}(z, 1) + \lambda^r_{ij}(1)P(z, 1) = 0 \]  
(65b)  
\[ P(0, \zeta) = 0 \]  
(65c)  
\[ P_z(0, \zeta) = 0 \]  
(65d)

and for \( Q(z, \zeta) = Q_{ij}(z, \zeta), i, j = 1, \ldots, n, \)

\[ \lambda^r_{ij}(z)Q_{zz}(z, \zeta) - (\lambda^r_{ij}(z)Q(z, \zeta))_{\zeta \zeta} = 0 \]  
(66a)  
\[ Q_z(z, z)\lambda^r_{ij}(z) + Q(z, z)\lambda^r_{ij}(z) + \lambda^r_{ij}(z)Q'(z, z) \]  
+ \lambda^r_{ij}(z)Q_z(z, z) = 0 \]  
(66b)  
\[ (\lambda^r_{ij}(z) - \lambda^r_{ij}(z))Q(z, z) = 0 \]  
(66c)  
\[ Q(0, 0) = 0 \]  
(66d)

The coupling BCs for (65) and (66) are

\[
i < j: \quad \lambda^r_{ij}(0)P(0, z) - \frac{1}{\tau_0} \lambda^r_{ij}(0)Q(z, 0) = -\sum_{k=1}^{n} \int_0^z Q_{ik}(z, \zeta)\bar{A}^r_{ik,kj}(\zeta)d\zeta - \sum_{k=1}^{n} \int_0^1 P_{ik}(z, \zeta)\bar{A}^1_{ik,kj}(\zeta)d\zeta + \bar{A}^1_{0,ij}(z)\]  
(67a)

\[
i = j: \quad \lambda^r_{0,ij}(z) = \lambda^r_{ij}(0)Q(z, 0) + \lambda^r_{ij}(0)Q(z, 0) + \lambda^r_{ij}(0)P(z, 0) \]  
(68a)  
\[ + \lambda^r_{ij}(0)P(0, z) - \sum_{k=1}^{n} \int_0^z Q_{ik}(z, \zeta)\bar{A}^r_{ik,kj}(\zeta)d\zeta - \sum_{k=1}^{n} \int_0^1 P_{ik}(z, \zeta)\bar{A}^1_{ik,kj}(\zeta)d\zeta + \bar{A}^1_{0,ij}(z)\]  
(68a)

\[
i > j: \quad \bar{A}^r_{0,ij}(z) = \bar{A}^r_{ij}(0)P(0, z) - \frac{1}{\tau_0} \bar{A}^r_{ij}(0)Q(z, 0) + \sum_{k=1}^{n} \int_0^z Q_{ik}(z, \zeta)\bar{A}^1_{ik,kj}(\zeta)d\zeta \]  
(68b)  
\[ + \sum_{k=1}^{n} \int_0^1 P_{ik}(z, \zeta)\bar{A}^1_{ik,kj}(\zeta)d\zeta + \bar{A}^1_{0,ij}(z)\]  
(68b)

Note that this degree of freedom is needed to ensure the well-posedness of the kernel equations for the Volterra-kernel \( Q(z, \zeta) \), just like it was for the backstepping kernel \( K(z, \zeta) \) (see Section 4.1.1). Since both kernels of the second transformation are of dimension \( n \times n \), the strictly lower triangular well-posedness terms defined by (68) are enough to ensure well-posedness of the kernel equations. This is a simplification compared to the first transformation with Figure 1.
While the BVP (65a)–(65d) is already in the form which can be transformed into canonical coordinates, a further simplification is required for (66a)–(66d). In contrast to (26), the BVPs for $Q(z, \zeta)$ contain no inhomogeneity except in the coupling BCs (67a), (67b). Thus, (66b) can be rewritten for $i = j$ to obtain

$$Q'(z, z) = -\frac{\lambda''(z)}{2\lambda'(z)} Q(z, z)$$

(69)

with the solution $Q(z, z) = 0$ following from (66d). Since $\lambda_i' \neq \lambda_j'$ holds for $i \neq j$, (66c) shows that $Q(z, z) = 0$ also holds in that case. Hence, the BVP (66a)–(66d) can be represented in simplified component form as

$$\lambda_i(z)Q_{zz}(z, \zeta) - (\lambda_i'(\zeta)Q(z, \zeta))_{\zeta\zeta} = 0$$

(70a)

$$Q(z, z) = 0$$

(70b)

$$[Q_{z}(z, z) = 0]_{i \neq j}$$

(70c)

complemented by the coupling BCs (67a), (67b).

