Partial Integral Equations (PIEs) Part 2: Estimation and Control of PIEs

Matthew M. Peet (w. S. Shivakumar, A. Das, S. Weiland, K. Gu, S. Wu)
Arizona State University
Tempe, AZ USA

SIAM CT19
Chengdu, China

June 21, 2019

Dynamics are usually expressed in the Primal State $\mathbf{x}_p \in X_p$:

$$\mathbf{x}_p \in L^2_{n_1} \times H^1_{n_2} \times H^2_{n_3} := X_p$$

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

Boundary Conditions:

$$B \begin{bmatrix} x_2(0) \\ x_2(L) \\ x_3(0) \\ x_3(L) \\ x_{3s}(0) \\ x_{3s}(L) \end{bmatrix} = 0, \quad \operatorname{rank}(B) = n_2 + 2n_3$$

Euler-Bernoulli Beam:

$$\mathbf{u}_t = \underbrace{\begin{bmatrix} 0 & -c \\ 1 & 0 \end{bmatrix}}_{=A_2 \; (A_0 = A_1 = 0)} \mathbf{u}_{ss}$$

State Space: $u \in H^2_2$:

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

Illustration 1: The Euler-Bernoulli Beam

Consider a simple cantilevered E-B beam:

$$u_{tt}(t,s) = -cu_{ssss}(t,s), \qquad \text{where} \quad u(0) = u_s(0) = u_{ss}(L) = u_{sss}(L) = 0$$

Define the States: Let

$$u_1 = u_t, \qquad \text{and} \qquad u_2 = u_{ss}$$

Define the Dynamics:

$$\dot{u}_1 = u_{tt} = -cu_{ssss} = -cu_{2ss}, \qquad \dot{u}_2 = u_{tss} = u_{1ss}.$$

Universal Formulation:

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

where $A_0 = A_1 = 0$, $n_3 = 2$, and $n_1 = n_2 = 0$.

Boundary Conditions:

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

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

This yields $rank(B) = 4$

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

Partial Integral Equations (PIEs)

An **ALGEBRAIC** Representation of PDEs

Original Form:

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

PIE Format: Write the PDE as a Partial Integral Equation!

$$\mathcal{T}\dot{\mathbf{x}}_f(t) = \mathcal{A}\mathbf{x}_f(t) + \mathcal{B}w(t) \qquad \mathbf{x}_f(t,s) := \begin{bmatrix} x_1(t,s) \\ x_{2s}(t,s) \\ x_{3ss}(t,s) \end{bmatrix}$$

where $\mathcal{T}, \mathcal{A}, \mathcal{B}$ are 3-PIE Operators (bounded).

3-PIE Operators ($\{N_i\}$):

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

Examples of Messy PIE Representation (no BCs)

Heat Equation: $\dot{\mathbf{u}}(t,s) = \mathbf{u}_{ss}(t,s),\ \mathbf{u}(t,0) = \mathbf{u}_s(t,0) = 0$

Messy:

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

Clean:

$$\mathcal{P}_{\{0,s-\eta,0\}}\dot{\mathbf{u}}(t) = \mathcal{P}_{\{I,0,0\}}\mathbf{u}(t)$$

No Partial Derivatives: $\dot{\mathbf{u}}(t,s) = \mathbf{u}(t,s),\ \mathbf{u}(t,0) = w_1(t),\ \mathbf{u}_s(t,0) = w_2(t)$

Messy:

$$\int_0^s (s-\eta)\dot{\mathbf{u}}_{ss}(t,\eta)d\eta = \int_0^s (s-\eta)\mathbf{u}_{ss}(t,\eta)d\eta + s(w_1(t)-\dot{w}_1(t)) + (w_2(t)-\dot{w}_2(t))$$

Clean:

$$\mathcal{P}_{\{0,s-\eta,0\}}\dot{\mathbf{u}}(t) = \mathcal{P}_{\{0,s-\eta,0\}}\mathbf{u}(t) + \mathcal{P}_{\{[s \quad 1],0,0\}}\begin{bmatrix} w_1(t)-\dot{w}_1(t) \\ w_2(t)-\dot{w}_2(t)\end{bmatrix}$$

