A Partial Integral Equation Representation of Coupled Linear PDEs and Scalable Stability Analysis using LMIs

Matthew M. Peet *Senior Member, IEEE*

***Abstract*— We present a new Partial Integral Equation (PIE) representation of Partial Differential Equations (PDEs) in which it is possible to use convex optimization to perform stability analysis with little or no conservatism. The first result shows that any set of coupled linear PDEs with interval domains and with any well-posed set of boundary conditions may be converted to this framework and we give formulae for performing this conversion. This leads to a new prima facie representation of the dynamics without the implicit constraints on state present in boundary conditions. The second result is to show that for systems in this partial integral representation, convex optimization may be used to verify stability without discretization. The resulting algorithms are implemented in the Matlab toolbox PIETOOLS, tested on several illustrative examples, compared with previous results, and the code has been posted on Code Ocean. Numerical testing indicates the algorithm can analyze systems of up to 20 coupled PDEs on a desktop computer.**

I. Introduction

Partial Differential Equations (PDEs) are used to model systems where the state depends continuously on both time and secondary independent variables. Common examples of such secondary dependence include space; as in flexible structures (Bernoulli-Euler beams) and fluid flow (Navier-Stokes); or maturation, as in cell populations and predator-prey dynamics.

The most common method for computational analysis of PDEs is to project the infinite-dimensional state onto a finite-dimensional vector space using, e.g. [1]–[3] and to use the existing extensive literature on control of ODEs to test stability and design controllers for the resulting finite-dimensional system. However, such discretization approaches are prone to instability, numerical ill-conditioning and large-dimensional state-spaces.

Work on computational methods for analysis and control of PDEs which do not rely on discretization has been more limited. Perhaps the most well-known computational method for *stabilization* of PDEs without discretization is the backstepping approach to controller synthesis [4], [5]. This approach cannot be used for stability analysis, however. Recently, there has been some work on the use of Linear Matrix Inequalities (LMIs) to find Lyapunov functions for linear and nonlinear PDEs - See [6]–[9]. However, the Lyapunov functions proposed herein are relatively simple and the resulting stability conditions conservative.

Numerous analytic (non-computational) methods have been proposed over the years for analysis of PDEs, including the well-developed literature on Semigroup theory [10]–[12] and the literature on Port-Hamiltonian systems [13] for selecting boundary inputs. However, these methods are not optimization-based - relying on the expertise of the user to propose and test energy metrics.

SOS-based methods for analysis of stability and the input-output properties of PDEs can be found in [14]–[16] and [17]–[20]. While these SOS-based works are relatively accurate, they: 1) Are mostly limited to scalar PDEs; 2) Suffer from high computational complexity; 3) Are mostly ad-hoc, requiring significant effort to extend the results to new PDEs. For example, these methods have never been able to analyze stability of beam or wave equations. The source of the difficulty in using LMIs and SOS for stability analysis of PDEs is that the solution of a PDE is required to satisfy three sets of constraints: the differential equation; the boundary conditions; and continuity constraints. This is in contrast to ODEs, which are defined by bounded linear operators (matrices) and solutions to which need only satisfy a single differential equation.

M. Peet is with Arizona State University `mpeet@asu.edu`
This work was supported by grant NSF # CMMI-1301851 and ONR #N000014-17-1-2117

In the following two illustrations, we attempt to capture some of the difficulties faced when using LMI-based methods for analysis of PDEs.

Illustration 1: First, we consider what happens when we treat a PDE like an ODE. That is, we only consider the differential equation and ignore boundary conditions and continuity constraints. Suppose we are given a PDE with the following differential equation.

$$\begin{aligned}&\mathbf{x}_t(t,s)\\&=A_0(s)\mathbf{x}(t,s)+A_1(s)\mathbf{x}_s(t,s)+A_2(s)\mathbf{x}_{ss}(t,s)\end{aligned}$$

An obvious Lyapunov function for this system is

$$V(\mathbf{x})=\int_a^b \mathbf{x}(s)^T M(s)\mathbf{x}(s)ds.$$

As would be the case for an ODE, $V(x)\geq \epsilon\|x\|^2$ if $M(s)\geq \epsilon I$ for all s and some $\epsilon>0$ - a constraint which is easy to enforce using SOS. However, if we now take the derivative of this Lyapunov function we obtain

$$\dot V(\mathbf{x})=\int_a^b \begin{bmatrix}\mathbf{x}(s)\\ \mathbf{x}_s(s)\\ \mathbf{x}_{ss}(s)\end{bmatrix}^T D(s)\begin{bmatrix}\mathbf{x}(s)\\ \mathbf{x}_s(s)\\ \mathbf{x}_{ss}(s)\end{bmatrix} ds$$

$$D(s):=$$
$$\begin{bmatrix}A_0(s)^TM(s)+M(s)A_0(s) & M(s)A_1(s) & M(s)A_2(s)\\ A_1(s)^TM(s) & 0 & 0\\ A_2(s)^TM(s) & 0 & 0\end{bmatrix}.$$

If we were to treat the system like an ODE, we would constrain $D(s)\leq 0$ and this would imply stability. The problem however, is that $D(s)\not\leq 0$ for ANY choice of $A_1,A_2\neq 0$! The problem is that the differentiation operator branches $\mathbf{x}$ into $\mathbf{x}_s$ and $\mathbf{x}_{ss}$, neither of which are independent of $\mathbf{x}$. Moreover, the information which determines the relationship between $\mathbf{x}$, $\mathbf{x}_s$ and $\mathbf{x}_{ss}$ is not embedded in the differential equation. Rather, this information is implicit in the boundary conditions and continuity constraints. While it may be possible to exploit boundary and continuity properties to obtain a new stability condition using, e.g. integration by parts or Stokes Theorem [20], such secondary steps complicate the analysis and may result in conservative conditions. The goal of this paper, then, is to reformulate the dynamics so that all necessary information on continuity and boundary conditions is represented in the differential equation.

Illustration 2: To understand how continuity and boundary conditions can be represented in the dynamics, let us now consider the following non-partial-differential distributed-parameter system.

$$\dot{\mathbf{u}}(t,s)=\mathbf{u}(t,s),\ \mathbf{u}(t,0)=w_1(t),\ \mathbf{u}_s(t,0)=w_2(t)$$

The continuity constraint is $\mathbf{u}\in H^2$. The exogenous functions w_i could result from coupling to an ODE (as in a delayed system). However, for our purposes, they could also be set to zero. The point to observe is that the system is not, prima facie, a PDE or even a distributed parameter system as the dynamics are identical at every point in the domain. However, if we now combine the fundamental theorem of calculus with integration by parts, we obtain a very different set of dynamics.

$$\dot{\mathbf{u}}(t,s)=sw_1(t)+w_2(t)+\int_0^s(s-\eta)\mathbf{u}_{ss}(\eta)d\eta$$

This formulation of the same system directly incorporates continuity and boundary conditions into the dynamics - which are now expressed using the state $\mathbf{u}_{ss}$, a state which has no continuity properties or boundary conditions. An interesting feature of this representation is that continuity of the original state ($\mathbf{u}\in H^2$) implies that the effect of the boundary conditions are felt instantaneously at every point in the domain.

The goal of this paper, then, is to show that any PDE can be written in this manner and to provide formulae for constructing such a representation. Because the new state is not continuous, however, such a representation does not admit differential operators. Rather, the construction uses the algebra of Partial Integral (3-PI) operators, to be discussed shortly.

A. PIE Representation and Stability Analysis

The class of systems we consider consists of coupled linear PDEs in a single spatial variable, extending the preliminary results in [21]. We write these systems in the form

$$\begin{bmatrix}x_1(t,s)\\x_2(t,s)\\x_3(t,s)\end{bmatrix}_t=A_0(s)\underbrace{\begin{bmatrix}x_1(t,s)\\x_2(t,s)\\x_3(t,s)\end{bmatrix}}_{\mathbf{x}_p}+A_1(s)\begin{bmatrix}x_2(t,s)\\x_3(t,s)\end{bmatrix}_s$$
$$+A_2(s)\left[x_3(t,s)\right]_{ss}$$

where the x_i are **vector**-valued functions $x_i:\mathbb{R}^+\times[a,b]\rightarrow\mathbb{R}^{n_i}$. When putting a PDE in this form, a given scalar state, u, is included in vector x_i as determined by continuity constraints. That is, $u\in x_3$ if $u\in H^2$, $u\in x_2$ if $u\in H^1$ and $u\in x_1$ if $u\in L_2$.

Given this assignment, we may write the vector of all possible boundary conditions as

$$B \begin{bmatrix} x_2(t,a) \\ x_2(t,b) \\ x_3(t,a) \\ x_3(t,b) \\ x_{3s}(t,a) \\ x_{3s}(t,b) \end{bmatrix} = 0.$$

If the Cauchy problem is well-defined, then B is of row rank n_2+2n_3. For convenience, we refer to $\mathbf{x}_p : \mathbb{R}^+ \times [a,b] \to \mathbb{R}^{n_1+n_2+n_3}$ with $\mathbf{x}_p(t) \in L_2[a,b]^{n_1} \times H^1[a,b]^{n_2} \times H^2[a,b]^{n_3}$ as the *primary state* as these are the variables in which the PDE is originally defined. These types of systems arise when there are multiple interacting spatially-distributed states and can be used to define wave equations, beam equations, et c.

The main technical contribution of the paper is to show that if $\mathbf{x}_p$ satisfies the proposed boundary conditions and is suitably differentiable, then both the state and the dynamics may be expressed in terms of the *fundamental state*,

$$\mathbf{x}_f(t,s) = \begin{bmatrix} x_1(t,s) \\ x_{2s}(t,s) \\ x_{3ss}(t,s) \end{bmatrix}$$

as a Partial Integral Equation (PIE) [22]

$$\mathcal{T}\dot{\mathbf{x}}_f(t) = \mathcal{A}\mathbf{x}_f(t)$$

in the state $\mathbf{x}_f$, where $\mathbf{x}_p$ may be recovered as

$$\mathbf{x}_p(t) = \mathcal{T}\mathbf{x}_f(t)$$

and where $\mathcal{T}$ and $\mathcal{A}$ are Partial Integral (3-PI) operators of the form

$$\begin{aligned}\left(\mathcal{P}_{\{N_0,N_1,N_2\}}\mathbf{x}\right)(s) &:= N_0(s)x(s)ds \\ &+ \int_a^s N_1(s,\theta)x(\theta)d\theta + \int_s^b N_2(s,\theta)x(\theta)d\theta,\end{aligned}$$

where $\mathcal{T}$ and $\mathcal{A}$ are uniquely determined by the matrix B and the matrix-valued functions A_i. In the PIE formulation, $\mathbf{x}_f(t) \in L_2[a,b]^{n_1+n_2+n_3}$ need not satisfy any boundary constraints or continuity constraints in order to define a solution.

