Discussion Paper: A New Mathematical Framework for Representation and Analysis of Coupled PDEs ⋆

Matthew M. Peet *

* *Arizona State University, Tempe, AZ 85287-6106, USA. (e-mail: mpeet@asu.edu).*

Abstract: We present a framework for stability analysis of systems of coupled linear Partial-Differential Equations (PDEs). The class of PDE systems considered in this paper includes parabolic, elliptic and hyperbolic systems with Dirichelet, Neuman and mixed boundary conditions. The results in this paper apply to systems with a single spatial variable. We exploit a new concept of state for PDE systems which allows us to include the boundary conditions directly in the dynamics of the PDE. The resulting algorithms are implemented in Matlab, tested on several motivating and illustrative examples, and the codes have been posted online. Numerical testing indicates the approach has little or no conservatism for a large class of systems and can analyze systems of up to 20 coupled PDEs.

Keywords: Distributed Parameter Systems, PDE, LMI, Convex.

1. INTRODUCTION

Partial Differential Equations (PDEs) are used to model systems where the state depends continuously on both time and secondary independent variables. The most common method for stability analysis of PDEs is to project the state onto a finite-dimensional vector space using, e.g. Marion and Temam (1989); Ravindran (2000); Rowley (2005) 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 often prone to instability and numerical ill-conditioning. Attempts to develop a rigorous state-space theory for PDEs without discretization includes the significant literature on Semigroup theory Lasiecka and Triggiani (2000); Curtain and Zwart (1995); Bensoussan et al. (1992). Perhaps the most well-known method for stabilization of PDEs without discretization is the backstepping approach to controller synthesis Smyshlyaev and Krstic (2005) (See the 2-state example in Aamo (2013)). Unfortunately, however, backstepping cannot currently be used for direct construction of Lyapunov functions for the purpose of stability analysis. Additional work on the use of computational methods and LMIs for computing Lyapunov functions for PDEs can be found in the work of Fridman and Orlov (2009); Fridman and Terushkin (2016); Solomon and Fridman (2015). Other examples of LMI methods for stability analysis of PDEs include Gaye et al. (2013).

Beginning in 2006 (Papachristodoulou and Peet (2006)), LMI and SOS methods have been used to analyze stability and input-output properties of PDEs. Examples of this work from our lab can be found in Gahlawat and Peet, 2016, 2015) and work from our colleagues can be found in Ahmadi et al. (2016); Valmorbida et al. (2014, 2016). While this previous work has proven somewhat effective, the results obtained are largely limited to scalar PDEs. To understand the source of the difficulty in extension to coupled PDEs, consider a relatively simple vector-valued PDE

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

where $\mathbf{x}(t,0) = \mathbf{x}_s(t,0) = 0$.

Problem 1: An obvious Lyapunov functional for this system is

$$V(\mathbf{x}) = \int_0^L \mathbf{x}(s)^T M(s)\mathbf{x}(s)ds.$$

Clearly $V(x) > 0$ if $M(s) \geq \epsilon I$ for all s and some $\epsilon > 0$. However, now take the derivative of this functional,

$$\dot V(\mathbf{x}) = \int_0^L \begin{bmatrix}\mathbf{x}\\ \mathbf{x}_s\\ \mathbf{x}_{ss}\end{bmatrix}(s)^T \underbrace{\begin{bmatrix} A_0(s)^T M(s) + M(s)A_0(s) & M(s)A_1(s) & M(s)A_2(s)\\ A_1(s)^T M(s) & 0 & 0\\ A_2(s)^T M(s) & 0 & 0\end{bmatrix}}_{D(s)} \begin{bmatrix}\mathbf{x}\\ \mathbf{x}_s\\ \mathbf{x}_{ss}\end{bmatrix}(s)ds.$$

The problem then, is that $D(s) \not< 0$ for ANY choice of A_i! Now obviously the problem is that the terms $\mathbf{x}$, $\mathbf{x}_s$ and $\mathbf{x}_{ss}$ cannot be considered independent. However, the relationship between these three vector-valued functions is not clear, and is only determined by the boundary conditions. Indeed it is easy to show that

$$\mathbf{x}(s) = s\mathbf{x}(0) + \mathbf{x}_s(0) + \int_0^s (s-\eta)\mathbf{x}_{ss}(\eta)d\eta.$$

⋆ This paper was supported by the office of Navy Research under Grant ONR #N000014-17-1-2117

However, what this example illustrates is that boundary conditions are not an afterthought to the formulation of the PDE, but have a profound impact on the distributed dynamics of the system. This impact can be clearly seen in the following extreme example.

Problem 2:

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

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. In the semigroup framework, we would define $D(\mathcal{A}) = \{\mathbf{u} \in H^2 : \mathbf{u}(0) = w_1(t),\ \mathbf{u}_s(0) = w_2(t)\}$. If we now use the identity mentioned previously, then we find

$$\dot{\mathbf{u}}(t,s) = s w_1(t) + w_2(t) + \int_0^s (s-\eta)\mathbf{u}_{ss}(\eta) d\eta.$$

This formulation of the same system directly incorporates the boundary conditions into the dynamics - which are now expressed using the more fundamental state $\mathbf{u}_{ss}$. Indeed, it is relatively easy to see that for any suitably well-defined PDE, the more primal states $\mathbf{u}$ and $\mathbf{u}_s$ can be expressed in terms of $\mathbf{u}_{ss}$. In this paper, we develop these observations into a mathematical framework which allows us to directly express the dynamics in the fundamental state by eliminating the boundary conditions and incorporating this auxiliary information directly into the "generator" of the dynamics.