A UNIVERSAL Transformation from PDE to PIE

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

Boundary Conditions:

$$B \begin{bmatrix} x_2(0) \\ x_2(L) \\ x_3(0) \\ x_3(L) \\ x_{3s}(0) \\ x_{3s}(L) \end{bmatrix} = 0, \quad \mathrm{rank}(B) = n_2 + 2n_3 \qquad \mathbf{x}_f := \begin{bmatrix} x_1(t) \\ x_{2s}(t) \\ x_{3ss}(t) \end{bmatrix}$$

Becomes:

$$\mathcal{E}\dot{\mathbf{x}}_f = \mathcal{A}\mathbf{x}_f(t), \qquad \mathcal{E} = \mathcal{P}_{\{G_i\}}, \qquad \mathcal{A} = \mathcal{P}_{\{J_i\}}$$

Where

$$J_0(s) = A_0(s)G_0(s) + A_1(s)G_3(s) + A_{20}(s), \qquad J_1(s,\theta) = A_0(s)G_1(s,\theta) + A_1(s)H_0(s,\theta),$$

$$J_2(s,\theta) = A_0(s)G_2(s,\theta) + A_1(s)H_1(s,\theta), \qquad A_{20}(s) = \begin{bmatrix} 0 & 0 & A_2(s) \end{bmatrix}$$

$$G_0(s) = L_0, \qquad G_1(s,\theta) = L_1(s,\theta) + G_2(s,\theta), \qquad G_2(s,\theta) = -K(s)(BT)^{-1}BQ(s,\theta)$$

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

where

$$T = \begin{bmatrix} I & 0 & 0 \\ I & 0 & 0 \\ 0 & I & 0 \\ 0 & I & (b-a)I \\ 0 & 0 & I \\ 0 & 0 & I \end{bmatrix}, \quad Q(s,\theta) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \\ 0 & 0 & (b-\theta)I \\ 0 & 0 & 0 \\ 0 & 0 & I \end{bmatrix}, \quad K(s) = \begin{bmatrix} I & 0 & 0 \\ I & 0 & 0 \\ 0 & I & (s-a) \end{bmatrix}, \quad L_0 = \begin{bmatrix} I & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$L_1(s,\theta) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & (s-\theta)I \end{bmatrix}, \quad F_0 = \begin{bmatrix} 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad F_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & I \end{bmatrix}, \quad V = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & I \end{bmatrix}$$

The Need for 4-PIE operators: A Time-Delay System

$$\begin{bmatrix} \dot{x}(t) \\ \dot{\phi}(t,s) \end{bmatrix} = \begin{bmatrix} A_0 x(t) + A_1 \phi(t,-\tau) \\ \phi_s(t,s) \end{bmatrix} + \begin{bmatrix} Bw(t) \\ 0 \end{bmatrix} \qquad \phi(t,0) = x(t)$$

$$y(t) = C_0 x(t) + C_1 \phi(t,-\tau) + Dw(t)$$

4-PIE Representation of a Time-Delay System:

$$\mathcal{T}\dot{\mathbf{x}}_f(t) = \mathcal{A}\mathbf{x}_f + \mathcal{B}w(t) \qquad \mathcal{T}\mathbf{x}_f(t) = \mathbf{x}_p, \qquad y(t) = \mathcal{C}\mathbf{x}_f(t) + \mathcal{D}w(t)$$

$$\mathcal{T} = \mathcal{P}\left\{ {}^{I,\ 0}_{I,\ \{0,0,-I\}} \right\},\ \mathcal{A} = \mathcal{P}\left\{ {}^{A_0 + A_1,\ -A_1}_{0,\ \{I,0,0\}} \right\}, \quad \mathbf{x}_f(t) := \begin{bmatrix} x(t) \\ \phi_s(s,t) \end{bmatrix}$$

$$\mathcal{C} = \mathcal{P}\left\{ {}^{C_0 + C_1,\ -C_1}_{\emptyset,\ \{\emptyset\}} \right\},\ \mathcal{B} = \mathcal{P}\left\{ {}^{B,\ \emptyset}_{0,\ \{\emptyset\}} \right\},\ \mathcal{D} = \mathcal{P}\left\{ {}^{D,\ \emptyset}_{\emptyset,\ \{\emptyset\}} \right\}$$