To convert the kernel equations into their canonical form, the canonical coordinates

$$\xi^i(z, \zeta) = \phi_i^0(z) + \phi_i(z)$$

(71a)

$$\eta^i(z, \zeta) = \phi_i^0(z) - \phi_i(z)$$

(71b)

$$\xi^i(z, \zeta) = \frac{1}{2}(1-s)(\phi_i^0(1) + \phi_i^0(1)) + s(\phi_i^0(z) + \phi_i^0(1))$$

(71c)

$$\eta^i(z, \zeta) = -\frac{1}{2}(1-s)(\phi_i^0(1) - \phi_i^0(1)) + \phi_i^0(z) - \phi_i^0(1)$$

(71d)

with

$$\phi_i^0(z) = \int_0^z \frac{d\zeta}{\sqrt{\lambda_i(z)}}$$

(72a)

$$\phi_i(z) = \int_0^z \frac{d\zeta}{\sqrt{\lambda_i'(\zeta)}}$$

(72b)

$i = 1, \ldots, n$ and the corresponding inverses

$$z^{P}((\xi, \eta)) = (\phi_i^0)^{-1} \left( \frac{1}{2}(\xi + \eta) \right)$$

(73a)

$$z^{Q}((\xi, \eta)) = (\phi_i^0)^{-1} \left( \frac{1}{2}(s\xi - \eta) \right)$$

(73b)

as well as

$$z^{P}((\xi, \eta)) = (\phi_i^0)^{-1} \left( \frac{1}{2}(s\xi + \eta) + \frac{1}{2}(1-s)\phi_i^0(1) \right)$$

(74a)

$$z^{Q}((\xi, \eta)) = (\phi_i^0)^{-1} \left( \frac{1}{2}(s\xi - \eta) + \frac{1}{2}(1-s)\phi_i^0(1) \right)$$

(74b)

respectively, are introduced along with the new kernel elements

$$D((\xi, \eta)) = D_{ij}(\xi^i((z, \zeta)), \eta^j((z, \zeta))) = \lambda_j'(\zeta)P(z, \zeta)$$

(75a)

$$G((\xi, \eta)) = G_{ij}(\xi^i((z, \zeta)), \eta^j((z, \zeta))) = \lambda_j'(\zeta)Q(z, \zeta).$$

(75b)

With the change of coordinates (71a), (71b), the boundaries of the original Fredholm-domain of $P(z, \zeta)$ are mapped as $(z, 0) \rightarrow (\eta, \eta), (0, \zeta) \rightarrow (\eta, -\eta)$,

$$(z, 1) \rightarrow (\xi, \xi - 2\phi_j^0(1)) \text{ and } (1, \zeta) \rightarrow (\xi, 2\phi_j^0(1) - \xi).$$

Hence the lower boundary

$$\eta(\xi) = \eta^P(\xi) = \begin{cases} -\xi, & \xi \leq \phi_j^0(1) \\ \xi - 2\phi_j^0(1), & \xi > \phi_j^0(1) \end{cases}$$

(76)

consists of two parts in this case. Of course, the same is valid for the left boundary

$$\xi(\eta) = \xi^P(\eta) = \begin{cases} \eta, & \eta \geq 0 \\ -\eta, & \eta < 0 \end{cases}$$

(77)

The resulting spatial domains for $D$ and $G$ are depicted in Figure 3. Due to the fact that (70a) only contains the diffusion coefficients of the right subsystem, the result $\lambda_j^r \leq \lambda_j^l$ automatically holds for $i \leq j$, which is why specifying the matrices $A_j^r(z)$, $i = 0, 1$, as strictly lower triangular according to (68) leads to a solvable BVP for $Q(z, \zeta)$. To ensure the same shape of the spatial domain for all $i, j$ (see Figure 2), $s_{ij}$ according to (39) is introduced in the transformation (71c), (71d) as it was applied in (37). In contrast to that, (65a) contains $\lambda_j^r$. Thus, in this case, $\lambda_i^r < \lambda_i^l$ for all $i, j = 1, \ldots, n$ due to the assumed sorting of the diffusion coefficients. Hence the resulting spatial domains for all kernel elements $D$, i.e., the Fredholm part, already have a similar shape. Moreover, due to the types of BCs, it can be seen from the transformation into integral equations that the simple change of coordinates (71c), (71d) is sufficient for the Fredholm kernel.

Together, (71), (75) lead to the canonical kernel equations

$$D_{\xi,\eta}(\xi, \eta) = -\frac{\Delta^P}{A_D}\frac{D_{\xi}(\xi, \eta)}{D_{\eta}(\xi, \eta)} + \frac{\Delta^Q}{A_D}\frac{D_{\eta}(\xi, \eta)}{D_{\xi}(\xi, \eta)}$$

(78a)

$$D(\eta, -\eta) = 0$$

(78b)

$$D_{\eta}(\eta, -\eta) = 0$$

(78c)

$$D_{\xi}(\eta, -\eta) = 0$$

(78d)

$$[D_{\xi}(\xi, \xi - 2\phi_j^0(1)) = D_{\eta}(\xi, \xi - 2\phi_j^0(1))]$$

(78e)

and

$$G_{\xi,\eta}(\xi, \eta) = -\frac{\Delta^P}{A_G}\frac{G_{\xi}(\xi, \eta)}{G_{\eta}(\xi, \eta)} + \frac{\Delta^Q}{A_G}\frac{G_{\eta}(\xi, \eta)}{G_{\xi}(\xi, \eta)}$$