One of the advantages of using a PIE representation of the PDE is that 3-PI operators are bounded and form an algebra, being closed under composition, addition, scalar multiplication and adjoint. This means that many of the results developed for state-space representations of ODEs carry over to analysis of PIEs. Specifically, because we do not need to account for implicit state constraints such as boundary conditions or continuity, it is possible to reformulate LMIs developed for ODEs as Linear Operator Inequalities (LOIs) defined using 3-PI operators.

In this paper, we focus on development of a LOI for stability analysis. To this end, we propose a Lyapunov function or energy metric of the form

$$V(\mathbf{x}_f) = \langle \mathcal{T}\mathbf{x}_f, \mathcal{P}\mathcal{T}\mathbf{x}_f \rangle$$

where $\mathcal{P}$ is a 3-PI operator. The derivative of this Lyapunov function is then

$$\begin{aligned}\dot{V}(\mathbf{x}_f) &= \langle \mathbf{x}_f, \mathcal{T}^*\mathcal{P}\mathcal{A}\mathbf{x}_f \rangle + \langle \mathbf{x}_f, \mathcal{A}^*\mathcal{P}\mathcal{T}\mathbf{x}_f \rangle \\ &= \langle \mathbf{x}_f, (\mathcal{T}^*\mathcal{P}\mathcal{A} + \mathcal{A}^*\mathcal{P}\mathcal{T})\mathbf{x}_f \rangle .\end{aligned}$$

Then, as formalized in Theorem 7, the exponential stability conditions become existence of a coercive 3-PI operator $\mathcal{P}$ such that

$$T^*\mathcal{P}\mathcal{A} + \mathcal{A}^*\mathcal{P}\mathcal{T} \leq -\delta\mathcal{T}^*\mathcal{T}.$$

This LOI can be converted to an LMI on the coefficients of the polynomials defining the operator $\mathcal{P}$ using the software package PIETOOLS [23].

The paper is summarized as follows. First, in Section III, we define the class of PDEs considered. This class includes almost any suitably well-posed PDE in a single spatial variable with almost any form of boundary conditions. Next, in Section IV, we define the algebra of 3-PI partial integral operators and give formulae for composition and adjoint. In Section V, we show how to convert any PDE of the class in Section III to a PIE. In Section VI, we show that stability of PIEs can be expressed as a LOI defined using 3-PI operators and operator positivity constraints. In Section VII, we show how to enforce 3-PI operator positivity constraints using LMIs on the coefficients of the polynomials which define the 3-PI operators. In Section, IX, we show how to use the Matlab toolbox PIETOOLS to solve the resulting LOIs. In Section X, we compare our algorithm with previous work by using results taken from the literature. This analysis shows significant improvements in accuracy and little or no conservatism when compared to analytic limits or as compared with simulations when no analytic limits are available. The analysis also shows the algorithm is scalable up to at least 20 coupled PDEs. Finally, in Section XI, we apply the algorithm to beam and wave equations for which there are no previous LMI-based stability conditions. This section includes a discussion on converting scalar higher-order PDEs to the proposed state-space representation.

II. Notation

We define $L_2[X]^n$ to be space of $\mathbb{R}^n$-valued Lesbegue integrable functions defined on X and equipped with the standard inner product. We use $W^{k,p}[X]^n$ to denote the Sobolev subspace of $L_p[X]^n$ defined as $\{u \in L_p[X]^n : \frac{\partial^q}{\partial x^q} u \in L_p \text{ for all } q \leq k\}$. $H^k := W^{k,2}$. I and 0 are used to denote the identity and zero matrices when the dimension of the matrices is clear from context. I also denotes the indicator function $I : \mathbb{R} \to \{0, 1\}$, defined as

$$I(s) = \begin{cases} 1, & \text{if } s > 0 \\ 0, & \text{otherwise.} \end{cases}$$

III. A Standardized PDE Representation

Before conversion of the PDE to a Partial-Integral Equation, we first represent the PDE using notation which makes the conversion relatively straightforward. Specifically, we partition the states. States in L_2 are assigned to $x_1(t) \in L_2[a,b]^{n_1}$. These states admit no boundary conditions or partial spatial derivatives. States in H^1 are assigned to $x_2(t) \in H^1[a,b]^{n_2}$ and admit boundary conditions and first-order spatial derivatives. States in H^2 are assigned to $x_3(t) \in H^2[a,b]^{n_3}$ and admit boundary conditions and second-order spatial derivatives. We do not consider states in H^k where $k > 2$, as there are relatively few examples of such systems. However, the results of the paper can be readily extended to such systems. Any admissable coupled inhomogeneous PDE in a single spatial variable may now be expressed using matrix-valued functions A_0, A_1, A_2 as

$$\underbrace{\begin{bmatrix} x_1(t,s) \\ x_2(t,s) \\ x_3(t,s) \end{bmatrix}_t}_{\dot{\mathbf{x}}_p(t)} = A_0(s) \begin{bmatrix} x_1(t,s) \\ x_2(t,s) \\ x_3(t,s) \end{bmatrix} + A_1(s) \begin{bmatrix} x_2(t,s) \\ x_3(t,s) \end{bmatrix}_s + A_2(s) \begin{bmatrix} x_3(t,s) \end{bmatrix}_{ss}. \quad (1)$$

Any boundary condition on $\mathbf{x}_p$ may now be expressed in the form

$$B \begin{bmatrix} x_2(t,a) \\ x_2(t,b) \\ x_3(t,a) \\ x_3(t,b) \\ x_{3s}(t,a) \\ x_{3s}(t,b) \end{bmatrix} = 0. \quad (2)$$

The matrix B must have row rank $n_2 + 2n_3$ in order for the solution to be uniquely defined. We further assume that none of the boundary conditions are of the form $x(a) = x(b)$, $x(a) + (b-a)x_s(a) = x(b)$, or $x_s(a) = x_s(b)$, as it can be shown that these three cases imply integral constraints on x_2, x_3, and x_3, respectively. The use of integral constraints on the state is not included in this manuscript due to space limitations.

A. Euler-Bernoulli Beam Example

To better illustrate this standardized notation, we consider the cantilevered Euler-Bernoulli beam:

$$\begin{aligned} &u_{tt}(t,s) = -c u_{ssss}(t,s), \qquad \text{where} \\ &u(0) = u_s(0) = u_{ss}(L) = u_{sss}(L) = 0. \end{aligned}$$

Our first step is to eliminate the second order time-derivative, u_{tt}, so we create an augmented state $u_1 = u_t$. Next, we would like to eliminate the fourth-order spatial derivative, so we create the augmented state $u_2 = u_{ss}$. Taking the time-derivative of these states, we obtain

$$\begin{aligned} \dot{u}_1 &= u_{tt} = -c u_{ssss} = -c u_{2ss} \\ \dot{u}_2 &= u_{tss} = u_{1ss}. \end{aligned}$$

These equations are now in the universal form

$$\mathbf{x}_t = \underbrace{\begin{bmatrix} 0 & -c \\ 1 & 0 \end{bmatrix}}_{A_2} \mathbf{x}_{ss}$$

where $A_0 = A_1 = 0$, $n_3 = 2$, and $n_1 = n_2 = 0$. We now examine the boundary conditions using these states:

$$u_{ss}(L) = u_2(L) = 0 \quad \text{and} \quad u_{sss}(L) = u_{2s}(L) = 0.$$

These boundary conditions are insufficient, as the resulting rank is 2. However, we may differentiate boundary conditions in time to obtain

$$u_t(0) = u_1(0) = 0 \quad \text{and} \quad u_{ts}(0) = u_{1s}(0) = 0.$$

We now have 4 boundary conditions, which we use to construct the B matrix as

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{B} \begin{bmatrix} u_1(0) \\ u_2(0) \\ u_1(L) \\ u_2(L) \\ u_{1s}(0) \\ u_{2s}(0) \\ u_{1s}(L) \\ u_{2s}(L) \end{bmatrix} = 0.$$

The B matrix is now of rank $4 = n_2 + 2n_3$.

IV. The Alegbra of 3-PI Operators

The goal of this paper is to construct an algebraic representation of coupled PDEs as defined in Section III. To construct such a representation, we define an algebra of bounded operators on L_2. This is the 3-PI algebra of Partial Integral (PI) operators. Any element of this class is parameterized by three bounded matrix-valued functions $N_0 : [a,b] \to \mathbb{R}^{n\times n}$, $N_1 : [a,b]^2 \to \mathbb{R}^{n\times n}$, and $N_2 : [a,b]^2 \to \mathbb{R}^{n\times n}$. For given N_0, N_1, N_2, we use $\mathcal{P}_{\{N_0,N_1,N_2\}} : L_2^n[a,b] \to L_2^n[a,b]$ to denote the corresponding PI operator of the form

$$\left(\mathcal{P}_{\{N_0,N_1,N_2\}}\mathbf{x}\right)(s) := N_0(s)x(s) + \int_a^s N_1(s,\theta)x(\theta)d\theta + \int_s^b N_2(s,\theta)x(\theta)d\theta,$$

where N_0 defines a multiplier operator and N_1, N_2 define the kernel of an integral operator. When clear from context, we typically use the shorthand notation $\mathcal{P}_{\{N_i\}}$ to indicate $\mathcal{P}_{\{N_0,N_1,N_2\}}$. One may consider such bounded operators to be an extension of matrices, wherein N_0 defines the diagonal of the matrix, N_1 contains the sub-diagonal terms, and N_2 contains the terms above the diagonal. In the following subsections, we show that this class of operators is closed under composition and adjoint (closure under addition follows immediately from addition of parameters). Furthermore, we simultaneously define the sub-algebra of 3-PI operators with polynomial parameters N_0, N_1, and N_2.

A. Composition of 3-PI operators

In this subsection, we derive a formula for the composition 3-PI operators. Specifically, we have the following.