4-PIE Operators $\mathcal{P} : \mathbb{R}^p \times L_2^q \to \mathbb{R}^m \times L_2^n$

$$\left(\mathcal{P}\left\{ {}^{P,\ Q_1}_{Q_2,\ \{R_i\}} \right\} \begin{bmatrix} x \\ \mathbf{x} \end{bmatrix} \right)(s) := \begin{bmatrix} Px + \int_{-\tau}^{0} Q_1(s)\mathbf{x}(s)ds \\ Q_2(s)x + \left(\mathcal{P}_{\{R_i\}}\mathbf{x}\right)(s) \end{bmatrix}.$$

4-PIE Operators Include a 3-PIE Operator

4-PIE Operators in a Matlab Structure

A general operator on $\mathcal{P}\left\{\begin{smallmatrix}P, & Q_1\\ Q_2, & \{R_i\}\end{smallmatrix}\right\} : \mathbb{R}^p \times L_2^q[a,b] \to \mathbb{R}^m \times L_2^n[a,b]$

$$\left(\mathcal{P}\left\{\begin{smallmatrix}P, & Q_1\\ Q_2, & \{R_i\}\end{smallmatrix}\right\}\begin{bmatrix}x\\ \mathbf{x}\end{bmatrix}\right)(s) := \begin{bmatrix}Px + \int_a^b Q_1(s)\mathbf{x}(s)ds\\ Q_2(s)x + \left(\mathcal{P}_{\{R_i\}}\mathbf{x}\right)(s)\end{bmatrix}.$$

MATLAB structure has following elements.

1. `opvar P`: declares P to be a 4-PIE operator object.
2. `P.P`: a $m \times p$ matrix
3. `P.Q1, P.Q2`: $m \times q$ and $n \times p$ matrix valued polynomials in s, respectively
4. `P.R`: a structure with entities R_0, R_1, and R_2
5. `P.R.R0` : $n \times q$ matrix valued polynomial in s
6. `P.R.R1, P.R.R2` : $n \times q$ matrix valued polynomials in s and θ
7. `P.dim`: $\begin{bmatrix}m & p\\ n & q\end{bmatrix}$.
8. `P.I`: [a,b].
9. `P.var1`: s (default)
10. `P.var2`: th (default)

Composition Formula in the 3-PIE $\mathcal{P}_{\{N_i\}}$ Operator Algebra

Property 1: Composition

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

where

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

Triple Notation:

$$\{R_i\} = \{B_i\} \times \{N_i\}$$

Matlab Implementation:

$$\{N_i\} = \{T_i\} \times \{R_i\} \quad \rightarrow \quad \mathcal{P}_{\{N_i\}} = \mathcal{P}_{\{T_i\}}\mathcal{P}_{\{R_i\}}$$

```
opvar T R
T.R.R0=...; T.R.R1=...; T.R.R2=...; T.dim=[0 0;m n]; T.I=[-tau,0]
R.R.R0=...; R.R.R1=...; R.R.R2=...; R.dim=[0 0;n q]; R.I=[-tau,0]
N=T*R
```

Composition Formula in the 4-PIE Algebra

$$\mathcal{P}_{\left[\substack{L,\ M_1\\ M_2,\ \{N_i\}}\right]}\mathcal{P}_{\left[\substack{P,\ Q_1\\ Q_2,\ \{R_i\}}\right]}\begin{bmatrix} x\\ \mathbf{x}\end{bmatrix}$$

$$=\begin{bmatrix} \left(LP+\int_a^b M_1(s)Q_2(s)\right)x+\int_a^b LQ_1(s)\mathbf{x}(s)ds+\int_a^b M_1(s)\left(\mathcal{P}_{\{R_i\}}\mathbf{x}\right)(s)ds \\ \left(M_2(s)P+\mathcal{P}_{\{N_i\}}Q_2ds\right)x+M_2(s)\int_a^b Q_1(\theta)\mathbf{x}(\theta)d\theta+\left(P_{\{N_i\}}\mathcal{P}_{\{R_i\}}\mathbf{x}\right)(s)(s)\end{bmatrix}$$