(79a)

$$G(\xi, \eta) = 0$$

(79b)

$$G_{\xi}(\xi, \eta) = 0$$

(79c)

$$[G_{\eta}(\xi, \eta) = 0 \eta < 0$$

(79d)

respectively, are introduced along with the new kernel elements

$$D(\eta, \eta) = \frac{1}{\Delta^P}G(\eta, \eta) + A_j^r(z^P(\eta, \eta))$$

(78a)

$$D(\eta, -\eta) = 0$$

(78b)

$$D(\eta, -\eta) = 0$$

(78c)

$$D(\eta, -\eta) = 0$$

(78d)

$$[D(\xi, \xi - 2\phi_j^0(1)) = D(\xi, \xi - 2\phi_j^0(1))]$$

(78e)

and

$$G(\xi, \eta) = 0$$

(79b)

$$G(\xi, \eta) = 0$$

(79c)

$$[G_{\eta}(\xi, \eta) = 0 \eta < 0$$

(79d)

respectively, are introduced along with the new kernel elements

$$i \leq j :$$

$$D(\eta, \eta) = \frac{1}{\Delta^P}G(\eta, \eta) + A_j^r(z^P(\eta, \eta))$$

(78a)

$$D(\eta, -\eta) = 0$$

(78b)

$$D(\eta, -\eta) = 0$$

(78c)

$$D(\eta, -\eta) = 0$$

(78d)

$$[D(\xi, \xi - 2\phi_j^0(1)) = D(\xi, \xi - 2\phi_j^0(1))]$$

(78e)

and

$$G(\xi, \eta) = 0$$

(79b)

$$G(\xi, \eta) = 0$$

(79c)

$$[G_{\eta}(\xi, \eta) = 0 \eta < 0$$

(79d)
Fig. 3. The spatial domains of the canonical kernel equations (78), (79) and (83) in the canonical coordinate systems \((\xi, \eta)\) for \(i \leq j\) with the utilized directions of integration. The spatial domains are characterized by \(b_r^i = \phi_i^r(1) - \phi_j^r(1), b_r^f = \phi_i^r(1) + \phi_j^r(1), b_-^r = \phi_i^r(1) - \phi_j^r(1), b_-^f = \phi_i^r(1) + \phi_j^r(1)\). In the left picture, the lower boundary \(\eta_0(\xi)\) is given by (76). In the right picture, it is determined the same way as in Figure 2. The thick coloured lines represent the BCs and the coloured arrows the respective directions of integration for the corresponding variables. Herein, the blue and red arrows symbolize the first and second integration, respectively. For the left elements, the BC for \(J(\eta, \eta)\) depends on \(H(\eta, \eta)\), for which the respective integral equation can be substituted, which is symbolized by the green arrow. On the other hand, \(J(\eta, \eta)\) depends on \(M(\eta, \eta)\) for which again the integral equation can be substituted.

\[-\int_0^{z^P(\eta, \eta)} \frac{1}{\lambda^r_j(\zeta)} G_{ik}(\xi^Q_k, \eta^Q_k, (z^P(\eta, \eta)), \zeta) \tilde{A}_{ik}^r(\zeta) \, d\zeta + \int_0^{1} \frac{1}{\lambda^r_j(\zeta)} D_{ik}(\xi^Q_k, \eta^Q_k, (z^Q(\eta, \eta)), \zeta) \tilde{A}_{ik}^r(\zeta) \, d\zeta \quad (80a)\]

\[G_\xi(\eta, \eta) - G_\eta(\eta, \eta) = -\tilde{z}_0(D_\xi(\eta, \eta) - D_\eta(\eta, \eta)) + \int_0^{z^Q(\eta, \eta)} \frac{\sqrt{\lambda^r_i(0)}}{\lambda^r_j(\zeta)} G_{ik}(\xi^Q_k, \eta^Q_k, (z^Q(\eta, \eta)), \zeta) \tilde{A}_{ik}^r(\zeta) \, d\zeta + \int_0^{1} \frac{\sqrt{\lambda^r_i(0)}}{\lambda^r_j(\zeta)} D_{ik}(\xi^Q_k, \eta^Q_k, (z^Q(\eta, \eta)), \zeta) \tilde{A}_{ik}^r(\zeta) \, d\zeta \]

\[-\sqrt{\lambda^r_j(0)} \tilde{A}_{ik}^r(\zeta) \quad (80b)\]

where

\[\tilde{a}^r_i = \tilde{a}^r_j(\xi, \eta) = \frac{\lambda^r_i(0)}{2\lambda^r_j(\zeta)} - \frac{\lambda^r_j(0)}{\sqrt{\lambda^r_j(\zeta)}}, \quad \tilde{a}^r_i = \tilde{a}^r_j(\xi, \eta) = \frac{\lambda^r_i(0)}{2\lambda^r_j(\zeta)} + \frac{\lambda^r_j(0)}{\sqrt{\lambda^r_j(\zeta)}} \quad (81a)\]