1.1 A Universal Framework

In this paper, we consider the problem of stability analysis of multiple coupled linear PDEs in a single spatial variable. We write these systems in the universal form

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

where the x_i are **vector**-valued functions $x_i : [a,b] \times \mathbb{R}^+ \to \mathbb{R}^{n_i}$ and with boundary constraints of the form

$$B\mathrm{col}\left[x_2(t,a)\ x_2(t,b)\ x_3(t,a)\ x_3(t,b)\ x_{3s}(t,a)\ x_{3s}(t,b)\right] = 0$$

where B is of row rank $n_2 + 2n_3$. We refer to $\mathbf{x}_p : [a,b] \times \mathbb{R}^+ \to \mathbb{R}^{n_1+n_2+n_3}$ as the *primal state*. These types of systems arise when there are multiple interacting spatially-distributed states and include wave equations, beam equations, et c.

The main technical contribution is to show that if $\mathbf{x}_p$ satisfies the 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

$$\dot{\mathbf{x}}_p = \mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f, \qquad \mathbf{x}_p = \mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f$$

where $\mathcal{P}_{\{H_0,H_1,H_2\}}$ and $\mathcal{P}_{\{G_0,G_1,G_2\}}$ are multiplier/integral operators which have the form

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

where the matrix-valued functions G_i and H_i are uniquely determined by the matrix B and where $\mathbf{x}_f \in L_2[a,b]$ need not satisfy any boundary constraints in order to define a solution. This identity implies that for any $\mathbf{x}_f$, the initial value problem is well-defined - implying that this is a boundary-condition independent representation of the state of the system.

We then use these identities to propose a Lyapunov function of the form

$$V(\mathbf{x}_p) = \left\langle \mathbf{x}_p, \mathcal{P}_{\{M,N_1,N_2\}}\mathbf{x}_p \right\rangle$$

whose derivative is then

$$\begin{aligned}\dot{V}(\mathbf{x}_p) &= \left\langle \mathbf{x}_f, \mathcal{P}^*_{\{G_0,G_1,G_2\}}\mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f \right\rangle \\ &+ \left\langle \mathbf{x}_f, \mathcal{P}^*_{\{H_0,H_1,H_2\}}\mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f \right\rangle \\ &= \left\langle \mathbf{x}_f, P_{\{D_0,D_1,D_2\}}\mathbf{x}_f \right\rangle.\end{aligned}$$

for some D_i where the transformation from the variables M, N_i to D_i is linear. We note that the structure of these quadratic Lyapunov functions are implied by the closed-loop stability conditions established via the backstepping transformation, as shown in Gahlawat and Peet. We then proceed to parameterize the variables M, N_i using polynomials and show that positivity and negativity of operators of the form $\mathcal{P}_{\{M,N_1,N_2\}}$ may be enforced using an LMI on the coefficients of the polynomials M, N_i. Finally, present a software tool which constructs and tests the resulting LMI and show that almost any stability result on coupled PDEs may be verified using this tool.

2. NOTATION

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

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

3. THE M, N_1, N_2 PARAMETRIZATION OF OPERATORS

In this section, we propose a new parameterizations of multiplier and integral operators with kernels of the semi-separable class. This notation will allow us to efficiently represent both our state transformation and the stability conditions proposed in the following sections. First, we define the parameterizations.

For given bounded functions $M : [a,b] \to \mathbb{R}^{n \times n}$, $N_1 : [a,b]^2 \to \mathbb{R}^{n \times n}$, and $N_2 : [a,b]^2 \to \mathbb{R}^{n \times n}$, we use $\mathcal{P}_{\{M,N_1,N_2\}} : L_2^n[a,b] \to L_2^n[a,b]$ to denote the multiplier and integral operator

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

where M is the multiplier and the kernel of the corresponding integral operator is given by

$$N(s,\theta) = \begin{cases} N_1(s,\theta) & \theta < s \\ N_2(s,\theta) & \theta \geq s. \end{cases}$$

In this paper, the functions M, N_1, and N_2 will always be polynomial and hence all functions are bounded.

3.1 Composition of M, N_1, N_2 operators

We now derive expressions for the composition and adjoint of $\mathcal{P}_{\{M,N_1,N_2\}}$ operators. Both composition and adjoint of $\mathcal{P}_{\{M,N_1,N_2\}}$ operators are of the $\mathcal{P}_{\{M,N_1,N_2\}}$ class and can be efficiently expressed in terms of matrix operators on the functions M, N_1, and N_2. First, we address composition.

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

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

where

$$R_0(s) = B_0(s)N_0(s) \tag{2}$$
$$\begin{aligned} R_1(s,\theta) &= B_0(s)N_1(s,\theta) + B_1(s,\theta)N_0(\theta) \\ &\quad + \int_a^\theta B_1(s,\xi)N_2(\xi,\theta)d\xi + \int_\theta^s B_1(s,\xi)N_1(\xi,\theta)d\xi \\ &\quad + \int_s^b B_2(s,\xi)N_1(\xi,\theta)d\xi \\ R_2(s,\theta) &= B_0(s)N_2(s,\theta) + B_2(s,\theta)N_0(\theta) \\ &\quad + \int_a^s B_1(s,\xi)N_2(\xi,\theta)d\xi + \int_s^\theta B_2(s,\xi)N_2(\xi,\theta)d\xi \\ &\quad + \int_\theta^b B_2(s,\xi)N_1(\xi,\theta)d\xi \end{aligned}$$

Proof. Proof Omitted.