Triple-Triple Notation:

$$\begin{bmatrix} P,\ Q_1\\ Q_2,\{R_i\}\end{bmatrix}=\begin{bmatrix} L,\ M_1\\ M_2,\{N_i\}\end{bmatrix}\times\begin{bmatrix} F,\ G_1\\ G_2,\{H_i\}\end{bmatrix}$$

Matlab Implementation:

$$\mathcal{P}\left\{\substack{P,\ Q_1\\ Q_2,\ \{R_i\}}\right\}=\mathcal{P}\left\{\substack{L,\ M_1\\ M_2,\ \{N_i\}}\right\}\mathcal{P}\left\{\substack{F,\ G_1\\ G_2,\ \{H_i\}}\right\}$$

```
opvar T R
T.P=; T.Q1=; T.Q2=; T.R.R0=; T.R.R1=; T.R.R2=; T.dim=[a c;b d]; T.I=;
R.P=; R.Q1=; R.Q2=; R.R.R0=; R.R.R1=; R.R.R2=; R.dim=[c e;d f]; R.I=;
N=T*R
```

Property 2: Transpose/Adjoint

$$\left\langle \mathbf{x}, \mathcal{P}\left\{\begin{smallmatrix}\hat{P}, & \hat{Q}_1\\ \hat{Q}_2, & \{\hat{R}_i\}\end{smallmatrix}\right\}\mathbf{y}\right\rangle_{\mathbb{R}^n\times L_2} = \left\langle \mathcal{P}\left\{\begin{smallmatrix}P, & Q_1\\ Q_2, & \{R_i\}\end{smallmatrix}\right\}\mathbf{x}, \mathbf{y}\right\rangle_{\mathbb{R}^n\times L_2}$$

where

$$\hat{P} = P^T, \quad \hat{Q}_1(s) = Q_2(s)^T, \quad \hat{Q}_2(s) = Q_1(s)^T,$$
$$\hat{R}_0(s) = R_0(s)^T, \quad \hat{R}_1(s,\eta) = R_2(\eta,s)^T, \quad \hat{R}_2(s,\eta) = R_1(\eta,s)^T$$

Triple Notation:

$$\begin{bmatrix}\hat{L}, & \hat{M}_1\\ \hat{M}_2, & \{\hat{N}_i\}\end{bmatrix} = \begin{bmatrix}L, & M_1\\ M_2, & \{N_i\}\end{bmatrix}^*$$

Matlab Implementation:

```
opvar T
T.P=...; T.Q1=...;T.Q2=...; T.R.R0=...; T.R.R1=...; T.R.R2=...;
T.dim=[p q;m n]; T.I=[a,b];
N=T'
```

Note that N.dim will be [q p; n m].

Stability of Coupled ODE-PDE Systems

Armed with PIEs

PIE Dynamics:

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

We now propose a Lyapunov function of the form

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

The time-derivative of the Lyapunov function is

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

Stability Condition: $\mathcal{P} > 0$ and

$$\mathcal{T}^*\mathcal{P}\mathcal{A} + \mathcal{A}^*\mathcal{P}\mathcal{T} \leq 0$$

LMI Equivalent:
Descriptor Systems:

$$E\dot{x}(t) = Ax(t)$$

$$V(x) = x^T E^T P E x$$

$$\begin{aligned}\dot{V}(x) &= \dot{x}^T E^T P E x \\ &\quad + x^T E^T P E \dot{x} \\ &= x^T (E^T P A + A^T P E) x\end{aligned}$$

$$E^T P A + A^T P E < 0$$

Positivity in the 3-PIE N_0, N_1, N_2 Algebra using LMIs

Positivity of 3-PIE Operator is an LMI constraint on the coefficients of the polynomials $\{N_i\}$.

Theorem 1.