and \((z, \zeta)\) are substituted by (73) as well as

\[\tilde{a}^r_i = \tilde{a}^r_j(\xi, \eta) = \frac{\lambda^r_i(0)}{2\lambda^r_j(\zeta)} - \frac{\lambda^r_j(0)}{\sqrt{\lambda^r_j(\zeta)}}, \quad \tilde{a}^r_i = \tilde{a}^r_j(\xi, \eta) = \frac{\lambda^r_i(0)}{2\lambda^r_j(\zeta)} + \frac{\lambda^r_j(0)}{\sqrt{\lambda^r_j(\zeta)}} \quad (82a)\]

in which \((z, \zeta)\) are substituted by (74). To simplify the notation, the summation convention \(\sum_{k=1}^{n}(c_k A_{ik} B_{kj}) := c_k A_{ik} B_{kj}\) is introduced in (80), which means that all expressions having \(k\) as an index, are summed from 1 to \(n\). To be able to convert the canonical kernel equations into integral equations, the coupling BCs (80) require an additional reformulation. To this end, (80a) is differentiated w.r.t. \(\eta\). The result is utilized in (80b) to obtain the BCs

\[D_\eta(\eta_0, \eta) = \frac{1}{\tilde{z}_0} G_\xi(\eta, \eta) - \frac{1}{\tilde{z}_0} \int_0^{z} \frac{\sqrt{\lambda^r_i(0)}}{\lambda^r_j(\zeta)} G_{ik} \tilde{A}_{ik}^r(\zeta) \, d\zeta \]

\[-\frac{1}{2\tilde{z}_0} \int_0^{1} \frac{\sqrt{\lambda^r_i(0)}}{\lambda^r_j(\zeta)} D_{ik} \tilde{A}_{ik}^r(\zeta) \, d\zeta \]

\[-\int_0^{z} \frac{1}{2\lambda^r_j(\zeta)} (s_{ik} g_{ik, \xi} + G_{ik, \eta} A_{ik}^r(\zeta) \, d\zeta \]

\[-\int_0^{1} \frac{1}{2\lambda^r_j(\zeta)} (D_{ik} \xi + D_{ik} \eta) A_{ik}^r(\zeta) \, d\zeta \]

\[-\frac{\sqrt{\lambda^r_i(0)}}{2\tilde{z}_0} A_{ik}^r(z) + \frac{\sqrt{\lambda^r_j(0)}}{2\tilde{z}_0} (\tilde{A}_{ik}^r)^r(z) \quad (83a)\]

\[G_\eta(\eta_0, \eta) = \tilde{z}_0 D_\xi(\eta, \eta) - \int_0^{z} \frac{\sqrt{\lambda^r_i(0)}}{2\lambda^r_j(\zeta)} G_{ik} \tilde{A}_{ik}^r(\zeta) \, d\zeta \]

\[-\int_0^{1} \frac{\sqrt{\lambda^r_i(0)}}{2\lambda^r_j(\zeta)} D_{ik} \tilde{A}_{ik}^r(\zeta) \, d\zeta \]

\[+ \int_0^{z} \frac{\tilde{z}_0}{2\lambda^r_j(\zeta)} (s_{ik} g_{ik, \xi} + G_{ik, \eta} A_{ik}^r(\zeta) \, d\zeta \]

\[+ \int_0^{1} \frac{\tilde{z}_0}{2\lambda^r_j(\zeta)} (D_{ik} \xi + D_{ik} \eta) A_{ik}^r(\zeta) \, d\zeta \]

\[+ \frac{\sqrt{\lambda^r_i(0)}}{2\tilde{z}_0} A_{ik}^r(z) - \frac{\sqrt{\lambda^r_j(0)}}{2\tilde{z}_0} (\tilde{A}_{ik}^r)^r(z). \quad (83b)\]

In (83a) \(z = z^P(\eta, \eta)\), \(G_{ik}\) and its derivatives have the argument \((\xi^Q_k, \eta^Q_k)(z^P(\eta, \eta), \zeta)\) and \(D_{ik}\) and its derivatives are evaluated at \((\xi^Q_k, \eta^Q_k)(z^Q(\eta, \eta), \zeta)\). In (64a) \(z = z^Q(\eta, \eta)\), \(G_{ik}\) and its derivatives have the argument \((\xi^Q_k, \eta^Q_k)(z^Q(\eta, \eta), \zeta)\), and \(D_{ik}\) and its derivatives are evaluated at \((\xi^Q_k, \eta^Q_k)(z^Q(\eta, \eta), \zeta)\).

By mapping \((\tilde{A}_{ik}^r)^r(z)\) in (30b) (see (11)) to canonical coordinates, it can be shown that it is a piecewise
continuous function.

To uniquely determine the kernels, the BCs are complemented by the artificial BCs

\[
\begin{aligned}
&\frac{i > j}{i > j} : \\
&D_\eta(\eta, \eta) = g_D(\eta) \quad \text{(84a)} \\
&G_\eta(\eta, \eta) = g_G(\eta) \quad \text{(84b)}
\end{aligned}
\]

(see (41h), (41i)) with the degrees of freedom \(g_D = g_{D,ij} \in C[0, \phi_i(1)]\) and \(g_G = g_{G,ij} \in C[0, \phi_i(1)]\), \(i, j = 1, \ldots, n\).