An interesting corollary of this theorem is that if either $B_0 = 0$ or $N_0 = 0$, then $R_0 = 0$.

Notation To avoid writing out cumbersome integrals, we will use the notation

$$\{R_0,R_1,R_2\} = \{B_0,B_1,B_2\} \times \{N_0,N_1,N_2\}$$

to mean that the functions $\{R_0,R_1,R_2\}$ satisfy Eqns. (2).

3.2 The Adjoint of M, N_1, N_2 operators

Lemma 1. 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

$$\hat{N}_0(s) = N_0(s)^T, \quad \hat{N}_1(s,\eta) = N_2(\eta,s)^T$$
$$\hat{N}_2(s,\eta) = N_1(\eta,s)^T. \tag{3}$$

Proof. Proof Omitted.

Notation We will use the notation

$$\{\hat{N}_0,\hat{N}_1,\hat{N}_2\} = \{N_0,N_1,N_2\}^*$$

to mean that the functions $\{\hat{N}_0,\hat{N}_1,\hat{N}_2\}$ satisfy Eqn. (3).

4. FUNDAMENTAL IDENTITIES

In this section, we show that if

$$B\mathrm{col}\left[x_2(t,a)\ x_2(t,b)\ x_3(t,a)\ x_3(t,b)\ x_{3s}(t,a)\ x_{3s}(t,b)\right] = 0$$

where B is of row rank $n_1 + 2n_2$, then the following identities hold

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

where

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

where the G_i are uniquely determined by the matrix B.

First, we establish the auxiliary identities:

Lemma 2. Suppose that $\mathbf{x} : [a,b] \to \mathbb{R}^n$ is twice continuously differentiable. Then

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

Proof. The proof follows directly from the fundamental theorem of calculus

As an obvious corollary, we have

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

The implication is that any boundary value can be expressed using two other boundary identities. We can now generalize this to the main result.

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

$$B\mathrm{col}\left[x_2(t,a)\ x_2(t,b)\ x_3(t,a)\ x_3(t,b)\ x_{3s}(t,a)\ x_{3s}(t,b)\right] = 0$$

where B is of row rank $n_1 + 2n_2$, then the following identities hold

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

where

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

$$G_0(s) = L_0, \qquad G_1(s,\theta) = L_1(s,\theta) + G_2(s,\theta)$$
$$G_2(s,\theta) = -K(s)(BT)^{-1}BQ(s,\theta)$$
$$G_3(s) = F_0, \qquad G_4(s,\theta) = F_1 + L_1(s,\theta) + G_5(s,\theta)$$
$$G_5(s,\theta) = -V(BT)^{-1}BQ(s,\theta)$$

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

$$K(s) = \begin{bmatrix} 0 & 0 & 0 \\ I & 0 & 0 \\ 0 & I & (s-a) \end{bmatrix}, \quad L_1(s,\theta) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & (s-\theta)I \end{bmatrix}$$

$$L_0 = \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad F_0 = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad F_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & I \end{bmatrix}, \quad V = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & I \end{bmatrix}$$

Proof. Let us define the vectors

$$x_{bf}(t) = \mathrm{col}\,[x_2(t,a)\;\; x_2(t,b)\;\; x_3(t,a)\;\; x_3(t,b)\;\; x_{3s}(t,a)\;\; x_{3s}(t,b)]$$
$$x_{bc}(t) = \mathrm{col}\,[x_2(t,a)\;\; x_3(t,a)\;\; x_{3s}(t,a)]$$

Using Lemma 2, we can express x_{bf} using x_{bc} and $x_{ss}(s)$ as

$$x_{bf}(t) = \mathcal{P}_{\{T,0,0\}} x_{bc}(t) + \mathcal{P}_{\{0,Q,Q\}} \mathbf{x}_f(t).$$

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

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

We may now express the boundary conditions as

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

Since B has row rank $n_2 + 2n_3$, BT is invertible and hence

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

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,

$$\begin{aligned} \mathbf{x}_h &= \mathcal{P}_{\{L_0,0,0\}} x_{bc} + \mathcal{P}_{\{F_0,F_1,0\}} \mathbf{x}_f \\ &= \mathcal{P}_{\{F_0,F_1,0\}} \mathbf{x} - \mathcal{P}_{\{L_0,0,0\}} \mathcal{P}_{\{0,L_0(BJ_0)^{-1}BN_1,L_0(BJ_0)^{-1}BN_1\}} \mathbf{x}_f \\ &= \mathcal{P}_{\{F_0,F_1-V(BT)^{-1}BQ,-V(BT)^{-1}BQ\}} \mathbf{x}_f = \mathcal{P}_{\{G_3,G_4,G_5\}} \mathbf{x}_f. \end{aligned}$$

We refer to x_{bc} as the "core" boundary conditions. Given these, $\mathbf{x}_p$ and $\mathbf{x}_h$ are the combination of a uniquely determined semiseparable operator acting on the "fundamental" state $\mathbf{x}_f$ and a separable operator, determined by the true boundary conditions, also acting on the fundamental state.