For any functions $Z(s)$ and $Z(s,\theta)$, and $g(s) \geq 0$ for all $s \in [a,b]$

$$N_0(s) = g(s)Z(s)^T P_{11} Z(s)$$

$$N_1(s,\theta) = g(s)Z(s)^T P_{12} Z(s,\theta) + g(\theta)Z(\theta,s)^T P_{31} Z(\theta) + \int_a^\theta g(\nu)Z(\nu,s)^T P_{33} Z(\nu,\theta)d\nu$$
$$+ \int_\theta^s g(\nu)Z(\nu,s)^T P_{32} Z(\nu,\theta)d\nu + \int_s^b g(\nu)Z(\nu,s)^T P_{22} Z(\nu,\theta)d\nu$$

$$N_2(s,\theta) = g(s)Z(s)^T P_{13} Z(s,\theta) + g(\theta)Z(\theta,s)^T P_{21} Z(\theta) + \int_a^s g(\nu)Z(\nu,s)^T P_{33} Z(\nu,\theta)d\nu$$
$$+ \int_s^\theta g(\nu)Z(\nu,s)^T P_{23} Z(\nu,\theta)d\nu + \int_\theta^b g(\nu)Z(\nu,s)^T P_{22} Z(\nu,\theta)d\nu,$$

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 $\langle \mathbf{x}, \mathcal{P}_{\{N_i\}}\mathbf{x}\rangle_{L_2} \geq 0$ for all $\mathbf{x} \in L_2[a,b]$.

Positivity in the 4-PIE Algebra using LMIs

Proof of 3-PIE Positivity Thm: Let

$$\{Z_0, Z_1, Z_2\} := \left\{ \underbrace{\begin{bmatrix} \sqrt{g(s)} Z_{d1}(s) \\ \\ \end{bmatrix}}_{Z_0}, \underbrace{\begin{bmatrix} \\ \sqrt{g(s)} Z_{d2}(s,\theta) \\ \\ \end{bmatrix}}_{Z_1}, \underbrace{\begin{bmatrix} \\ \\ \sqrt{g(s)} Z_{d2}(s,\theta) \end{bmatrix}}_{Z_2} \right\}$$

Then

$$\{N_i\} = \{Z_i\}^* \times \{P, 0, 0\} \times \{Z_i\}$$

Triple-Triple Notation:

$$\begin{bmatrix} P, \ Q_1 \\ Q_2, \{R_i\} \end{bmatrix} = \begin{bmatrix} L, \ M_1 \\ M_2, \{N_i\} \end{bmatrix} \times \begin{bmatrix} F, \ G_1 \\ G_2, \{H_i\} \end{bmatrix}$$

Positivity Theorem:

$\begin{bmatrix} P, \ Q_1 \\ Q_2, \{R_i\} \end{bmatrix} \geq 0$ if there exists $P = \begin{bmatrix} P_1 & P_2 \\ P_2^T & P_3 \end{bmatrix} \geq 0$ such that

$$\begin{bmatrix} P, \ Q_1 \\ Q_2, \{R_i\} \end{bmatrix} = \begin{bmatrix} I, \ 0 \\ 0, \{Z_i\} \end{bmatrix}^* \times \begin{bmatrix} P_1, \ P_2 \\ P_2^T, \{P_3, 0, 0\} \end{bmatrix} \times \begin{bmatrix} I, \ 0 \\ 0, \{Z_i\} \end{bmatrix}.$$

Matlab Implementation:

```
[prog, N] = sosposop_RL2RL(prog,[nR nL],X,s,th,[d1 d2]);
```

Matlab Toolbox Implementation (Stability Analysis)

Stability Condition: $\mathcal{P} > 0$ and

$$\mathcal{T}^*\mathcal{P}\mathcal{A} + \mathcal{A}^*\mathcal{P}\mathcal{T} \le 0$$

Almost Complete Matlab Code:

```
pvar s th; opvar A T
A=...
T=...;
X=[a,b];
prog = sosprogram([s th])
[prog, P] = sosposop_RL2RL(prog,[nR nL],X,s,th,[d1 d2]);
[prog, N] = sosposop_RL2RL_noR0(prog,[nR nL],X,s,th,[d1 d2]);
[prog, gN] = sosposop_RL2RL_noR0(prog,[nR nL],X,s,th,[d1 d2]);
[prog] = sosopeq(prog,A'*P*T+T'*P*A+N+gN)
prog = sossolve(prog,pars)
```

Stability Conditions:

$$\{N_i\} - \{\epsilon I, 0, 0\} \in \Phi_d$$

$$\{K_i\} = \{G_i\}^* \times \{N_i\} \times \{H_i\}$$

$$-\{K_i\} - \{K_i\}^* \in \Phi_{d+2}$$

Testing for Accuracy

Example 1: Adapted from Valmorbida, 2014:

$$\dot{x}(t,s) = \lambda x(t,s) + x_{ss}(t,s) \qquad x(0) = x(1) = 0$$

Stable iff $\lambda < \pi^2 \cong 9.8696$. We prove stability for $\lambda = 9.8696$.