### 4.2.2 Kernel integral equations

Similar to Section 4.1.2, the canonical kernel equations (78), (79) with (83), (84) are converted into integral equations by formally integrating the PDEs (78a), (79a) w.r.t. \(\eta\) and \(\eta\). A BC at \((\xi, \eta) = (\eta, \eta)\) for the respective derivative w.r.t. \(\eta\) is available for both \(D\) and \(G\) by (83). Together with (78c), (79d) and (80), \(D_\eta(\eta, \eta, \eta, \eta)\) and \(G_\eta(\eta, \eta, \eta, \eta)\) are known at the whole lower boundary \(\xi(\eta)\) of the respective spatial domain (see Figure 3).

To get the BC for \(D\) and \(G\) on the lower boundary, note that \(D'(\xi, \xi - 2\phi_j'(1)) = D_\xi(\xi, \xi - 2\phi_j'(1)) + D_\eta(\xi, \xi - 2\phi_j'(1))\). Inserting (78e) and integrating w.r.t. \(\xi\) then yields

\[
\begin{aligned}
[D(\xi, \xi - 2\phi_j'(1)) &= D(\phi_j'(1), -\phi_j'(1)) \\
&+ \int_{\phi_j'(1)}^\xi 2D_\eta(\xi, \xi - 2\phi_j'(1))d\xi] > \phi_j'(1),
\end{aligned}
\]

Together with (78c) and (79b), BCs \(D(\xi, \xi(\eta))\) and \(G(\xi, \eta(\xi))\) are available at the whole lower boundary \(\eta(\xi)\) of the respective spatial domain (see Figure 3).

Furthermore, note that even for the respective derivatives \(D_\xi\) and \(G_\xi\), a BC on the whole lower boundary of the domains is determined by (78d), (78e) and (78f).

Hence, introducing the variables

\[
\begin{aligned}
N(\xi, \eta) &= N_{ij}(\xi, \eta) := D_\eta(\xi, \eta) \quad \text{(86a)} \\
J(\xi, \eta) &= J_{ij}(\xi, \eta) := G_\eta(\xi, \eta) \quad \text{(86b)}
\end{aligned}
\]

and

\[
\begin{aligned}
M(\xi, \eta) &= M_{ij}(\xi, \eta) := D_\xi(\xi, \eta) \quad \text{(87a)} \\
H(\xi, \eta) &= H_{ij}(\xi, \eta) := G_\xi(\xi, \eta) \quad \text{(87b)}
\end{aligned}
\]

and inserting the BCs (78b)–(78e), (79b)–(79d), (85), (83) and (84), the kernel integral equations finally read

\[
\begin{aligned}
N &= N_0 + F_N[N, D, M, J, G, H] \quad \text{(88a)} \\
D &= D_0 + F_D[N] \quad \text{(88b)} \\
M &= M_0 + F_M[N, D, M, J, G, H] \quad \text{(88c)} \\
J &= J_0 + F_J[N, D, M, J, G, H] \quad \text{(88d)} \\
G &= F_G[J] \quad \text{(88e)} \\
H &= F_H[J, H] \quad \text{(88f)}
\end{aligned}
\]

with

\[
\begin{aligned}
N_0 &= [g_D(\eta)]_{i > j} \eta \geq 0 \quad \text{(89a)} \\
&+ \left[ \frac{\sqrt{\lambda_k^i(0)}}{2z^0} \bar{A}_{0,i}^r(z) + \frac{\sqrt{\lambda_k^i(z)}}{2} (\bar{A}_{0,j}^r(z))' \right]_{i \leq j} \eta \geq 0 \\
D_0 &= \left[ \int_{\phi_j'(1)}^\xi 2N_0(\eta(\xi))d\xi \right]_{\xi > \phi_j'(1)} \quad \text{(89b)} \\
M_0 &= [\{[N_0(\eta(\xi))\}_{\xi > \phi_j'(1)}] \quad \text{(89c)} \\
J_0 &= [g_G(\eta)]_{i > j} \eta \geq 0 \quad \text{(89d)} \\
&+ \left[ \frac{\sqrt{\lambda_k^i(0)}}{2} \bar{A}_{0,i}^r(z) + \frac{\sqrt{\lambda_k^i(z)}}{2} (\bar{A}_{0,j}^r(z))' \right]_{i \leq j} \eta \geq 0
\end{aligned}
\]

and

\[
\begin{aligned}
F_N &= \int_{\eta(\xi)}^{\eta} N(\xi, \eta)\eta d\eta \\
F_D &= \int_{\eta(\xi)}^{\eta} D(\xi, \eta)\eta d\eta \\
F_M &= \int_{\eta(\xi)}^{\eta} M(\xi, \eta)\eta d\eta \\
F_J &= \int_{\eta(\xi)}^{\eta} J(\xi, \eta)\eta d\eta \\
F_G &= \int_{\eta(\xi)}^{\eta} G(\xi, \eta)\eta d\eta \\
F_H &= \int_{\eta(\xi)}^{\eta} H(\xi, \eta)\eta d\eta
\end{aligned}
\]
Fredholm integrals

\[ + \int_0^1 \frac{z_0}{2\lambda_k^2(\xi)} (M_{ik} + N_{ik}) \tilde{A}_{l,kj}(\xi) \, d\xi \quad i \leq j \quad z = z^Q(\eta, \eta) \]

\[ \int_0^\xi \mathcal{A}_G[H, J](\xi, \eta) \, d\xi \quad (90d) \]

\[ F_G[J] = \int_{n(\xi)}^\eta J(\xi, \eta) \, d\eta \quad (90e) \]

\[ F_H[J, H] = \int_{n(\xi)}^\eta \mathcal{A}_G[H, J](\xi, \eta) \, d\eta. \quad (90f) \]