These relationships imply that if we can solve for $\mathbf{x}_f$, then we can reconstruct the full solution $\mathbf{x}_p$. Furthermore, stability of $\mathbf{x}_f$ clearly implies stability of $\mathbf{x}_p$. Note that the converse is also true, the transformation is invertible, where obviously, we may differentiate $\mathbf{x}_p$ to obtain $\mathbf{x}_f$. This implies there is a one-to-one relationship between the subspace $L_2 \times H^1 \times H^2$ which satisfy the boundary conditions and the entire space L_2.

With this in mind, let us re-examine the dynamics of the original PDE to see whether these can be expressed using online the fundamental state, $\mathbf{x}_f$.

4.1 Expression for the Fundamental Dynamics

Lemma 3. If

$$\begin{bmatrix} x_1(t,s) \\ x_2(t,s) \\ x_3(t,s) \end{bmatrix}_t = A_0(s) \begin{bmatrix} x_1(t,s) \\ x_2(t,s) \\ x_3(t,s) \end{bmatrix} + A_1(s) \begin{bmatrix} x_2(t,s) \\ x_3(t,s) \end{bmatrix}_s + A_2(s) \begin{bmatrix} x_3(t,s) \end{bmatrix}_{ss}$$

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

$$B\mathrm{col}\,[x_2(t,a)\;\; x_2(t,b)\;\; x_3(t,a)\;\; x_3(t,b)\;\; x_{3s}(t,a)\;\; x_{3s}(t,b)] = 0. \quad (4)$$

Then

$$\dot{\mathbf{x}}_p = \mathcal{P}_{\{H_0,H_1,H_2\}} \mathbf{x}_f$$

$$\begin{aligned} H_0(s) &= A_0(s)G_0(s) + A_1(s)G_3(s) + A_{20}(s) \\ H_1(s,\theta) &= A_0(s)G_1(s,\theta) + A_1(s)G_4(s,\theta), \\ H_2(s,\theta) &= A_0(s)G_2(s,\theta) + A_1(s)G_5(s,\theta), \\ A_{20}(s) &= [0\;\; 0\;\; A_2(s)] \end{aligned}$$

where the G_i are as defined in Theorem 2.

Proof. Follows from Theorem 2 and Theorem 1.

This representation of the dynamics is useful in that we no longer need to account for the boundary conditions. This begs the question of whether the dynamics may be written solely in terms of $\mathbf{x}_f$. Clearly, we have

$$\mathcal{P}_{\{G_0,G_1,G_2\}} \mathbf{x}_{f,t} = \mathcal{P}_{\{H_0,H_1,H_2\}} \mathbf{x}_f.$$

However, this is an integro-differential equation and hence somewhat difficult to study. Furthermore, in this case, we cannot simply invert $\mathcal{P}_{\{G_0,G_1,G_2\}}$, as this would lead to a differential operator and $\mathbf{x}_f$ is not necessarily differentiable. For this reason, we take a hybrid approach and express our stability conditions using a Lyapunov function defined on $\mathbf{x}_p$. Having defined our Lyapunov function, we proceed to take the derivative of the function and reformulate this derivative solely in terms of $\mathbf{x}_f$. However, before we express these stability conditions, we propose an LMI for ensuring positivity of operators of the $\mathcal{P}_{\{M,N_1,N_2\}}$ class when the functions M, N_1, N_2 are polynomial.

5. POSITIVITY OF OPERATORS

In the following section, we will show how to represent our Lyapunov stability conditions as positivity of operators of the form $\mathcal{P}_{\{M,N_1,N_2\}}$. First, however, we show how to use LMIs to enforce positivity of these operators when M, N_1 and N_2 are polynomials. This is a slight generalization of the result in Peet (2014).

Theorem 3. For any square-integrable functions $Z(s)$ and $Z(s,\theta)$, if $g(s) \geq 0$ for all $s \in [a,b]$ and

$$\begin{aligned} &M(s) = g(s)Z(s)^T P_{11} Z(s) \\ &N_1(s,\theta) = g(s)Z(s)^T P_{12} Z(s,\theta) + g(\theta)Z(\theta,s)^T P_{31} Z(\theta) \\ &+ \int_a^\theta g(\nu)Z(\nu,s)^T P_{33} Z(\nu,\theta) d\nu + \int_\theta^s g(\nu)Z(\nu,s)^T P_{32} Z(\nu,\theta) d\nu \\ &+ \int_s^L g(\nu)Z(\nu,s)^T P_{22} Z(\nu,\theta) d\nu \\ &N_2(s,\theta) = g(s)Z(s)^T P_{13} Z(s,\theta) + g(\theta)Z(\theta,s)^T P_{21} Z(\theta) \\ &+ \int_a^s g(\nu)Z(\nu,s)^T P_{33} Z(\nu,\theta) d\nu + \int_s^\theta g(\nu)Z(\nu,s)^T P_{23} Z(\nu,\theta) d\nu \\ &+ \int_\theta^L g(\nu)Z(\nu,s)^T P_{22} Z(\nu,\theta) d\nu, \end{aligned}$$

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 $\mathcal{P}^*_{\{M,N_1,N_2\}} = \mathcal{P}_{\{M,N_1,N_2\}}$ and $\left\langle \mathbf{x}, \mathcal{P}_{\{M,N_1,N_2\}} \mathbf{x} \right\rangle_{L_2} \geq 0$ for all $\mathbf{x} \in L_2[a,b]$.