Example 2: From Valmorbida, 2016,

$$\dot{x}(t,s) = \lambda x(t,s) + x_{ss}(t,s) \qquad x(0) = 0, \quad x_s(1) = 0$$

Unstable for $\lambda > 2.467$. We prove stability for $\lambda = 2.467$.

Example 3: From Gahlawat, 2017:

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

with $x(0) = 0$ and $x_s(1) = 0$. Unstable for $\lambda > 4.65$. For $d = 1$, we prove stability for $\lambda = 4.65$.

Example 4: From Valmorbida,2014,

$$\dot{x}(t,s) = \begin{bmatrix} 1 & 1.5 \\ 5 & .2 \end{bmatrix} x(t,s) + R^{-1} x_{ss}(t,s), \qquad x(0) = x_s(1) = 0$$

With $d = 1$, we prove stability for $R = 2.93$ (improvement over $R = 2.45$).

Example 5: From Valmorbida,2016,

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

Using $d = 1$, we prove stability for $R = 21$ (and greater) with a computation time of $4.06s$.

Complexity and Accuracy of Dual Stability ($\mathcal{AP} < 0$)

$$\dot{x}(t) = -x(t-\tau)$$

d	1	2	3	4	analytic
$\tau_{\max}$	1.408	1.5707	1.5707	1.5707	1.5707
CPU sec	.18	.21	.25	.47	

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -2 & .1 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} x(t-\tau)$$

d	1	2	3	4	limit
$\tau_{\max}$	1.6581	1.716	1.7178	1.7178	1.7178
$\tau_{\min}$	.10019	.10018	.10017	.10017	.10017
CPU sec	.25	.344	.678	1.725	

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -1 & .1 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 \\ -1 & 0 \end{bmatrix} x(t-\tau/2) + \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} x(t-\tau)$$

d	1	2	3	4	limit
$\tau_{\max}$	1.33	1.371	1.3717	1.3718	1.372
CPU sec	2.13	6.29	24.45	79.0	

$$\dot{x}(t) = -\sum_{i=1}^{K} \frac{x(t-i/K)}{K}$$

$K \downarrow\ n \rightarrow$	1	2	3	5	10
1	.366	.094	.158	.686	12.8
2	.112	.295	1.260	10.83	61.05
3	.177	1.311	6.86	96.85	5223
5	.895	13.05	124.7	2014	200950
10	13.09	59.5	5077	200231	NA

Table: CPU sec indexed by # of states (n) and # of delays (K)

Complexity Scaling Results:

- Viable when $nK < 50$

Significant reduction possible using Differential-Difference Formulation.

Illustration 2: The Timoschenko Beam

Consider a simple Timoschenko beam model:

$$\ddot{w} = \partial_s(w_s - \phi) \qquad = -\phi_s + w_{ss}$$
$$\ddot{\phi} = \phi_{ss} + (w_s - \phi) \qquad = -\phi + w_s + \phi_{ss}$$

with boundary conditions

$$\phi(0) = 0, \quad w(0) = 0, \quad \phi_s(L) = 0, \quad w_s(L) - \phi(L) = 0$$

Step 1: Eliminate w_{tt} and ϕ_{tt} - $u_1 = w_t$ and $u_3 = \phi_t$.
Step 2: Use BCs to pick the state - $u_2 = w_s - \phi$ and $u_4 = \phi_s$.