Therein, the integral equation (88a) has been utilized in (78e) and (85). Moreover, the arguments of \( D_{ik} \) and its derivatives as well as \( G_{ik} \) and its derivatives are the same as in (83).

4.3 Successive approximation

The integral equations (61) with (62) and (63) as well as (88) with (89) and (90) are of a similar form as in [8] and can now be solved by the method of successive approximations, i.e., by applying fixed-point iteration. Thereby, the solutions may be represented by

\[ G = \sum_{l=0}^{\infty} \Delta G_l, \quad H = \sum_{l=0}^{\infty} \Delta H_l, \quad J = \sum_{l=0}^{\infty} \Delta J_l \quad (91) \]

\[ D = \sum_{l=0}^{\infty} \Delta D_l, \quad M = \sum_{l=0}^{\infty} \Delta M_l, \quad N = \sum_{l=0}^{\infty} \Delta N_l \quad (92) \]

with \( \Delta G_l = [\Delta G_{ij}^l], \Delta H_l = [\Delta H_{ij}^l] \) and \( \Delta J_l = [\Delta J_{ij}^l] \), \( i, j = 1, \ldots, 2n \) in the case of (61) and \( i, j = 1, \ldots, n \) in the case of (88) as well as \( \Delta D_l = [\Delta D_{ij}^l], \Delta M_l = [\Delta M_{ij}^l] \) and \( \Delta N_l = [\Delta N_{ij}^l] \), \( i, j = 1, \ldots, n \), which are calculated by the update law

\[ \Delta G_l^{i+1} = F_G[\Delta H_l, \Delta J_l], \quad \Delta G_0 = G_0(\xi, \eta) \quad (93a) \]

\[ \Delta H_l^{i+1} = F_H[\Delta G^l, \Delta H^l, \Delta J^l], \quad \Delta H_0 = H_0(\xi, \eta) \quad (93b) \]

\[ \Delta J_l^{i+1} = F_J[\Delta G^l, \Delta H^l, \Delta J^l], \quad \Delta J_0 = J_0(\xi, \eta) \quad (93c) \]

in the case of (61) and similar for (88). Of course, the integral equations (61)–(63) and (88)–(90) differ from the form in [8] due to the coupling BCs (26e), (26f) and (83), respectively. However, the types of the appearing terms in the integral operators (63), (90) are the same except for the terms with the Fredholm integrals and the boundary integral \( \int_0^\xi \phi'(l) \) \( \ldots \) \( d\xi \). Noting that inside the Fredholm integrals \( z = z(\eta, \eta) \) always holds allows to determine integral estimates for both terms with a similar reasoning as in [8]. Hence, with the same approach as in the latter reference, absolute and uniform convergence of the series (91) follows by proving

\[ |\Delta X^l(\xi, \zeta, \eta(z, \zeta))| \leq \frac{M_l^{i+1}}{\Pi}(z - \gamma)^i \quad (94) \]

for each \( \Delta X \in \{ \Delta G, \Delta H, \Delta J, \Delta D, \Delta M, \Delta N \} \), some \( M > 0 \) and

\[ \gamma \in \left( \max_{i \geq j} \sqrt{\frac{\lambda_i(z_\Delta)}{\lambda_j(z_\Delta)}}, 1 \right), \quad (95) \]

where \( z_\Delta = \arg\min(|\lambda_i(z) - \lambda_j(z)|) \) is the point of minimal difference between the diffusion coefficients \( \lambda_i \) and \( \lambda_j \) (see [8]). Note that \( \max_{i > j} \sqrt{\lambda_i(z_\Delta)/\lambda_j(z_\Delta)} \leq \max_{i > j} \sqrt{\lambda_i(z_\Delta)}/\lambda_j(z_\Delta) \) holds so that (95) is sufficient for the convergence proof of (88). Therefore, (91) provides the piecewise continuous solution of the integral equations (61) and (88), which proves Theorems 3 and 4.