Proof. Proof Omitted.

A typical choice for Z is a vector of monomials. 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 $[a,b]$. In practice, and motivated by Positivstellensatz-type results, we combine both choices for g. For convenience, now define the set of functions which satisfy Theorem 3 in this way. Specifically, we denote $Z_d(x)$ as the vector of monomials of degree d or less and define the cone of positive operators with polynomial multipliers and kernels associated with degree d as

$$\begin{aligned}\Phi_d :=\{&(M,N_1,N_2) : \\ &(M,N_1,N_2) = (M_a,N_{1a},N_{2a}) + (M_b,N_{1b},N_{2b}) \\ &\text{where } (M_a,N_{1a},N_{2a}) \text{ and } (M_b,N_{1b},N_{2b}) \text{ satisfy} \\ &\text{the conditions of Thm. 3 with } Z = Z_d \text{ and} \\ &\text{where } g(s)=1 \text{ and } g(s)=(s-a)(b-s), \text{ resp.}\}\end{aligned} \tag{5}$$

where the dimension of the matrices M, N_1 and N_2 should be clear from context. The constraint $(M,N_1,N_2) \in \Phi_d$ may thus be considered an LMI constraint. A Matlab toolbox for enforcing this LMI constraint is discussed in Section 8.

6. LYAPUNOV STABILITY CONDITIONS

Using the $\mathcal{P}_{\{M,N_1,N_2\}}$ parameterization of operators, we may now succinctly represent our Lyapunov Stability conditions. The procedure is relatively straightforward. We propose a Lyapunov function of the form

$$V(\mathbf{x}_p) = \left\langle \mathbf{x}_p, \mathcal{P}_{\{M,N_1,N_2\}}\mathbf{x}_p \right\rangle_{L_2}$$

such that $\mathcal{P}^*_{\{M,N_1,N_2\}} = \mathcal{P}_{\{M,N_1,N_2\}}$. The derivative of the Lyapunov function is

$$\begin{aligned}\dot V(\mathbf{x}_p) &= \left\langle \dot{\mathbf{x}}_p, \mathcal{P}_{\{M,N_1,N_2\}}\mathbf{x}_p \right\rangle + \left\langle \mathbf{x}_p, \mathcal{P}_{\{M,N_1,N_2\}}\dot{\mathbf{x}}_p \right\rangle \\ &= \left\langle \mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f, \mathcal{P}_{\{M,N_1,N_2\}}\mathbf{x}_p \right\rangle \\ &\quad + \left\langle \mathbf{x}_p, \mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f \right\rangle \\ &= \left\langle \mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f, \mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f \right\rangle \\ &\quad + \left\langle \mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f, \mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f \right\rangle \\ &= \left\langle \mathbf{x}_f, \mathcal{P}^*_{\{H_0,H_1,H_2\}}\mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_{ss} \right\rangle \\ &\quad + \left\langle \mathbf{x}_f, \mathcal{P}^*_{\{G_0,G_1,G_2\}}\mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f \right\rangle \\ &= \left\langle \mathbf{x}_f, \mathcal{P}_{\{K_0,K_1,K_2\}}\mathbf{x}_f \right\rangle + \left\langle \mathbf{x}_f, \mathcal{P}^*_{\{K_0,K_1,K_2\}}\mathbf{x}_f \right\rangle.\end{aligned}$$

If we then constrain $\mathcal{P}_{\{M,N_1,N_2\}} > 0$ and $\mathcal{P}_{\{K_0,K_1,K_2\}} < 0$, then by standard Lyapunov arguments we have stability.

Theorem 4. Suppose there exist $\epsilon, \epsilon_2 > 0$, $d \in \mathbb{Z}$, $M : [a,b] \to \mathbb{R}^{n\times n}$, $N_1, N_2 : [a,b]^2 \to \mathbb{R}^{n\times n}$ such that

$$(M - \epsilon I, N_1, N_2) \in \Phi_d$$

and

$$-\{K_0,K_1,K_2\} - \{K_0,K_1,K_2\}^* - \epsilon_2\{T_0,T_1,T_2\} \in \Phi_d$$

where

$$\begin{aligned}\{K_0,K_1,K_2\} &= \{H_0,H_1,H_2\}^* \times \{J_0,J_1,J_2\} \\ \{J_0,J_1,J_2\} &= \{M,N_1,N_2\}^* \times \{G_0,G_1,G_2\} \\ \{T_0,T_1,T_2\} &= \{G_0,G_1,G_2\}^* \times \{G_0,G_1,G_2\}\end{aligned}$$

where G_i are as defined in Thm. 2 and H_i are as defined in Lem. 3. Then any solution of Eqns. (1) and (4) is exponentially stable.