Theorem 1: For any bounded functions $B_0, N_0 : [a,b] \to \mathbb{R}^{n\times n}$, $B_1, B_2, N_1, N_2 : [a,b]^2 \to \mathbb{R}^{n\times n}$, we have

$$\mathcal{P}_{\{R_i\}} = \mathcal{P}_{\{B_i\}}\mathcal{P}_{\{N_i\}}$$

where

$$\begin{aligned}
&R_0(s) = B_0(s)N_0(s) \\
&R_1(s,\theta) = B_0(s)N_1(s,\theta) + B_1(s,\theta)N_0(\theta) \\
&+ \int_a^\theta B_1(s,\xi)N_2(\xi,\theta)d\xi + \int_\theta^s B_1(s,\xi)N_1(\xi,\theta)d\xi \\
&+ \int_s^b B_2(s,\xi)N_1(\xi,\theta)d\xi \\
&R_2(s,\theta) = B_0(s)N_2(s,\theta) + B_2(s,\theta)N_0(\theta) \\
&+ \int_a^s B_1(s,\xi)N_2(\xi,\theta)d\xi + \int_s^\theta B_2(s,\xi)N_2(\xi,\theta)d\xi \\
&+ \int_\theta^b B_2(s,\xi)N_1(\xi,\theta)d\xi
\end{aligned} \tag{3}$$

Proof: The proof of this theorem can be found in the Appendix ■

This theorem proves that both the class of 3-PI operators and the sub-class of 3-PI operators with polynomial parameters is closed under composition.

B. The Adjoint of 3-PI operators

Next, we give a formula for the adjoint of a 3-PI operator.

Lemma 2: For any bounded functions $N_0 : [a,b] \to \mathbb{R}^{n\times n}$, $N_1, N_2 : [a,b]^2 \to \mathbb{R}^{n\times n}$ and any $\mathbf{x}, \mathbf{y} \in L_2^n[a,b]$, we have

$$\left\langle \mathcal{P}_{\{N_i\}}\mathbf{x}, \mathbf{y}\right\rangle_{L_2} = \left\langle \mathbf{x}, \mathcal{P}_{\{\hat{N}_i\}}\mathbf{y}\right\rangle_{L_2}$$

where

$$\begin{aligned}
\hat{N}_0(s) &= N_0(s)^T, \quad \hat{N}_1(s,\eta) = N_2(\eta,s)^T \\
\hat{N}_2(s,\eta) &= N_1(\eta,s)^T
\end{aligned} \tag{4}$$

Proof: The proof of this minor lemma can also be found in the Appendix ■

V. Conversion from PDE state to PIE state

In this section, we show that there is a one-to-one map between the space

$$X = \left\{ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in L_2 \times H^1 \times H^2 \ : \ B\begin{bmatrix} x_2(a) \\ x_2(b) \\ x_3(a) \\ x_3(b) \\ x_{3s}(a) \\ x_{3s}(b) \end{bmatrix} = 0 \right\}$$

and the space $L_2^{n_1+n_2+n_3}$. Specifically, we define G_i such that if

$$\mathbf{x}_p = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \in X \qquad \text{and} \qquad \mathbf{x}_f = \begin{bmatrix} x_1 \\ x_{2s} \\ x_{3ss} \end{bmatrix}$$

then

$$\mathbf{x}_p = \mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f$$

where the G_i are uniquely determined by the matrix B. The map from $\mathbf{x}_p$ to $\mathbf{x}_f$ is simply differentiation.

$$G_0(s) = \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad G_1(s,\theta) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & (s-\theta)I \end{bmatrix} + G_2(s,\theta),$$
$$G_2(s,\theta) = -K(s)(BT)^{-1}BQ(s,\theta)$$
$$G_3(s) = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}, \qquad G_4(s,\theta) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & I \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & (s-\theta)I \end{bmatrix} + G_5(s,\theta),$$
$$G_5(s,\theta) = -V(BT)^{-1}BQ(s,\theta) \tag{5}$$

$$T = \begin{bmatrix} I & 0 & 0 \\ I & 0 & 0 \\ 0 & I & 0 \\ 0 & I & (b-a)I \\ 0 & 0 & I \\ 0 & 0 & I \end{bmatrix}, \quad Q(s,\theta) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \\ 0 & 0 & (b-\theta)I \\ 0 & 0 & 0 \\ 0 & 0 & I \end{bmatrix}, \quad K(s) = \begin{bmatrix} 0 & 0 & 0 \\ I & 0 & 0 \\ 0 & I & (s-a) \end{bmatrix}, \quad V = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & I \end{bmatrix}.$$

A. The map from $\mathbf{x}_f$ to $\mathbf{x}_p$

First, we establish the auxiliary identities:

Lemma 3: Suppose that $\mathbf{x} : [a,b] \to \mathbb{R}^n$ is twice continuously differentiable. Then

$$x(s) = x(a) + \int_a^s x_s(\eta)d\eta$$
$$x_s(s) = x_s(a) + \int_a^s x_{ss}(\eta)d\eta$$
$$x(s) = x(a) + x_s(a)(s-a) + \int_a^s (s-\eta)x_{ss}(\eta)d\eta.$$

Proof: The first two identities are the fundamental theorem of calculus. The third identity is a repeated application of the fundamental theorem of calculus, combined with a change of variables. That is,

$$\begin{aligned} x(s) &= x(a) + \int_a^s x_s(\eta)d\eta \\ &= x(a) + \int_a^s x_s(a)ds + \int_a^s \int_a^\eta x_{ss}(\zeta)d\zeta d\eta. \end{aligned}$$

Examining the 3rd term, where $I(s)$ is the indicator function,

$$\begin{aligned} &\int_a^s \int_a^\eta x_{ss}(\zeta)d\zeta d\eta \\ &= \int_a^b \int_a^b I(s-\eta)I(\eta-\zeta)x_{ss}(\zeta)d\zeta d\eta \\ &= \int_a^b \left(\int_a^b I(s-\eta)I(\eta-\zeta)d\eta \right) x_{ss}(\zeta)d\zeta \\ &= \int_a^b I(s-\zeta) \left(\int_s^\zeta d\eta \right) x_{ss}(\zeta)d\zeta \\ &= \int_a^s (s-\zeta)\, x_{ss}(\zeta)d\zeta \end{aligned}$$

which is the desired result. ∎

As an obvious corollary, we have

$$x(b) = x(a) + \int_a^b x_s(\eta)d\eta$$
$$x_s(b) = x_s(a) + \int_a^b x_{ss}(\eta)d\eta$$
$$x(b) = x(a) + x_s(a)(b-a) + \int_a^b (b-\eta)x_{ss}(\eta)d\eta.$$

The implication is that any boundary value can be expressed using two other boundary identities. We can now generalize this to the main result which shows how the boundary conditions influence the map from $\mathbf{x}_f$ to $\mathbf{x}_p$.

Theorem 4: Suppose $\mathbf{x}_p \in L_2 \times H^1 \times H^2$ and

$$B\begin{bmatrix}x_2(a)\\x_2(b)\\x_3(a)\\x_3(b)\\x_{3s}(a)\\x_{3s}(b)\end{bmatrix}=0$$

where B is of row rank n_1+2n_2 and none of the boundary conditions are of the form $x(a)=x(b)$, $x(a)+(b-a)x_s(a)=x(b)$, or $x_s(a)=x_s(b)$. The following identities hold

$$\mathbf{x}_h=\mathcal{P}_{\{G_3,G_4,G_5\}}\mathbf{x}_f \quad \text{and} \quad \mathbf{x}_p=\mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f$$

where

$$\mathbf{x}_p=\begin{bmatrix}x_1\\x_2\\x_3\end{bmatrix}, \qquad \mathbf{x}_h=\begin{bmatrix}x_{2s}\\x_{3s}\end{bmatrix}, \qquad \mathbf{x}_f=\begin{bmatrix}x_1\\x_{2s}\\x_{3ss}\end{bmatrix}$$

and the G_i are as defined in Eqn. (5).

Proof: Let us define the vectors

$$x_{bf}=\begin{bmatrix}x_2(a)\\x_2(b)\\x_3(a)\\x_3(b)\\x_{3s}(a)\\x_{3s}(b)\end{bmatrix}, \quad x_{bc}=\begin{bmatrix}x_2(a)\\x_3(a)\\x_{3s}(a)\end{bmatrix}.$$

Using Lemma 3, we can express x_{bf} using x_{bc} and $\mathbf{x}_f$ as

$$x_{bf}=Tx_{bc}+\mathcal{P}_{\{0,Q,Q\}}\mathbf{x}_f$$

Likewise, we may express $\mathbf{x}_p$ in terms of x_{bc} and $\mathbf{x}_f$ as

$$\mathbf{x}_p=K(s)x_{bc}+\mathcal{P}_{\{L_0,L_1,0\}}\mathbf{x}_f.$$

We may now express the boundary conditions as

$$Bx_{bf}=BTx_{bc}+B\mathcal{P}_{\{0,Q,Q\}}\mathbf{x}_f=0.$$

Now $Im(T)^\perp$ can be represented as

$$\begin{bmatrix}I&-I&0&0&0&0\\0&0&I&-I&I(b-a)&0\\0&0&0&0&I&-I\end{bmatrix}.$$

Since B has row rank n_2+2n_3 and by assumption $Im(B^T)\cap Im(T)^\perp=\emptyset$, BT is invertible and hence

$$\begin{aligned}x_{bc}&=-(BT)^{-1}B\mathcal{P}_{\{0,Q,Q\}}\mathbf{x}_f\\&=-\mathcal{P}_{\{(BT)^{-1}B,0,0\}}\mathcal{P}_{\{0,Q,Q\}}\mathbf{x}_f\\&=-\mathcal{P}_{\{0,(BT)^{-1}BQ,(BT)^{-1}BQ\}}\mathbf{x}_f.\end{aligned}$$

This yields the following expression for $\mathbf{x}_p$.

$$\begin{aligned}\mathbf{x}_p&=\mathcal{P}_{\{K,0,0\}}x_{bc}+\mathcal{P}_{\{L_0,L_1,0\}}\mathbf{x}_f\\&=-\mathcal{P}_{\{K,0,0\}}\mathcal{P}_{\{0,(BT)^{-1}BQ,(BT)^{-1}BQ\}}\mathbf{x}_f+\mathcal{P}_{\{L_0,L_1,0\}}\mathbf{x}_f\\&=-\mathcal{P}_{\{0,K(BT)^{-1}BQ,K(BT)^{-1}BQ\}}\mathbf{x}_f+\mathcal{P}_{\{L_0,L_1,0\}}\mathbf{x}_f\\&=\mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f\end{aligned}$$

Likewise,

$$\begin{aligned}\mathbf{x}_h&=Vx_{bc}+\mathcal{P}_{\{F_0,F_1,0\}}\mathbf{x}_f\\&=-V\mathcal{P}_{\{0,(BT)^{-1}BQ,(BT)^{-1}BQ\}}\mathbf{x}_f+\mathcal{P}_{\{F_0,F_1,0\}}\mathbf{x}_f\\&=\mathcal{P}_{\{F_0,F_1-V(BT)^{-1}BQ,-V(BT)^{-1}BQ\}}\mathbf{x}_f\\&=\mathcal{P}_{\{G_3,G_4,G_5\}}\mathbf{x}_f.\end{aligned}$$

■

These relationships imply that given $\mathbf{x}_f(t)$, we can reconstruct the original variables $\mathbf{x}_p(t)$. Furthermore, since 3-PI operators are bounded, stability with respect to $\mathbf{x}_f$ implies stability with respect to $\mathbf{x}_p$.