5 Example

Consider a system (1) consisting of two coupled PDEs with the parameters

\[ A(\zeta) = \begin{bmatrix} z^2 + 2 & 0 \\ 0 & e^{-z} + 1/2 \end{bmatrix} \quad (96a) \]

\[ \tilde{A}(\zeta) = \begin{bmatrix} 1 & 1 + \zeta \\ \frac{1}{2} + \zeta & 1 \end{bmatrix} \quad (96b) \]

subject to Neumann BCs, i.e., \( B_0 = B_1 = 0 \) in (1b) and (1c), which is open-loop unstable. For the bilateral controller design, a numerical analysis of the possible folding points to ensure non-intersection distance coefficients yields the intervals \( \tilde{z}_0 \in I_1, I_1 = (0, 0.412), I_2 = (0.424, 0.488), I_3 = (0.568, 0.596) \) and \( I_4 = (0.604, 1) \). The folding point is set to \( \tilde{z}_0 = 0.325 \), leading to descending diffusion coefficients \( \lambda_1(z) > \ldots > \lambda_4(z) \) in the folded system. After specifying \( \mu = 10 \), the degrees of freedom \( q_1(\eta) \) in the well-posed BCs (41h), (41i) and \( q_D(\eta), g_D(\eta) \) in (84) are set to zero for simplicity, which determines the kernels. The successive approximations (91) are implemented in MATLAB and truncated after the maximum deviation \( \max_{\xi < \xi, \eta} |\Delta X_{ij}^l| \leq 10^{-3} \), \( \Delta X \in \{ \Delta G, \Delta H, \Delta J, \Delta D, \Delta M, \Delta N \} \), which lead to 9 iterations for \( K(z, \zeta) \) and 8 iterations for \( P(z, \zeta) \) and \( Q(z, \zeta) \). The kernels are discretized by 51 points in each direction \( z, \zeta \). In order to limit the number of resulting nodes, the grids \( (\xi, \eta) \) in the canonical coordinates are resampled to 100 nodes in \( \xi \)-direction and the same node distance for \( \eta \).

To evaluate the effect of the bilateral control, the designed state feedback controller (20) is compared with an unilateral backstepping controller according to [8], i.e., \( u_0(t) = 0 \). The left picture of Figure 4 shows the resulting weighted \( L_2 \)-norms \( ||h||_{L_2} = (\int_0^1 |h| \frac{1}{\sqrt{2\pi}} \sin(\pi \xi + \frac{2\pi}{2}) \cos(3\pi \xi + \frac{2\pi}{2}) d\xi)^{1/2} \) of the state \( w(\xi, t) \) with the IC \( w_0(\xi, 0) = \frac{1}{2\pi} \sin(\pi \xi + \frac{2\pi}{2}), i = 1, 2 \), in both cases. It can be seen that the bilateral controller is able to exponentially stabilize the system with the desired decay rate. Moreover, transforming the resulting state profiles into the target coordinates by applying (8) and (13) and comparing with a simulation of the target system (22) shows
that the desired behaviour is achieved with a maximum deviation of $3.7 \cdot 10^{-3}$, occurring solely due to numerics.

The right picture of Figure 4 shows the corresponding control efforts in both cases. Due to the active left input $u_0(t)$ in the bilateral case, the required control effort $u_1(t)$ of the right input to achieve the desired decay rate is significantly reduced.

The upper plots in Figure 5 show the resulting closed-loop profiles $w_1(z, t)$ for the cases $\tilde{z}_0 = 0.16$ and $\tilde{z}_0 = 0.66$. Note that $\tilde{z}_0 = 0.66$ requires a reordering of the states to ensure $\lambda_1 > \ldots > \lambda_4$. Obviously, the choice of the folding point $\tilde{z}_0$ has a serious influence on the spatial and temporal evolution of the state. A similar simulation result is obtained for the corresponding second state $w_2(z, t)$. Moreover, the design parameter $\tilde{z}_0$ can be used to adjust the distribution of the control effort between the inputs $u_0(t)$ and $u_1(t)$. This can be verified in the lower plots of Figure 5, where it is compared in the cases $\tilde{z}_0 = 0.16$ and $\tilde{z}_0 = 0.66$. It can be seen that shifting the folding point to the left increases the control effort for the right input and vice versa. This effect is further investigated in [5].

6 Concluding Remarks

An obvious extension in the case of coupled PDEs is to specify individual folding points for each state. This, however, leads to couplings between different spatial areas of different states in the folded system, hindering the application of the standard backstepping transformation and leading to a very involved problem. As already mentioned, the case of non-sorted diffusion coefficients, where some or all of them are equal can be included straightforwardly. Moreover, the observer design for coupled parabolic PDEs with in-domain measurements like in [5] can be realized by making use of the ideas in [9]. The extension to time-varying reaction coefficients with the results in [15] poses no new challenges. Furthermore, the results of the paper allow to investigate more sophisticated systems like parabolic PDEs with couplings to an ODE at an in-domain point.

Fig. 4. Comparison of the bilateral with an unilateral backstepping controller. The left picture shows the $L_2$-norms of the closed-loop system in the bilateral and unilateral cases. The corresponding control efforts are depicted in the right picture, where the solid lines belong to the first element of the input $w_i(t)$, $i = 0, 1$, and the dashed lines to their second element.

Fig. 5. Profiles of the first state $w_1(z, t)$ of the closed-loop system (1) with (20) for the choices $\tilde{z}_0 = 0.16$ and $\tilde{z}_0 = 0.66$, which are marked by the red lines and the corresponding control efforts $u_0(t)$ and $u_1(t)$ for $\tilde{z}_0 = 0.16$ (green) and $\tilde{z}_0 = 0.66$ (blue). The solid lines correspond to the first element of the input $u_i(t)$, $i = 0, 1$, and the dashed lines to their second element.