Proof. Define the Lyapunov Functional

$$\begin{aligned}V(\mathbf{x}_p) &= \left\langle \mathbf{x}_p, \mathcal{P}_{\{M,N_1,N_2\}}\mathbf{x}_p \right\rangle \\ &= \left\langle \mathbf{x}_p, \mathcal{P}_{\{M-\epsilon I,N_1,N_2\}}\mathbf{x}_p \right\rangle + \epsilon\|\mathbf{x}_p\|^2_{L_2} \ge \epsilon\|\mathbf{x}_p\|^2_{L_2}\end{aligned}$$

Then

$$\begin{aligned}\dot V(\mathbf{x}) &= \left\langle \mathbf{x}_p, \mathcal{P}_{\{M,N_1,N_2\}}\dot{\mathbf{x}}_p \right\rangle + \left\langle \dot{\mathbf{x}}_p, \mathcal{P}_{\{M,N_1,N_2\}}\mathbf{x}_p \right\rangle \\ &= \left\langle \mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f, \mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f \right\rangle \\ &\quad + \left\langle \mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f, \mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f \right\rangle.\end{aligned}$$

Examining the second term, we have by Theorem 1

$$\begin{aligned}&\left\langle \mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f, \mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f \right\rangle \\ &= \left\langle \mathbf{x}_f, \mathcal{P}^*_{\{H_0,H_1,H_2\}}\mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f \right\rangle \\ &= \left\langle \mathbf{x}_f, \mathcal{P}^*_{\{H_0,H_1,H_2\}}\mathcal{P}_{\{J_0,J_1,J_2\}}\mathbf{x}_f \right\rangle\end{aligned}$$

where $\{J_0,J_1,J_2\} = \{M,N_1,N_2\}^* \times \{G_0,G_1,G_2\}$. Then, by Lemma 1 and Theorem 1, we have

$$\left\langle \mathbf{x}_f, \mathcal{P}^*_{\{H_0,H_1,H_2\}}\mathcal{P}_{\{J_0,J_1,J_2\}}\mathbf{x}_f \right\rangle = \left\langle \mathbf{x}_f, \mathcal{P}_{\{K_0,K_1,K_2\}}\mathbf{x}_f \right\rangle$$

where $\{K_0,K_1,K_2\} = \{H_0,H_1,H_2\}^* \times \{J_0,J_1,J_2\}$. By symmetry and Lemma 1, we have for the first term,

$$\begin{aligned}&\left\langle \mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f, \mathcal{P}_{\{M,N_1,N_2\}}\mathcal{P}_{\{H_0,H_1,H_2\}}\mathbf{x}_f \right\rangle \\ &= \left\langle \mathbf{x}_f, \mathcal{P}^*_{\{K_0,K_1,K_2\}}\mathbf{x}_f \right\rangle = \left\langle \mathbf{x}_f, \mathcal{P}_{\{\hat K_0,\hat K_1,\hat K_2\}}\mathbf{x}_f \right\rangle.\end{aligned}$$

Therefore, by Lemma 1 and Theorem 1,

$$\begin{aligned}\dot V(\mathbf{x}_p) &= \left\langle \mathbf{x}_f, \mathcal{P}_{\{K_0+\hat K_0,K_1+\hat K_1,K_2+\hat K_2\}}\mathbf{x}_f \right\rangle \\ &= \left\langle \mathbf{x}_f, \mathcal{P}_{\{K_0+\hat K_0,K_1+\hat K_1,K_2+\hat K_2\}}\mathbf{x}_f \right\rangle \\ &\quad + \epsilon\left\langle \mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f, \mathcal{P}_{\{G_0,G_1,G_2\}}\mathbf{x}_f \right\rangle - \epsilon\|\mathbf{x}_p\|^2_{L_2} \\ &= \left\langle \mathbf{x}_f, \mathcal{P}_{\{K_0+\hat K_0,K_1+\hat K_1,K_2+\hat K_2\}}\mathbf{x}_f \right\rangle \\ &\quad + \epsilon\left\langle \mathbf{x}_f, \mathcal{P}_{\{T_0,T_1,T_2\}}\mathbf{x}_f \right\rangle - \epsilon\|\mathbf{x}_p\|^2_{L_2} \le -\epsilon\|\mathbf{x}_p\|^2_{L_2}.\end{aligned}$$

We conclude exponential stability.

7. SOFTWARE IMPLEMENTATION

In this section, we examine the accuracy and computational complexity of the proposed stability algorithm by applying the results to several well-studied and relatively trivial test cases. The algorithms are implemented using a Matlab toolbox which is an adaptation of SOSTOOLS Prajna et al. (2002) and which can be found online at `http://control.asu.edu` and on Code Ocean.

For convenience we here describe several functions in the code which implement operations described in this paper. In each case, we give the notation presented followed by the Matlab function implementing the notation.

Notation and Associated Code:

$$\{M,N_1,N_2\} \in \Phi_d \qquad \to \qquad \mathcal{P}_{\{M,N_1,N_2\}} \ge 0$$

```
[prog, M, N1, N2]= sosjointpos_mat_ker_semisep(prog,n,d,d,s,th,[a,b])
```

$$\begin{aligned}\{M,N_1,N_2\} &= \{T_0,T_1,T_2\} \times \{R_0,R_1,R_2\} \\ &\to \quad \mathcal{P}_{\{M,N_1,N_2\}} = \mathcal{P}_{\{T_0,T_1,T_2\}}\mathcal{P}_{\{R_0,R_1,R_2\}}\end{aligned}$$

```
[M, N1, N2] = semisep_MN1N2_compose(T0,T1,T2,R0,R1,R2,s,th,[a,b])
```

$$\{M,N_1,N_2\} = \{T_0,T_1,T_2\}^* \to \mathcal{P}_{\{M,N_1,N_2\}} = \mathcal{P}^*_{\{T_0,T_1,T_2\}}$$

```
[M, N1, N2] = semisep_MN1N2_transpose(T0,T1,T2,s,th)
```