Equipped with this conversion formula, we now express the PDE as a Partial Integral Equation (PIE) on the state $\mathbf{x}_f$ using only 3-PI operators.

B. Expression for the Fundamental Dynamics

Now that we have $\mathbf{x}_p=\mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f$, conversion of the PDE to a PIE (Eqn. (7)) is direct.

Lemma 5: If $\mathbf{x}_p(t)$ satisfies

$$\begin{bmatrix}x_1(t,s)\\x_2(t,s)\\x_3(t,s)\end{bmatrix}_t=A_0(s)\overbrace{\begin{bmatrix}x_1(t,s)\\x_2(t,s)\\x_3(t,s)\end{bmatrix}}^{\mathbf{x}_p}+A_1(s)\overbrace{\begin{bmatrix}x_2(t,s)\\x_3(t,s)\end{bmatrix}_s}^{\mathbf{x}_h}+A_2(s)\left[x_3(t,s)\right]_{ss} \tag{6}$$

where $x_i:\mathbb{R}^+\times[a,b]\to\mathbb{R}^{n_i}$ is such that

$$B\begin{bmatrix}x_2(t,a)\\x_2(t,b)\\x_3(t,a)\\x_3(t,b)\\x_{3s}(t,a)\\x_{3s}(t,b)\end{bmatrix}=0 \quad \forall t\geq 0,$$

then $\mathbf{x}_f(t)$ satisfies

$$\mathcal{T}\dot{\mathbf{x}}_f(t)=\mathcal{A}\mathbf{x}_f(t) \tag{7}$$

where

$$\begin{aligned}\mathcal{T}&:=\mathcal{P}_{\{G_0,G_1,G_2\}}, \qquad \mathcal{A}:=\mathcal{P}_{\{H_i\}}\\H_0(s)&=A_0(s)G_0(s)+A_1(s)G_3(s)+A_{20}(s)\\H_1(s,\theta)&=A_0(s)G_1(s,\theta)+A_1(s)G_4(s,\theta),\\H_2(s,\theta)&=A_0(s)G_2(s,\theta)+A_1(s)G_5(s,\theta),\\A_{20}(s)&=\begin{bmatrix}0&0&A_2(s)\end{bmatrix}\end{aligned} \tag{8}$$

where the G_i are as defined in Eqns. (5). Conversely, if $\mathbf{x}_f(t)$ satisfies Eqn. (7), then $\mathbf{x}_p(t):=\mathcal{T}\mathbf{x}_f(t)$ satisfies Eqn. (6).

Proof: By Theorem 4 and Theorem 1 and the definition of the G_i, we have

$$
\begin{aligned}
\dot{\mathbf{x}}_p &= \mathcal{P}_{\{A_0,0,0\}}\mathbf{x}_p + \mathcal{P}_{\{A_1,0,0\}}\mathbf{x}_h + \mathcal{P}_{\{A_{20},0,0\}}\mathbf{x}_f \\
&= \mathcal{P}_{\{A_0,0,0\}}\mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f \\
&\quad + \mathcal{P}_{\{A_1,0,0\}}\mathcal{P}_{\{G_3,G_4,G_5\}}\mathbf{x}_f + \mathcal{P}_{\{A_{20},0,0\}}\mathbf{x}_f \\
&= \mathcal{P}_{\{A_0G_0,A_0G_1,A_0G_2\}}\mathbf{x}_f \\
&\quad + \mathcal{P}_{\{A_1G_3,A_1G_4,A_1G_5\}}\mathbf{x}_f + \mathcal{P}_{\{A_{20},0,0\}}\mathbf{x}_f \\
&= \mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f.
\end{aligned}
$$

Finally, $\dot{\mathbf{x}}_p(t) = \mathcal{T}\dot{\mathbf{x}}_f(t)$. ∎

This representation of the dynamics is useful in that we no longer need to account for the boundary conditions. Note that the PIE in Eqn. (7) has the form of a singular system (often referred to as descriptor or differential-algebraic). However, in our case, the operator $\mathcal{T}$ is invertible through differentiation. To avoid questions of inversion, however, it is often convenient to formulate the dynamics in non-singular form using the time-derivative of $\mathbf{x}_p$ as

$$\dot{\mathbf{x}}_p(t) = \mathcal{A}\mathbf{x}_f(t).$$

Having expressed the dynamics as a PIE, we now proceed to define a Linear Operator Inequality (LOI), whose feasibility guarantees stability of the PIE and hence the PDE.

VI. Lyapunov Stability Conditions

Using the 3-PI parameterization of operators, we may now succinctly represent our Lyapunov stability conditions. The procedure is relatively straightforward.

Theorem 6: Suppose there exist $\epsilon, \delta > 0$, $N_0 : [a,b] \to \mathbb{R}^{n\times n}$, $N_1, N_2 : [a,b]^2 \to \mathbb{R}^{n\times n}$ such that $\mathcal{P} := \mathcal{P}^*_{\{N_0,N_1,N_2\}} = \mathcal{P}_{\{N_0,N_1,N_2\}} \geq \epsilon I$ and

$$\mathcal{A}^*\mathcal{P}\mathcal{T} + \mathcal{T}^*\mathcal{P}\mathcal{A} \leq -\delta\mathcal{T}^*\mathcal{T}$$

where $\mathcal{T}$ and $\mathcal{A}$ are as defined in Eqn. (8). Then any solution of Eqns. (1) and (2) is exponentially stable.

Proof: We propose a Lyapunov function of the form

$$
\begin{aligned}
V(t) &= \langle \mathbf{x}_f(t), \mathcal{T}^*\mathcal{P}\mathcal{T}\mathbf{x}_f(t)\rangle_{L_2} \\
&= \langle \mathbf{x}_p(t), \mathcal{P}\mathbf{x}_p(t)\rangle_{L_2} \geq \epsilon\|\mathbf{x}_p\|^2.
\end{aligned}
$$

Since $\mathcal{P}$ is 3-PI, it is bounded and thus there exists $\gamma > 0$ such that $V(t) \leq \gamma\|\mathbf{x}_p(t)\|^2$. Then if $\mathbf{x}_f(t)$ satisfies (7), $\dot{V}$ is given by

$$
\begin{aligned}
\dot{V}(t) &= \langle \mathcal{T}\dot{\mathbf{x}}_f(t), \mathcal{P}\mathcal{T}\mathbf{x}_f(t)\rangle + \langle \mathbf{x}_f(t), \mathcal{P}\mathcal{T}\dot{\mathbf{x}}_f(t)\rangle \\
&= \langle \mathcal{A}\mathbf{x}_f(t), \mathcal{P}\mathcal{T}\mathbf{x}_f(t)\rangle + \langle \mathcal{T}\mathbf{x}_f(t), \mathcal{P}\mathcal{A}\mathbf{x}_f(t)\rangle \\
&= \langle \mathbf{x}_f(t), (\mathcal{A}^*\mathcal{P}\mathcal{T} + \mathcal{T}^*\mathcal{P}\mathcal{A})\,\mathbf{x}_f(t)\rangle \\
&\leq -\delta\|\mathcal{T}\mathbf{x}_f(t)\|^2 = -\delta\|\mathbf{x}_p(t)\|^2.
\end{aligned}
$$

Thus we conclude exponential stability of the solution $\mathbf{x}_p(t)$. ∎

Theorem 6 poses a convex optimization problem, whose feasibility implies stability of solutions of a coupled linear PDE. We refer to such optimization problems as Linear Operator Inequalities (LOIs). Solving a LOI requires parameterizing the 3-PI operator $\mathcal{P}$ using polynomials and enforcing the inequalities using LMIs. In the following section, we briefly introduce a method of enforcing positivity of a 3-PI operator using LMI constraints.

VII. Enforcing Positivity of 3-PI Operators

In the previous section, we showed how to pose the question of Lyapunov stability using 3-PI operator inequality constraints. In this and the remaining sections, we assume these operators are parameterized by polynomials and give an LMI which enforces positivity of these operators. The following theorem gives necessary and sufficient conditions for a 3-PI operator to have a 3-PI square root.

Theorem 7: For any square-integrable functions $Z(s)$ and $Z(s,\theta)$, and g where $g(s) \geq 0$ for all $s \in [a,b]$ and

$$
\begin{aligned}
N_0(s) &= g(s)Z(s)^T P_{11} Z(s) \\
N_1(s,\theta) &= g(s)Z(s)^T P_{12} Z(s,\theta) \\
&\quad + g(\theta)Z(\theta,s)^T P_{31} Z(\theta) \\
&\quad + \int_a^\theta g(\nu)Z(\nu,s)^T P_{33} Z(\nu,\theta)d\nu \\
&\quad + \int_\theta^s g(\nu)Z(\nu,s)^T P_{32} Z(\nu,\theta)d\nu \\
&\quad + \int_s^L g(\nu)Z(\nu,s)^T P_{22} Z(\nu,\theta)d\nu
\end{aligned}
$$

$$
\begin{aligned}
N_2(s,\theta) &= g(s)Z(s)^T P_{13} Z(s,\theta) \\
&\quad + g(\theta)Z(\theta,s)^T P_{21} Z(\theta) \\
&\quad + \int_a^s g(\nu)Z(\nu,s)^T P_{33} Z(\nu,\theta)d\nu \\
&\quad + \int_s^\theta g(\nu)Z(\nu,s)^T P_{23} Z(\nu,\theta)d\nu \\
&\quad + \int_\theta^L g(\nu)Z(\nu,s)^T P_{22} Z(\nu,\theta)d\nu,
\end{aligned}
$$

where

$$P = \begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix} \geq 0,$$

then $\mathcal{P}^*_{\{N_i\}} = \mathcal{P}_{\{N_i\}}$ and $\left\langle \mathbf{x}, \mathcal{P}_{\{N_i\}}\mathbf{x}\right\rangle_{L_2} \geq 0$ for all $\mathbf{x} \in L_2[a,b]$.