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}_t = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}}_{A_0} \underbrace{\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}}_{\mathbf{x}_2} + \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{A_1} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}_s$$

where $A_2 = []$ and $n_1 = n_3 = 0$ and $n_2 = 4$ - a purely "hyperbolic" form. We only need 4 BCs:

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

This gives a B has row rank $n_2 = 4$:

$$\underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}}_{B} \begin{bmatrix} u_1(0) \\ u_2(0) \\ u_3(0) \\ u_4(0) \\ u_1(L) \\ u_2(L) \\ u_3(L) \\ u_4(L) \end{bmatrix} = 0$$

Stable! However, not exponentially stable ($\dot{V} \not< 0$) in all the given states.

Illustration 3: The Tip-Damped Wave Equation

The simplest tip-damped wave equation is

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

Guided by the boundary conditions, we choose

$$\begin{aligned} u_1(t,s) &= u_s(t,s) \\ u_2(t,s) &= u_t(t,s) \end{aligned}$$

This yields

$$\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}_t = \underbrace{\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}}_{A_1} \underbrace{\begin{bmatrix} u_1 \\ u_2 \end{bmatrix}_s}_{x_2}$$

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

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

We prove exp. stability in the given states u_t, u_s for $k > 0$.

Converting an LMI to an LOI:

The LMI to LOI conversion process:

Step 1: Write the dynamics

$$\mathcal{T}\dot{\mathbf{x}}_f(t) = \mathcal{A}\mathbf{x}_f(t) + \mathcal{B}w(t), \qquad y(t) = \mathcal{C}\mathbf{x}_f(t) + Dw(t), \qquad \mathbf{x}_p(t) = \mathcal{T}\mathbf{x}_f$$

where $\mathcal{A}, \mathcal{B}, \mathcal{C}$ are in the $\{N_0, N_1, N_2\}$ algebra.

Step 2: Replace Matrices with Operators

$$\begin{bmatrix} A^T P + PA & PB & C^T \\ B^T P & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} \le 0 \;\rightarrow\; \begin{bmatrix} u \\ v \\ \mathbf{x}_f \end{bmatrix}^T \begin{bmatrix} -\gamma I & D^T & \mathcal{B}^*\mathcal{P}\mathcal{H} \\ D & -\gamma I & \mathcal{C} \\ \mathcal{H}^*\mathcal{P}\mathcal{B} & \mathcal{C}^* & \mathcal{A}^*\mathcal{P}\mathcal{H} + \mathcal{H}^*\mathcal{P}\mathcal{A} \end{bmatrix} \begin{bmatrix} u \\ v \\ \mathbf{x}_f \end{bmatrix} \prec 0$$

Why Does This Work?:

- The conversion between primal and fundamental state is a 4-PIE operator.
- We express the dynamics as a 4-PIE operator.
- We express the Lyapunov Function using a 4-PIE operator.
- 4-PIE operators are closed under composition, adjoint, and addition.
- We can parameterize 4-PIE operators using real numbers
- We can enforce positivity of 4-PIE operators using LMIs.

The KYP Lemma and 4-PIE

$$\mathcal{T}\dot{\mathbf{x}}_f(t) = \mathcal{A}\mathbf{x}_f + \mathcal{B}w(t)$$
$$z(t) = \mathcal{C}_1\mathbf{x}_f(t) + \mathcal{D}_1 w(t)$$

Theorem 2 (KYP and H_∞-Gain).

Suppose there exists operator $\mathcal{P} = \mathcal{P}\left\{\begin{smallmatrix} P, & Q_1 \\ Q_2, & \{R_i\}\end{smallmatrix}\right\} \geq 0$: *such that*

$$\left\langle \begin{bmatrix} w \\ v \\ \mathbf{x}_f \end{bmatrix}, \begin{bmatrix} -\gamma I & \mathcal{D}_1^* & \mathcal{B}^*\mathcal{P}\mathcal{T} \\ \mathcal{D}_1 & -\gamma I & \mathcal{C}_1 \\ \mathcal{T}^*\mathcal{P}\mathcal{B} & \mathcal{C}_1^* & \mathcal{A}^*\mathcal{P}\mathcal{T} + \mathcal{T}^*\mathcal{P}\mathcal{A} \end{bmatrix} \begin{bmatrix} w \\ v \\ \mathbf{x}_f \end{bmatrix} \right\rangle < 0$$

for any $\begin{bmatrix} w^T & v^T & \mathbf{x}_f^T \end{bmatrix} \in \mathbb{R}^{r+p+n} \times L_2^n[-\tau, 0]$, *where* $\mathcal{T}, \mathcal{A}, \mathcal{B}, \mathcal{C}_1, \mathcal{D}_1$ *are as defined previously. Then* $\|z\|_{L_2} \leq \gamma\|\omega\|_{L_2}$.