Equipped with these Matlab functions, we can give an almost complete implementation of the stability test as

```
pvar s th
[prog, G0, G1, G2]=...
[prog, H0, H1, H2]=...
prog = sosprogram([s th])
```

```
[prog, M, N1, N2]= sosjointpos_mat_ker_semisep(prog,n,d,d,s,th,II)
[J0, J1, J2] = semisep_MN1N2_compose(M+ep*I,N1,N2,G0,G1,G2,s,th,II)
[H0s, H1s, H2s] = semisep_MN1N2_transpose(H0,H1,H2,s,th)
[K0, K1, K2] = semisep_MN1N2_compose(H0s,H1s,H2s,J0,J1,J2,s,th,II)
[K0s, K1s, K2s] = semisep_MN1N2_transpose(K0,K1,K2,s,th)
[prog, [],N1e, N2e] = sosjointpos_mat_ker_semisep(prog,n,d+2,d+2,s,th,II)
[prog, [],gN1e, gN2e] = sosjointpos_mat_ker_semisep_psatz(prog,n,d+2,d+2,s,th,II)
[prog] = sosmateq(prog,K1+K1s+N1eq+gN1eq)
prog = sossolve(prog,pars)
```

7.1 Demonstration of Accuracy

We now give several examples which show the stability test is not conservative in any significant sense.

Example 1: We begin with several variations of the diffusion equation. The first is adapted from Valmorbida et al. (2014).

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

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

Example 2: The second example from Valmorbida et al. (2016) 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 Gahlawat and Peet is not homogeneous

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

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 Valmorbida et al. (2014), we have

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

with $x(0) = 0$ and $x_s(1) = 0$. In this case, using $d = 1$, we can prove stability for $R = 2.93$ (improvement over $R = 2.45$ in Valmorbida et al. (2014)) with a computation time of $1.21s$.

Example 5: In this example from Valmorbida et al. (2016), 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 $4.06s$.

7.2 Demonstration of Accuracy

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 for different size problems, from $n = 1$ to $n = 20$.

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

8. ILLUSTRATION BY EXAMPLE

In this section, we use several well-known examples to illustrate the process by which these problems are posed in the universal framework proposed in this paper. There are several significant questions to keep in mind when constructing the universal formulation.

- What are the states? e.g. Is the correct state u or u_s?
- What are the boundary conditions?

Choice of State: The choice of states determines not only the complexity of the computation, but feasibility of the stability. This is because many PDEs are exponentially stable with respect to some states, but not others. Ideally, we would use the fundamental state, $\mathbf{x}_f$, as stability in this state implies stability in the primal state, $\mathbf{x}_p$. However, as mentioned earlier, it is not generally possible to express the dynamics of $\mathbf{x}_f$ in the form of Equation (1). As a rule of thumb, it is generally better to use states such as u_s instead of u, as stability in u_s implies stability in u. Moreover, as we will see, the choice of state is limited by the boundary conditions imposed.

Boundary Conditions: Identification of boundary conditions in the universal framework is particularly important, as the B matrix must have sufficient rank. Boundary conditions represent redundant information and hence implicit constraints on the primal state $\mathbf{x}_p$. The presence of a boundary condition also restricts the choice of state directly. For example, if there is a boundary condition on u_t, then u_t must appear as a state (or u_{st}, in which case the boundary conditions is removed). Furthermore, if there is a boundary condition on the spatial derivative u_s, then either u_s must be a state ($u_s \in x_2$) or u must be a diffusive state ($u \in x_3$). In the following examples, we illustrate the process of choosing state and constructing the A_i and B matrices.

8.1 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.

8.1.0.1. Euler-Bernoulli In this case, we consider the simplest expression of the cantilevered E-B beam:

$$\begin{aligned}&u_{tt}(t,s) = -cu_{xxxx}(t,s), \qquad \text{where}\\ &u(0) = u_x(0) = u_{xx}(L) = u_{xxx}(L) = 0.\end{aligned}$$

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

$$\begin{aligned}\dot{u}_1 &= u_{tt} = -cu_{xxxx} = -cu_{2xx}\\ \dot{u}_2 &= u_{txx} = u_{1xx}.\end{aligned}$$

These equations are now in the universal form

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

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

$$u_{xx}(L) = u_2(L) = 0 \quad \text{and} \quad u_{xxx}(L) = u_{2x}(L) = 0.$$

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

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

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

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

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_{xx}.

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

$$\ddot{w} = \partial_x(w_x - \phi) \qquad = -\phi_x + w_{xx}$$
$$\ddot{\phi} = \phi_{xx} + (w_x - \phi) \qquad = -\phi + w_x + \phi_{xx}$$

with boundary conditions of the form

$$\phi(0) = 0, \quad w(0) = 0, \quad \phi_x(L) = 0, \quad w_x(L) - \phi(L) = 0.$$

As before, 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 $u_2 = w_x - \phi$ and $u_4 = \phi_x$. 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

$$u_{1t} = u_{2x}, \qquad u_{2t} = u_{1x} - u_3$$
$$u_{3t} = u_{4x} + u_2, \qquad u_{4t} = u_{3x}.$$

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

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

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

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

where B has row rank $n_2 = 4$. The code indicates this system is stable (using $\varepsilon_2 = 0$). However, when $\varepsilon_2 > 0$, the code is unable to find a Lyapunov function, indicating this formulation is probably not be exponentially stable in all the given states. 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 $u_2 = w_x$ and $u_4 = \phi$. This leads to a mixed hyperbolic-diffusive formulation where