Proof: The operator is self-adjoint as per Lemma 2. Now define the operator

$$\left(\mathcal{Z}\mathbf{x}\right)(s) = \begin{bmatrix} \sqrt{g(s)}Z(s)\mathbf{x}(s) \\ \int_a^s \sqrt{g(s)}Z(s,\theta)\mathbf{x}(\theta)d\theta \\ \int_s^b \sqrt{g(s)}Z(s,\theta)\mathbf{x}(\theta)d\theta \end{bmatrix}.$$

Then

$$\begin{aligned}\left\langle \mathbf{x}, \mathcal{P}_{\{N_i\}}\mathbf{x}\right\rangle &= \left\langle \mathcal{Z}\mathbf{x}, P\mathcal{Z}\mathbf{x}\right\rangle \\ &= \left\langle P^{\frac{1}{2}}\mathcal{Z}\mathbf{x}, P^{\frac{1}{2}}\mathcal{Z}\mathbf{x}\right\rangle \geq 0.\end{aligned}$$

Thus $\mathcal{P}$ has square root $\mathcal{P}^{\frac{1}{2}} = P^{\frac{1}{2}}\mathcal{Z}$. ■

When the N_i are polynomial, we typically choose Z to be the vector of monomials of bounded degree. For $g(s) = 1$, the operators are positive on any domain. However, for $g(s) = (s-a)(b-s)$ the operator is only positive on the given domain $[a,b]$. For the most accurate results, we combine both choices of g. For notational convenience, we now define the set of functions which satisfy Theorem 7 in this way. Specifically, we denote $Z_d(x)$ as the matrix whose rows are a vector monomial basis for the vector-valued polynomials of degree d or less and define the cone of positive operators with polynomial multipliers and kernels associated with degree d as

$$\begin{aligned}&\Omega_d := \{\mathcal{P}_{\{N_i\}} + \mathcal{P}_{\{M_i\}} \;:\; \{N_i\} \text{ and } \{M_i\} \text{ satisfy} \\ &\text{the conditions of Thm. 7 with } Z = Z_d \text{ and} \\ &\text{where } g(s) = 1 \text{ and } g(s) = (s-a)(b-s), \text{ resp.}\}\end{aligned}$$

The dimension of the matrices M_i and N_i should be clear from context. The constraint $\mathcal{P}_{\{N_i\}} \in \Omega_d$ is then an LMI constraint on the coefficients of the polynomials $\{N_i\}$ and guarantees that $P_{\{N_i\}} \geq 0$. A Matlab toolbox (PIETOOLS) for setting up and solving LOIs based on Theorem 7 has recently been proposed and is discussed in Section IX.

VIII. An LMI for Stability Analysis

In this section, we briefly summarize the LMI for stability analysis using the Ω_d notation.

Theorem 8: Suppose there exist $\epsilon, \delta > 0$, $N_0 : [a,b] \to \mathbb{R}^{n\times n}$, $N_1, N_2 : [a,b]^2 \to \mathbb{R}^{n\times n}$ such that

$$\mathcal{P} := \mathcal{P}_{\{N_0 - \epsilon I, N_1, N_2\}} \in \Omega_d$$

and

$$-\delta\mathcal{T}^*\mathcal{T} - \mathcal{A}^*\mathcal{P}\mathcal{T} - \mathcal{T}^*\mathcal{P}\mathcal{A} \in \Omega_d$$

where $\mathcal{T}$ and $\mathcal{A}$ are as defined in Eqn. (8). Then any solution of Eqns. (1) and (2) is exponentially stable.

In the following section, we discuss the PIETOOLS toolbox which may be used for constructing the LMI associated with Theorem 8.

IX. PIETOOLS Implementation

In this section, we briefly discuss implementation of the stability test in Theorem 8 using the PIETOOLS Matlab toolbox. A manual for the PIETOOLS toolbox can be found in preliminary format on arxiv [23]. This toolbox allows for declaration and manipulation of 3-PI operators and 3-PI decision variables and enforcement of LOI constraints. PIETOOLS uses the SOSTOOLS LMI conversion process and pvar polynomial objects as defined in MULTIPOLY. PIETOOLS defines the opvar class of operators and overloads the multiplication and adjoint operations using the formulae in Theorem 1 and Lemma 2. Addition, concatenation, and scalar multiplication are likewise defined so that 3-PI operators can be treated in a similar manner to matrices. We include here an almost complete implementation of the stability analysis LMI and a brief description of each step.

1) Define independent polynomial variables.

```
pvar s,th;
```

2) Initialize an optimization problem structure (called sosprogram in SOSTOOLS).

```
X = sosprogram([s,th]);
```

3) Define relevant opvar data-objects.

```
opvar A,T;
A=..;T=..;
```

4) Declare the positive operator $\mathcal{P}$ and add inequality constraints.

```
[X,P] = sos_posopvar(X,n,I,s,th);
D = -del*T'*T-A'*P*T-T'*P*A
X = sosopineq(X,D);
```

5) Call the SDP solver.

```
X = sossolve(X);
```

6) Get the solution.

```
P_s = sosgetsol_opvar(X,P);
```

Clearly, step 3 requires use of the formulae in Eqns. (5) and (8). A script for declaration of such variables is included in the PIETOOLS package under the title `setup_PIETOOLS_PDE` which is then called in the script `solver PIETOOLS PDE`. Instructions for declaring the PDE are included in the header to `solver PIETOOLS PDE`. The stability analysis can then be performed using the option to call `executive_PIETOOLS_stability`.

X. Comparison with Previous Results

In this section, we examine the accuracy and computational complexity of the proposed stability algorithm by applying Theorem 8 to several well-studied and relatively trivial test cases. The algorithms are implemented using the PIETOOLS toolbox described in the previous section. All computation times are listed for an Intel i7-6950x processor with 64GB RAM and only account for time taken to solve the resulting LMI using Sedumi, excluding time taken for problem setup and polynomial manipulations.

Example 1: We begin with several variations of the diffusion equation. The first is adapted from [18].

$$\dot{x}(t,s) = \lambda x(t,s) + x_{ss}(t,s)$$

where $x(0) = x(1) = 0$ and which is known to be stable if and only if $\lambda < \pi^2 = 9.869604\cdots$. For the choice of $d = 1$ in Thm. 8, the algorithm is able to prove stability for $\lambda = 9.8696$ with a computation time of .54s.

Example 2: The second example from [19] is the same, but changes the boundary conditions to $x(0) = 0$ and $x_s(1) = 0$ and is unstable for $\lambda > 2.467$. For $d = 1$, the algorithm is able to prove stability for $\lambda = 2.467$ with identical computation time.

Example 3: The third example from [14] is not homogeneous

$$\begin{aligned}\dot{x}(t,s) &= (-.5s^3 + 1.3s^2 - 1.5s + .7 + \lambda)x(t,s)\\ &\quad + (3s^2 - 2s)x_s(t,s) + (s^3 - s^2 + 2)x_{ss}(t,s)\end{aligned}$$

where $x(0) = 0$ and $x_s(1) = 0$ and was estimated numerically to be unstable for $\lambda > 4.65$. For $d = 1$, the algorithm is able to prove stability for $\lambda = 4.65$ with similar computation time.

Example 4: In this example from [18], we have

$$\dot{x}(t,s) = \begin{bmatrix} 1 & 1.5 \\ 5 & .2 \end{bmatrix} x(t,s) + R^{-1} x_{ss}(t,s)$$

n	1	5	10	20
CPU sec	.54	37.4	745	31620

Fig. 1. Number of PDEs vs. Computation Time for Stability Test

with $x(0) = 0$ and $x_s(1) = 0$. In this case, using $d = 1$, we can prove stability for $R = 2.93$ (improvement over $R = 2.45$ in [18]) with a computation time of $1.21s$.

Example 5: In this example from [19], we have

$$\dot{x}(t,s) = \begin{bmatrix} 0 & 0 & 0 \\ s & 0 & 0 \\ s^2 & -s^3 & 0 \end{bmatrix} x(t,s) + R^{-1} x_{ss}(t,s)$$

with $x(0) = 0$ and $x_s(1) = 0$. In this case, using $d = 1$, we prove stability for $R = 21$ (and greater) with a computation time of $4.06s$.

Example 6: Finally, we explore computational complexity using a simple n-dimensional diffusion equation

$$\dot{x}(t,s) = x(t,s) + x_{ss}(t,s)$$

where $x(t,s) \in \mathbb{R}^n$. We then evaluate the computation time to perform the feasibility test for different size problems, from $n = 1$ to $n = 20$, choosing $d = 1$ - See Fig. 1. Note that no factors other than d influence computation time and the result is always stability.

XI. Illustration by Example

In this section, we use wave and beam examples to illustrate the flexible and universal nature of the proposed algorithm. The beam examples are particularly important in that (to the best of our knowledge) they have not previously been analyzed using LMI-based methods. In each case, we focus is on rendering the problem in the form of a vector-valued PDE of the form of Eqns. (1)-(2). We call particular attention to the following two questions.

- What are the states?
- What are the boundary conditions?

Choice of State: Prior to the introduction of state-space, ODEs would often be represented using scalar equations. For example, the spring-mass:

$$m\ddot{x}(t) = -c\dot{x}(t) - kx(t) + F(t)$$

is a scalar ODE. To represent this in the vector-valued state-space framework, we use x_1 and define an auxiliary state $x_2 = \dot{x}$. Similarly, PDEs are often represented as scalar equations using higher-order time derivatives (e.g. The wave equation is $\ddot{w} = w_{xx}$). The use of Eqns. (1)-(2), however, requires only first-order time derivatives. A more subtle problem is that the choice of states affects the question of stability. Specifically, the exponential stability criterion in Theorem 8 implies all states decay exponentially. Occasionally, however, PDEs are exponentially stable in some states, but neutrally stable in others. When choosing states, therefore, it is generally better to use higher-derivative states such as u_s instead of u, as stability in u_s implies stability in u. That is, $\|\mathbf{x}_p\| \le \|\mathcal{T}\|\|\mathbf{x}_f\|$, but the reverse is not necessarily true.