Combining the folding technique with the methods for output regulation according to [9] and the internal model principle [10] to achieve robust output regulation in the bilateral case are of special interest for future work. Moreover, the bilateral backstepping approach opens the door to fault tolerant designs. For example, a bilateral setup can be rendered fault tolerant by switching to a stabilizing unilateral controller. It would also be interesting to investigate the input to state stability properties of bilateral designs w. r. t. boundary disturbances.

References

[1] B. Atkinson. Biochemical Reactors. Pion Limited, London, 1974.
[2] A. Baccoli, A. Pisano, and Y. Orlov. Boundary control of coupled reaction-diffusion processes with constant parameters. Automatica, 54:80–90, 2015.
[3] N. Bekiaris-Liberis and R. Vazquez. Nonlinear bilateral output-feedback control for a class of viscous Hamilton-Jacobi PDEs. Automatica, 101:223–231, 2018.
[4] S. Chen, R. Vazquez, and M. Krstic. Folding backstepping approach to parabolic PDE bilateral boundary control. IFAC-PapersOnLine, 52:76–81, 2019.
[5] S. Chen, R. Vazquez, and M. Krstic. Folding bilateral backstepping output-feedback control design for an unstable parabolic PDE. Submitted to IEEE Trans. Autom. Control, available at arXiv:1906.05434v2 [math.OC], 2019.
[6] P. Christofides. Model-based Control of Particulate Processes. Kluwer Academic Publishers, Dordrecht, 2002.
[7] R. F. Curtain and H. J. Zwart. *An Introduction to Infinite-Dimensional Linear Systems Theory*. Springer-Verlag, New York, 1995.

[8] J. Deutscher and S. Kerschbaum. Backstepping control of coupled linear parabolic PDEs with spatially varying coefficients. *IEEE Trans. Autom. Control*, 63:4218–4233, 2018.

[9] J. Deutscher and S. Kerschbaum. Output regulation for coupled linear parabolic PDEs. *Automatica*, 100:360–370, 2019.

[10] J. Deutscher and S. Kerschbaum. Robust output regulation by state feedback control for coupled linear parabolic PDEs. *IEEE Trans. Autom. Control*, 65:2207–2214, 2020.

[11] G. Freudenthaler, F. Götzsch, and T. Meurer. Backstepping-based extended Luenberger observer design for a Burgers-type PDE for multi-agent deployment. *IFAC-PapersOnLine*, 50:6780–6785, 2017.

[12] G. Freudenthaler and T. Meurer. PDE-based multi-agent formation control using flatness and backstepping: Analysis, design and robot experiments. *Automatica*, 115:108897, 2020.

[13] H. Jakobsen. *Chemical Reactor Modeling — Multiphase Reactive Flows*. Springer-Verlag, Berlin, 2014.

[14] T. Kato. *Perturbation Theory for Linear Operators*. Springer-Verlag, Berlin, 1995.

[15] S. Kerschbaum and J. Deutscher. Backstepping control of coupled linear parabolic PDEs with space and time dependent coefficients. *IEEE Trans. Autom. Control*, 65:3060–3067, 2020.

[16] M. Krstic and A. Smyshlyaev. *Boundary Control of PDEs — A Course on Backstepping Designs*. SIAM, Philadelphia, 2008.

[17] Y. Orlov, A. Pisano, A. Pilloni, and E. Usai. Output feedback stabilization of coupled reaction-diffusion processes with constant parameters. *SIAM J. Control Optim.*, 55:4112–4155, 2017.

[18] R. Vazquez and M. Krstic. Bilateral boundary control of one-dimensional first- and second-order PDEs using infinite-dimensional backstepping. In *Proc. IEEE 55th Conference on Decision and Control (CDC)*, pages 537–542, Melbourne, Australia, 2016.

[19] R. Vazquez and M. Krstic. Boundary control of coupled reaction-diffusion systems with spatially-varying reaction. *Proc. IFAC Workshop CPDE*, Bertinoro, Italy, pages 224–229, 2016.

[20] R. Vazquez and M. Krstic. Boundary control of coupled reaction-advection-diffusion systems with spatially-varying coefficients. *IEEE Trans. Autom. Control*, 62:2026–2033, 2017.

[21] R. Vazquez and M. Krstic. Taking a step back: a brief history of PDE backstepping. In *Proc. 20th IFAC World Congress, Toulouse, France*, pages 4258–4262, 2017.

[22] Vazquez, R. and Krstic, M. Explicit output-feedback boundary control of reaction-diffusion PDEs on arbitrary-dimensional balls. *ESAIM: COCV*, 22:1078–1096, 2016.

[23] N. C. A. Wilhelmsen, H. Anfinsen, and O. M. Aamo. Minimum time bilateral observer design for $2 \times 2$ linear hyperbolic systems. In *Proc. 8th European Control Conference (ECC)*, pages 2325–2331, Naples, Italy, 2019.