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}_t = \underbrace{\begin{bmatrix} 0&0&0&0 \\ 0&0&0&0 \\ 0&1&0&-1 \\ 0&0&1&0 \end{bmatrix}}_{A_0} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} + \underbrace{\begin{bmatrix} 0&1&0&-1 \\ 1&0&0&0 \\ 0&0&0&0 \\ 0&0&0&0 \end{bmatrix}}_{A_1} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}_x + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}}_{A_2} u_{4xx}$$

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

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

This formulation, however, does not appear to be stable in the given states. Since the only new state is ϕ, we may test this hypothesis 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

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

The code indicates that this formulation is now stable for any $c > 0$. These examples indicates the sensitivity of PDE models to the definition of stability - something that seems especially critical in wave-type PDEs.

8.2 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 Chen (1979); Datko et al. (1986). 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

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

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_x for $k > 0$. We now consider variations on this formulation.

Diffusive Formulation As a first variation, we consider a non-diffusive formulation from Chen (1979) which was shown to be asymptotically stable in the state u for $a^2 + k^2 > 0$.

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

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

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

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! These boundary conditions are expressed as

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

Computation indicates this model is stable, but not exponentially stable in the given state - a result confirmed in Chen (1979); Datko et al. (1986).

9. CONCLUSION

In this paper, we have shown that stability of a large class of PDE systems can be represented compactly in LMI form using a variation of Sum-of-Squares optimization. To achieve this result, we proposed that the state of a PDE of the form of Equation (1) is actually $\mathbf{x}_f$ and that all Lyapunov stability conditions may be represented on this state. A SOS-style algorithm to test these Lyapunov conditions is proposed and numerical examples indicate no conservatism in the stability conditions to at least 5 significant figures even for low polynomial degree.

REFERENCES

Aamo, O. (2013). Disturbance rejection in 2 x 2 linear hyperbolic systems. *IEEE Transactions on Automatic Control*, 58(5), 1095–1106.

Ahmadi, M., Valmorbida, G., and Papachristodoulou, A. (2016). Dissipation inequalities for the analysis of a class of PDEs. *Automatica*, 66, 163–171.

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

Chen, G. (1979). Energy decay estimates and exact boundary value controllability for the wave equation in a bounded domain. *J. Math. Pures Appl.*, 58, 249–273.

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

Datko, R., Lagnese, J., and Polis, M. (1986). An example on the effect of time delays in boundary feedback stabilization of wave equations. *SIAM Journal on Control and Optimization*, 24(1), 152–156.

Fridman, E. and Orlov, Y. (2009). An LMI approach to H_∞ boundary control of semilinear parabolic and hyperbolic systems. *Automatica*, 45(9), 2060–2066.

Fridman, E. and Terushkin, M. (2016). New stability and exact observability conditions for semilinear wave equations. *Automatica*, 63, 1–10.

Gahlawat, A. and Peet, M. (????). A convex sum-of-squares approach to analysis, state feedback and output feedback control of parabolic pdes. *IEEE Transactions on Automatic Control*, 62(4).

Gahlawat, A. and Peet, M. (2015). 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*.

Gahlawat, A. and Peet, M. (2016). Optimal state feedback boundary control of parabolic PDEs using SOS polynomials. In *Proceedings of the American Control Conference*.

Gaye, O., Autrique, L., Orlov, Y., Moulay, E., Brémond, S., and Nouailletas, R. (2013). h_∞ stabilization of the current profile in tokamak plasmas via an LMI approach. *Automatica*, 49(9), 2795–2804.

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

Marion, M. and Temam, R. (1989). Nonlinear Galerkin methods. *SIAM Journal on numerical analysis*, 26(5), 1139–1157.

Papachristodoulou, A. and Peet, M.M. (2006). On the analysis of systems described by classes of partial differential equations. In *Proceedings of the IEEE Conference on Decision and Control*, 747.

Peet, M. (2014). LMI parameterization of Lyapunov functions for infinite-dimensional systems: A toolbox. In *Proceedings of the American Control Conference*.

Peet, M. (2018). 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*.

Prajna, S., Papachristodoulou, A., and Parrilo, P.A. (2002). Introducing SOSTOOLS: a general purpose sum of squares programming solver. *Proceedings of the IEEE Conference on Decision and Control*.

Ravindran, S. (2000). A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. *International journal for numerical methods in fluids*, 34(5), 425–448.

Rowley, C. (2005). Model reduction for fluids, using balanced proper orthogonal decomposition. *International Journal of Bifurcation and Chaos*, 15(03), 997–1013.

Smyshlyaev, A. and Krstic, M. (2005). Backstepping observers for a class of parabolic pdes. *Systems & Control Letters*, 54(7), 613–625.

Solomon, O. and Fridman, E. (2015). Stability and passivity analysis of semilinear diffusion pdes with time-delays. *International Journal of Control*, 88(1), 180–192.

Valmorbida, G., Ahmadi, M., and Papachristodoulou, A. (2014). Semi-definite programming and functional inequalities for distributed parameter systems. In *Proceedings of the IEEE Conference on Decision and Control*, 4304–4309.

Valmorbida, G., Ahmadi, M., and Papachristodoulou, A. (2016). Stability analysis for a class of partial differential equations via semidefinite programming. *IEEE Transactions on Automatic Control*, 61(6), 1649–1654.