Boundary Conditions: Identification of a sufficient number of boundary conditions in the universal framework is particularly important. For the B matrix to have sufficient rank, the solution must be uniquely defined. One consideration to be aware of is that when we introduce additional variables to eliminate higher-order time-derivatives, these new variables must also have associated boundary conditions. This is typically solved by differentiating the original boundary conditions in time to obtain boundary conditions for the new variables.

In the following examples, we illustrate the process of choosing state and constructing the A_i and B matrices.

A. Beam Equation Examples

We first consider variations on the beam equation in both the Euler-Bernoulli (E-B) and Timoschenko (T) models. This case is particularly interesting, as the E-B model is fundamentally diffusive and the T model has hyperbolic character.

Euler-Bernoulli: In this first case, we recall our formulation of the cantilevered E-B beam:

$$\mathbf{x}_t = \underbrace{\begin{bmatrix} 0 & -c \\ 1 & 0 \end{bmatrix}}_{A_2} \mathbf{x}_{ss}$$

where $A_0 = A_1 = 0$, $n_3 = 2$, and $n_1 = n_2 = 0$. The boundary conditions take the form

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{B} \begin{bmatrix} u_1(0) \\ u_2(0) \\ u_1(L) \\ u_2(L) \\ u_{1s}(0) \\ u_{2s}(0) \\ u_{1s}(L) \\ u_{2s}(L) \end{bmatrix} = 0.$$

Entering this data into the software tool, we find the E-B beam is stable for any $c > 0$. Not that this implies the E-B is exponentially stable with respect to u_t and u_{ss}.

Timoschenko Beam We now consider the standard Timoschenko beam model where, for simplicity, we set $\rho = E = I = \kappa = G = 1$:

$$\begin{aligned} \ddot{w} &= \partial_s(w_s - \phi) & &= -\phi_s + w_{ss} \\ \ddot{\phi} &= \phi_{ss} + (w_s - \phi) & &= -\phi + w_s + \phi_{ss} \end{aligned}$$

with boundary conditions of the form

$$\begin{aligned} \phi(0) &= 0, & w(0) &= 0, \\ \phi_s(L) &= 0, & w_s(L) - \phi(L) &= 0. \end{aligned}$$

Our first step is to eliminate the second-order time-derivatives, and hence we choose $u_1 = w_t$ and $u_3 = \phi_t$. The next step is more problematic. The typical approach would be to use the boundary conditions as a guide and choose the remaining states as $u_2 = w_s - \phi$ and $u_4 = \phi_s$. This gives us 4 first order boundary conditions

$$u_1(0) = 0, \quad u_3(0) = 0, \quad u_4(L) = 0, \quad u_2(L) = 0.$$

Reconstructing the dynamics, we now have

$$\begin{aligned} u_{1t} &= u_{2s}, & u_{2t} &= u_{1s} - u_3 \\ u_{3t} &= u_{4s} + u_2, & u_{4t} &= u_{3s}. \end{aligned}$$

Expressing this in our standard form we have the purely hyperbolic construction

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}_t = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{A_0} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{A_1} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}_s$$

where $A_2 = []$ and $n_1 = n_3 = 0$ and $n_2 = 4$. The B matrix is then

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}}_{B} \begin{bmatrix} u_1(0) \\ u_1(0) \\ u_2(L) \\ u_2(L) \end{bmatrix} = 0$$

where B has row rank $n_2 = 4$. The code indicates this system is stable (using $\delta = 0$ in Thm. 8). However, when $\delta > 0$, the code is unable to find a Lyapunov function, indicating this formulation is probably not exponentially stable in all the given states. Numerical experimentation indicates that this question of exponential stability in some states but not others is common in wave-type equations of this form. To further illustrate, we now consider a slight modification - we choose $u_2 = w_s$ and $u_4 = \phi$. This leads to a mixed hyperbolic-diffusive formulation where

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}_t = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{A_0} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \tag{9}$$

$$+ \underbrace{\begin{bmatrix} 0 & 1 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{A_1} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}_s + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}}_{A_2} u_{4ss}$$

where $n_1 = 0$, $n_2 = 3$, and $n_3 = 1$ and with 5 boundary conditions

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \end{bmatrix}}_{B} \begin{bmatrix} u_{1-3}(0) \\ u_{1-3}(L) \\ u_4(0) \\ u_4(L) \\ u_{4s}(0) \\ u_{4s}(L) \end{bmatrix} = 0.$$

This formulation, however, does not appear to be stable in the given states. Since the only new state is ϕ, we hypothesize that the PDE model is not exponentially stable in $u_3 = \phi$. We may test this hypothesis numerically by adding a damping term $-cu_{4t} = -cu_3$ to the dynamics of $\dot{u}_3$. In this case, the only change is that now we modify Eqn. (9) so that

$$A_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & -c & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

The code indicates that this formulation is now stable for any $c > 0$. This simple numerical experiment indicates the Timoschenko beam model is stable with respect to $w_s - \phi, w_t, \phi_s$ and ϕ_t, but probably not with respect to ϕ.

B. Wave Equation with Boundary Feedback Examples

In this subsection, we consider wave equations attached at one end and free at the other with damping at the free end. This is a well-studied problem for which numerous stability results are available in the literature [24], [25]. The simplest formulation is

$$u_{tt}(t,s) = u_{ss}(t,s)$$
$$u(t,0) = 0 \qquad u_s(t,L) = -ku_t(t,L).$$

As with the beam examples, this has a purely hyperbolic formulation. Guided by the boundary conditions, we choose

$$u_1(t,s) = u_t(t,s), \qquad u_2(t,s) = u_s(t,s).$$

This yields

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}_t = \underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}}_{A_1} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}_s$$

where $A_0 = 0$, $A_2 = []$ $n_1 = n_3 = 0$ and $n_2 = 2$. The boundary conditions are now

$$\underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & k & 1 \end{bmatrix}}_{B} \begin{bmatrix} u(0) \\ u(L) \end{bmatrix} = 0.$$

This formulation is computed to be exponentially stable in the given state u_t, u_s for $k > 0$. We now consider a variation on this formulation.

Diffusive Formulation As a variation, we consider a non-diffusive formulation from [24] which was shown to be asymptotically stable in the state u for $a^2 + k^2 > 0$.

$$u_{tt}(t,s) = u_{ss}(t,s) - 2au_t(t,s) - a^2u(t,s), \; s \in [0,1]$$
$$u(t,0) = 0, \qquad u_s(t,1) = -ku_t(t,1)$$

In this case, we are forced to choose the variables $u_1 = u_t$ and $u_2 = u$ yielding the diffusive formulation

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}_t = \underbrace{\begin{bmatrix} -2a & -a^2 \\ 1 & 0 \end{bmatrix}}_{A_0} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{A_2} u_{2ss}$$

where $A_1 = 0$, $n_1 = 0$, $n_2 = 1$, and $n_3 = 1$. Note in this case that the boundary conditions on u_1 force us to consider this a hyperbolic state and the boundary conditions on u_2 make this a diffusive state! These boundary conditions are now expressed as

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & k & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_1(0) \\ u_1(L) \\ u_2(0) \\ u_2(L) \\ u_{2s}(0) \\ u_{2s}(L) \end{bmatrix} = 0.$$

Computation indicates this model is stable, but not exponentially stable in the given state - a result confirmed in [24], [25].

XII. Conclusion

In this paper, we have shown how to use LMIs to accurately test stability of a large class of coupled linear PDEs. To achieve this result, we have shown how to convert well-posed coupled linear PDEs - defined on state $\mathbf{x}_p$ with associated boundary conditions and continuity constraints - to Partial-Integral Equations (PIEs) with state $\mathbf{x}_f$ - a formulation which is defined using the algebra of 3-PI partial-integral operators and which does not require boundary conditions or continuity constraints on $\mathbf{x}_f$. We have shown that stability of PIEs can be reformulated as a Linear Operator Inequality (LOI) expressed using 3-PI operators and operator positivity constraints. We have shown how to parameterize 3-PI operators using polynomials and how to enforce positivity of 3-PI operators using LMI constraints on the coefficients of these polynomials. We have used the Matlab toolbox PIETOOLS to solve the resulting LOIs and applied the results to a variety of numerical examples. The numerical results indicate little or no conservatism in the resulting stability conditions to several significant figures even for low polynomial degree. By conversion of LMIs developed for ODEs to LOIs, it is possible that these results can be extended to: PDEs with uncertainty; H_∞-gain analysis of PDEs; H_∞-optimal observer synthesis for PDEs; and H_∞-optimal control of PDEs. In addition, it is possible that the framework may be extended to multiple spatial dimensions using the multivariate representation proposed in [26].

Appendix
Proof of Theorem 1

Theorem 9: For any bounded functions $B_0, N_0 : [a,b] \to \mathbb{R}^{n\times n}$, $B_1, B_2, N_1, N_2 : [a,b]^2 \to \mathbb{R}^{n\times n}$, we have

$$\mathcal{P}_{\{R_0,R_1,R_2\}} = \mathcal{P}_{\{B_0,B_1,B_2\}}\mathcal{P}_{\{N_0,N_1,N_2\}}$$

where

$$R_0(s) = B_0(s)N_0(s) \tag{10}$$
$$R_1(s,\theta) = B_0(s)N_1(s,\theta) + B_1(s,\theta)N_0(\theta) + \int_a^\theta B_1(s,\xi)N_2(\xi,\theta)d\xi + \int_\theta^s B_1(s,\xi)N_1(\xi,\theta)d\xi + \int_s^b B_2(s,\xi)N_1(\xi,\theta)d\xi$$
$$R_2(s,\theta) = B_0(s)N_2(s,\theta) + B_2(s,\theta)N_0(\theta) + \int_a^s B_1(s,\xi)N_2(\xi,\theta)d\xi + \int_s^\theta B_2(s,\xi)N_2(\xi,\theta)d\xi + \int_\theta^b B_2(s,\xi)N_1(\xi,\theta)d\xi$$

Proof:

To prove the theorem, we exploit the linear structure of the operator to decompose

$$\mathcal{P}_{\{B_0,B_1,B_2\}} = \mathcal{P}_{\{B_0,0,0\}} + \mathcal{P}_{\{0,B_1,0\}} + \mathcal{P}_{\{0,0,B_2\}}$$

and

$$\mathcal{P}_{\{N_0,N_1,N_2\}} = \mathcal{P}_{\{N_0,0,0\}} + \mathcal{P}_{\{0,N_1,0\}} + \mathcal{P}_{\{0,0,N_2\}}.$$

Then

$$\begin{aligned}&\mathcal{P}_{\{B_0,B_1,B_2\}}\mathcal{P}_{\{N_0,N_1,N_2\}}x\\ &= \mathcal{P}_{\{B_0,0,0\}}\mathcal{P}_{\{N_0,N_1,N_2\}} + \mathcal{P}_{\{0,B_1,B_2\}}\mathcal{P}_{\{N_0,0,0\}}\\ &\quad + \mathcal{P}_{\{0,B_1,0\}}\mathcal{P}_{\{0,N_1,0\}} + \mathcal{P}_{\{0,B_1,0\}}\mathcal{P}_{\{0,0,N_2\}}\\ &\quad + \mathcal{P}_{\{0,0,B_2\}}\mathcal{P}_{\{0,N_1,0\}} + \mathcal{P}_{\{0,0,B_2\}}\mathcal{P}_{\{0,0,N_2\}}.\end{aligned}$$

We now consider each term separately, starting with the first two, which are trivial. First,

$$\begin{aligned}&\left(\mathcal{P}_{\{B_0,0,0\}}\mathcal{P}_{\{N_0,N_1,N_2\}}x\right)(s)\\ &= B_0(s)N_0(s)x(s) + \int_a^s B_0(s)N_1(s,\theta)x(\theta)d\theta\\ &\qquad + \int_s^b B_0(s)N_2(s,\theta)x(\theta)d\theta\\ &= \mathcal{P}_{\{R_0,R_{1a},R_{2a}\}},\end{aligned}$$

where

$$R_0(s) = B_0(s)N_0(s), \quad R_{1a}(s,\theta) = B_0(s)N_1(s,\theta),$$
$$R_{2a}(s,\theta) = B_0(s)N_2(s,\theta).$$

Similarly,

$$\begin{aligned}&\left(\mathcal{P}_{\{0,B_1,B_2\}}\mathcal{P}_{\{N_0,0,0\}}x\right)(s)\\ &= \int_a^s B_1(s,\theta)N_0(\theta)x(\theta)d\theta + \int_s^b B_2(s,\theta)N_0(\theta)x(\theta)d\theta\\ &= \mathcal{P}_{\{0,R_{1b},R_{2b}\}},\end{aligned}$$

where

$$R_{1b}(s,\theta) = B_1(s,\theta)N_0(\theta),$$
$$R_{2b}(s,\theta) = B_2(s,\theta)N_0(\theta).$$

We now proceed to the more difficult terms. For these, recall the indicator function

$$I(s) = \begin{cases} 1, & \text{if } s > 0 \\ 0, & \text{otherwise.} \end{cases}$$

For the first term, we note that

$$I(s-\eta)I(\eta-\xi) = \begin{cases} I(s-\xi), & \text{if } \eta \in [\xi, s] \\ 0, & \text{otherwise.} \end{cases}$$

This allows us to change the variables of integration as follows.

$$\begin{aligned}
&\left(\mathcal{P}_{\{0,B_1,0\}}\mathcal{P}_{\{0,N_1,0\}}x\right)(s) \\
&= \int_a^s B_1(s,\eta)\int_a^\eta N_1(\eta,\xi)x(\xi)d\xi d\eta \\
&= \int_a^b I(s-\eta)B_1(s,\eta)\int_a^b I(\eta-\xi)N_1(\eta,\xi)x(\xi)d\xi d\eta \\
&= \int_a^b \left(\int_a^b I(s-\eta)I(\eta-\xi)B_1(s,\eta)N_1(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \int_a^b \left(\int_\xi^s I(s-\xi)B_1(s,\eta)N_1(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \int_a^s \left(\int_\xi^s B_1(s,\eta)N_1(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \left(\mathcal{P}_{\{0,R_{1c},0\}}x\right)(s),
\end{aligned}$$

where

$$R_{1c}(s,\theta) = \int_\theta^s B_1(s,\xi)N_1(\xi,\theta)d\xi.$$

Next, we use another identity

$$I(s-\eta)I(\xi-\eta) = I(s-\xi)I(\xi-\eta)+I(\xi-s)I(s-\eta)$$

to establish the following.

$$\begin{aligned}
&\left(\mathcal{P}_{\{0,B_1,0\}}\mathcal{P}_{\{0,0,N_2\}}x\right)(s) \\
&= \int_a^s B_1(s,\eta)\int_\eta^b N_2(\eta,\xi)x(\xi)d\xi d\eta \\
&= \int_a^b I(s-\eta)B_1(s,\eta)\int_a^b I(\xi-\eta)N_2(\eta,\xi)x(\xi)d\xi d\eta \\
&= \int_a^b \left(\int_a^b I(s-\eta)I(\xi-\eta)B_1(s,\eta)N_2(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \int_a^b I(s-\xi)\left(\int_a^\xi B_1(s,\eta)N_2(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&\qquad + \int_a^b I(\xi-s)\left(\int_a^s B_1(s,\eta)N_2(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \int_a^s \left(\int_a^\xi B_1(s,\eta)N_2(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&\qquad + \int_s^b \left(\int_a^s B_1(s,\eta)N_2(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \left(\mathcal{P}_{\{0,R_{1d},R_{2d}\}}x\right)(s),
\end{aligned}$$

where

$$R_{1d}(s,\theta) = \int_a^\theta B_1(s,\xi)N_2(\xi,\theta)d\xi$$

$$R_{2d}(s,\theta) = \int_a^s B_1(s,\xi)N_2(\xi,\theta)d\xi.$$

The remaining two identities are minor variations on the two we most recently derived.

$$\begin{aligned}
&\left(\mathcal{P}_{\{0,0,B_2\}}\mathcal{P}_{\{0,N_1,0\}}x\right)(s) \\
&= \int_s^b B_2(s,\eta)\int_a^\eta N_1(\eta,\xi)x(\xi)d\xi d\eta \\
&= \int_a^b I(\eta-s)B_2(s,\eta)\int_a^b I(\eta-\xi)N_1(\eta,\xi)x(\xi)d\xi d\eta \\
&= \int_a^b \left(\int_a^b I(\eta-s)I(\eta-\xi)B_2(s,\eta)N_1(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \int_a^b I(s-\xi)\left(\int_s^b B_2(s,\eta)N_1(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&+ \int_a^b I(\xi-s)\left(\int_\xi^b B_2(s,\eta)N_1(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \int_a^s \left(\int_s^b B_2(s,\eta)N_1(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&+ \int_s^b \left(\int_\xi^b B_2(s,\eta)N_1(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \left(\mathcal{P}_{\{0,R_{1e},R_{2e}\}}x\right)(s),
\end{aligned}$$

where

$$R_{1e}(s,\theta) = \int_s^b B_2(s,\xi)N_1(\xi,\theta)d\xi$$

$$R_{2e}(s,\theta) = \int_\theta^b B_2(s,\xi)N_1(\xi,\theta)d\xi.$$

Finally,

$$\begin{aligned}
&\left(\mathcal{P}_{\{0,0,B_2\}}\mathcal{P}_{\{0,0,N_2\}}x\right)(s) \\
&= \int_s^b B_2(s,\eta)\int_\eta^b N_2(\eta,\xi)x(\xi)d\xi d\eta \\
&= \int_a^b I(\eta-s)B_2(s,\eta)\int_a^b I(\xi-\eta)N_2(\eta,\xi)x(\xi)d\xi d\eta \\
&= \int_a^b \left(\int_a^b I(\xi-\eta)I(\eta-s)B_2(s,\eta)N_2(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \int_a^b I(\xi-s)\left(\int_s^\xi B_2(s,\eta)N_2(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \int_s^b \left(\int_s^\xi B_2(s,\eta)N_2(\eta,\xi)d\eta\right)x(\xi)d\xi \\
&= \left(\mathcal{P}_{\{0,0,R_{2f}\}}x\right)(s),
\end{aligned}$$

where

$$R_{2f}(s,\theta) = \int_s^\theta B_2(s,\xi)N_2(\xi,\theta)d\xi.$$

Combining all terms, we have

$$\begin{aligned}
&\mathcal{P}_{\{B_0,B_1,B_2\}}\mathcal{P}_{\{N_0,N_1,N_2\}}x \\
&= \mathcal{P}_{\{B_0,0,0\}}\mathcal{P}_{\{N_0,N_1,N_2\}} + \mathcal{P}_{\{0,B_1,B_2\}}\mathcal{P}_{\{N_0,0,0\}} \\
&\quad + \mathcal{P}_{\{0,B_1,0\}}\mathcal{P}_{\{0,N_1,0\}} + \mathcal{P}_{\{0,B_1,0\}}\mathcal{P}_{\{0,0,N_2\}} \\
&\quad + \mathcal{P}_{\{0,0,B_2\}}\mathcal{P}_{\{0,N_1,0\}} + \mathcal{P}_{\{0,0,B_2\}}\mathcal{P}_{\{0,0,N_2\}} \\
&= \mathcal{P}_{\{R_0,R_{1a},R_{2a}\}} + \mathcal{P}_{\{0,R_{1b},R_{2b}\}} + \mathcal{P}_{\{0,R_{1c},0\}} \\
&\quad + \mathcal{P}_{\{0,R_{1d},R_{2d}\}} + \mathcal{P}_{\{0,R_{1e},R_{2e}\}} + \mathcal{P}_{\{0,0,R_{2f}\}} \\
&= \mathcal{P}_{\{R_0,R_{1a}+R_{1b}+R_{1c}+R_{1d}+R_{1e},R_{2a}+R_{2b}+R_{2d}+R_{2e}+R_{2f}\}} \\
&= \mathcal{P}_{\{R_0,R_1,R_2\}}.
\end{aligned}$$

This follows since

$$\begin{aligned}
&R_1(s,\theta) = R_{1a}(s,\theta) + R_{1b}(s,\theta) \\
&\qquad + R_{1c}(s,\theta) + R_{1d}(s,\theta) + R_{1e}(s,\theta) \\
&= B_0(s)N_1(s,\theta) + B_1(s,\theta)N_0(\theta) \\
&+ \int_\theta^s B_1(s,\xi)N_1(\xi,\theta)d\xi + \int_a^\theta B_1(s,\xi)N_2(\xi,\theta)d\xi \\
&\qquad + \int_s^b B_2(s,\xi)N_1(\xi,\theta)d\xi
\end{aligned}$$

and

$$\begin{aligned}
&R_2(s,\theta) = R_{2a}(s,\theta) + R_{2b}(s,\theta) \\
&\qquad + R_{2d}(s,\theta) + R_{2e}(s,\theta) + R_{2f}(s,\theta) \\
&= B_0(s)N_2(s,\theta) + B_2(s,\theta)N_0(\theta) \\
&+ \int_a^s B_1(s,\xi)N_2(\xi,\theta)d\xi + \int_\theta^b B_2(s,\xi)N_1(\xi,\theta)d\xi \\
&\qquad + \int_s^\theta B_2(s,\xi)N_2(\xi,\theta)d\xi.
\end{aligned}$$

■

Appendix
Proof of Lemma 2

Lemma 10: For any bounded functions $N_0 : [a,b] \to \mathbb{R}^{n\times n}$, $N_1, N_2 : [a,b]^2 \to \mathbb{R}^{n\times n}$ and any $\mathbf{x}, \mathbf{y} \in L_2^n[a,b]$, we have

$$\left\langle \mathcal{P}_{\{N_0,N_1,N_2\}}\mathbf{x}, \mathbf{y} \right\rangle_{L_2} = \left\langle \mathbf{x}, \mathcal{P}_{\{\hat{N}_0,\hat{N}_1,\hat{N}_2\}}\mathbf{y} \right\rangle_{L_2}$$

where

$$\begin{aligned}
\hat{N}_0(s) &= N_0(s)^T, \\
\hat{N}_1(s,\eta) &= N_2(\eta,s)^T, \\
\hat{N}_2(s,\eta) &= N_1(\eta,s)^T.
\end{aligned}$$

Proof: Noting that

$$\begin{aligned}
\left\langle \mathcal{P}_{\{N_0,N_1,N_2\}}\mathbf{x}, \mathbf{y} \right\rangle_{L_2} &= \left\langle \mathcal{P}_{\{N_0,0,0\}}\mathbf{x}, \mathbf{y} \right\rangle_{L_2} \\
&\quad + \left\langle \mathcal{P}_{\{0,N_1,0\}}\mathbf{x}, \mathbf{y} \right\rangle_{L_2} + \left\langle \mathcal{P}_{\{0,0,N_2\}}\mathbf{x}, \mathbf{y} \right\rangle_{L_2},
\end{aligned}$$

we can decompose the adjoint as follows. Clearly

$$\left\langle \mathcal{P}_{\{N_0,0,0\}}\mathbf{x}, \mathbf{y} \right\rangle_{L_2} = \left\langle \mathbf{x}, \mathcal{P}_{\{N_0,0,0\}}\mathbf{y} \right\rangle_{L_2}.$$

For the other terms, we use a change of integration to yield

$$\begin{aligned}
&\left\langle \mathcal{P}_{\{0,N_1,0\}}\mathbf{x}, \mathbf{y} \right\rangle \\
&= \int_a^b \left(\int_a^s N_1(s,\eta)\mathbf{x}(\eta)d\eta \right)^T \mathbf{x}(s)ds \\
&= \int_a^b \int_a^s \mathbf{x}(\eta)^T N_1(s,\eta)^T \mathbf{x}(s)d\eta ds \\
&= \int_a^b \int_a^b I(s-\eta)\mathbf{x}(\eta)^T N_1(s,\eta)^T \mathbf{x}(s)d\eta ds \\
&= \int_a^b \int_\eta^b \mathbf{x}(\eta)^T N_1(s,\eta)^T \mathbf{x}(s)ds d\eta \\
&= \int_a^b \int_s^b \mathbf{x}(s)^T N_1(\eta,s)^T \mathbf{x}(\eta)d\eta ds \\
&= \left\langle \mathbf{x}, \mathcal{P}_{\{0,0,\hat{N}_2\}}\mathbf{y} \right\rangle.
\end{aligned}$$

Likewise,

$$\begin{aligned}
&\left\langle \mathcal{P}_{\{0,0,N_2\}}\mathbf{x}, \mathbf{y} \right\rangle \\
&= \int_a^b \left(\int_s^b N_2(s,\eta)\mathbf{x}(\eta)d\eta \right)^T \mathbf{x}(s)ds \\
&= \int_a^b \int_s^b \mathbf{x}(\eta)^T N_2(s,\eta)^T \mathbf{x}(s)d\eta ds \\
&= \int_a^b \int_a^b I(\eta-s)\mathbf{x}(\eta)^T N_2(s,\eta)^T \mathbf{x}(s)d\eta ds \\
&= \int_a^b \int_a^\eta \mathbf{x}(\eta)^T N_2(s,\eta)^T \mathbf{x}(s)ds d\eta \\
&= \int_a^b \int_a^s \mathbf{x}(s)^T N_2(\eta,s)^T \mathbf{x}(\eta)d\eta ds \\
&= \left\langle \mathbf{x}, \mathcal{P}_{\{0,\hat{N}_1,0\}}\mathbf{y} \right\rangle.
\end{aligned}$$

Combining these terms, we confirm the statement of the lemma. ■

PLACE PHOTO HERE

Matthew M. Peet received the B.S. degree in physics and in aerospace engineering from the University of Texas, Austin, TX, USA, in 1999 and the M.S. and Ph.D. degrees in aeronautics and astronautics from Stanford University, Stanford, CA, in 2001 and 2006, respectively. He was a Postdoctoral Fellow at INRIA, Paris, France from 2006 to 2008. He was an Assistant Professor of Aerospace Engineering at the Illinois Institute of Technology, Chicago, IL, USA, from 2008 to 2012. Currently, he is an Associate Professor of Aerospace Engineering at Arizona State University, Tempe, AZ, USA. Dr. Peet received a National Science Foundation CAREER award in 2011.

References

[1] M. Marion and R. Temam, "Nonlinear Galerkin methods," *SIAM Journal on numerical analysis*, vol. 26, no. 5, pp. 1139–1157, 1989.

[2] S. Ravindran, "A reduced-order approach for optimal control of fluids using proper orthogonal decomposition," *International journal for numerical methods in fluids*, vol. 34, no. 5, pp. 425–448, 2000.

[3] C. Rowley, "Model reduction for fluids, using balanced proper orthogonal decomposition," *International Journal of Bifurcation and Chaos*, vol. 15, no. 03, pp. 997–1013, 2005.

[4] A. Smyshlyaev and M. Krstic, "Backstepping observers for a class of parabolic PDEs," *Systems & Control Letters*, vol. 54, no. 7, pp. 613–625, 2005.

[5] O. Aamo, "Disturbance rejection in 2 x 2 linear hyperbolic systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 5, pp. 1095–1106, 2013.

[6] E. Fridman and Y. Orlov, "An LMI approach to H_∞ boundary control of semilinear parabolic and hyperbolic systems," *Automatica*, vol. 45, no. 9, pp. 2060–2066, 2009.

[7] E. Fridman and M. Terushkin, "New stability and exact observability conditions for semilinear wave equations," *Automatica*, vol. 63, pp. 1–10, 2016.

[8] O. Solomon and E. Fridman, "Stability and passivity analysis of semilinear diffusion pdes with time-delays," *International Journal of Control*, vol. 88, no. 1, pp. 180–192, 2015.

[9] O. Gaye, L. Autrique, Y. Orlov, E. Moulay, S. Brémond, and R. Nouailletas, "H_∞ stabilization of the current profile in tokamak plasmas via an LMI approach," *Automatica*, vol. 49, no. 9, pp. 2795–2804, 2013.

[10] I. Lasiecka and R. Triggiani, *Control theory for partial differential equations: Volume 1, Abstract parabolic systems: Continuous and approximation theories*. Cambridge University Press, 2000.

[11] R. Curtain and H. Zwart, *An Introduction to Infinite-Dimensional Linear Systems Theory*. Springer-Verlag, 1995.

[12] A. Bensoussan, G. D. Prato, M. C. Delfour, and S. K. Mitter, *Representation and Control of Infinite Dimensional Systems Volume I*. Birkhäuser, 1992.

[13] J. Villegas, "A port-hamiltonian approach to distributed parameter systems," Ph.D. dissertation, 2007.

[14] A. Gahlawat and M. Peet, "A convex sum-of-squares approach to analysis, state feedback and output feedback control of parabolic PDEs," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1636–1651, 2017.

[15] ——, "Optimal state feedback boundary control of parabolic PDEs using SOS polynomials," in *Proceedings of the American Control Conference*, 2016.

[16] ——, "Output feedback control of inhomogeneous parabolic PDEs with point actuation and point measurement using SOS and semi-separable kernels," in *Proceedings of the IEEE Conference on Decision and Control*, 2015.

[17] M. Ahmadi, G. Valmorbida, and A. Papachristodoulou, "Dissipation inequalities for the analysis of a class of PDEs," *Automatica*, vol. 66, pp. 163–171, 2016.

[18] G. Valmorbida, M. Ahmadi, and A. Papachristodoulou, "Semidefinite programming and functional inequalities for distributed parameter systems," in *Proceedings of the IEEE Conference on Decision and Control*, 2014, pp. 4304–4309.

[19] ——, "Stability analysis for a class of partial differential equations via semidefinite programming," *IEEE Transactions on Automatic Control*, vol. 61, no. 6, pp. 1649–1654, 2016.

[20] A. Gahlawat and G. Valmorbida, "A semi-definite programming approach to stability analysis of linear partial differential equations," in *Proceedings of the IEEE Conference on Decision and Control*, 2017, pp. 1882–1887.

[21] M. Peet, "A new state-space representation for coupled PDEs and scalable Lyapunov stability analysis in the SOS framework," in *Proceedings of the IEEE Conference on Decision and Control*, 2018.

[22] J. Appell, A. Kalitvin, and P. Zabrejko, *Partial Integral Operators and Integro-Differential Equations: Pure and Applied Mathematics*. CRC Press, 2000.

[23] S. Shivakumar, A. Das, and M. Peet, "Pietools: A matlab toolbox for manipulation and optimization of partial integral operators," in *Proceedings of the American Control Conference*, 2020.

[24] G. Chen, "Energy decay estimates and exact boundary value controllability for the wave equation in a bounded domain," *J. Math. Pures Appl.*, vol. 58, pp. 249–273, 1979.

[25] R. Datko, J. Lagnese, and M. Polis, "An example on the effect of time delays in boundary feedback stabilization of wave equations," *SIAM Journal on Control and Optimization*, vol. 24, no. 1, pp. 152–156, 1986.

[26] M. M. Peet, "Exponentially stable nonlinear systems have polynomial Lyapunov functions on bounded regions," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, 2009.