Proof **Choose Lyapunov function as**

$$V(\mathbf{x}_f) = \langle \mathcal{T}\mathbf{x}_f, \mathcal{P}\left\{\begin{smallmatrix} P, & Q_1 \\ Q_2, & \{R_i\}\end{smallmatrix}\right\}\mathcal{T}\mathbf{x}_f \rangle$$

Then $\dot{V}(\mathbf{x}_f) - \gamma w^T(t)w(t) - \gamma v(t)^T v(t) + \langle z(t), v(t)\rangle + \langle v(t), z(t)\rangle < 0$, where $v(t) = \frac{1}{\gamma}z(t)$, hence $\|z\|_{L_2} \leq \gamma\|\omega\|_{L_2}$.

The KYP Lemma and 4-PIE

Almost Complete Matlab Code:

```
pvar s th gam; opvar T A B C1 D1;
A=...;B=...;C1=...;D1=...;T=...;
X=[-tau,0];
prog = sosprogram([s;th],gam)
[prog, P] = sosposopvar(prog,[n n],X,s,th,[d1 d2]);
D=[-gam*eye(nw) D1'         B'*P*T;
   D1          -gam*eye(ny) C1;
   T'*P*B       C1'         T'*P*A+A'*P*T];
```

$$D = \begin{bmatrix} -\gamma I & \mathcal{D}_1^* & \mathcal{B}^*\mathcal{P}\mathcal{T} \\ \mathcal{D}_1 & -\gamma I & \mathcal{C}_1 \\ \mathcal{T}^*\mathcal{P}\mathcal{B} & \mathcal{C}_1^* & \mathcal{A}^*\mathcal{P}\mathcal{T} + \mathcal{T}^*\mathcal{P}\mathcal{A} \end{bmatrix}$$

```
[prog, N] = sosposopvar_noR0(prog,D.dim(:,2),X,s,th,[d1 d2]);
[prog, gN] = sosposopvar_noR0(prog,D.dim(:,2),X,s,th,[d1 d2]);
prog = sosopeq(prog,D+N+gN);
prog = sossetobj(prog, gamma); prog = sossolve(prog);
```

Illustration of H_∞ Gain Analysis

Example 1:

$$\dot{x}(t) = \begin{bmatrix} -2 & 0 \\ 0 & -.9 \end{bmatrix} x(t) + \begin{bmatrix} -1 & 0 \\ -1 & -1 \end{bmatrix} x(t-\tau) + \begin{bmatrix} -.5 \\ 1 \end{bmatrix} w(t), \quad y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t)$$

d	1	2	3	Padé	[Fridman 2001]	[Shaked 1998]
$\gamma_{\min}$	.2373	.2365	.2365	.2364	.32	2

Example 2: Stable for $\tau \in [.100173, 1.71785]$:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -2 & .1 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} x(t-\tau) + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} w(t)$$

$$y(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} x(t)$$

We plot bounds for the H_∞ norm as the delay varies within this interval. As expected, the H_∞ norm approaches infinity quickly as we approach the limits of the stable region.

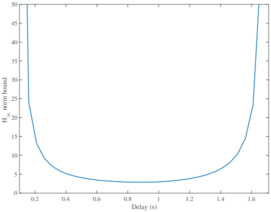

Figure: Calculated H_∞ norm bound vs. delay for Ex. 2

H_∞ Gain Analysis

Stable for $\lambda < 4.65$.

$$u_t(s,t) = A_0(s)u(s,t) + A_1(s)u_s(s,t) + A_2(s)u_{ss}(s,t) + w(t)$$
$$u(0,t) = 0 \qquad u_s(1,t) = 0$$
$$y(t) = \int_0^1 u(s,t)ds$$

$$A_0(s) = (-0.5s^3+1.3s^2-1.5s+0.7+\lambda), \quad A_1(s) = (3s^2-2s), \quad A_2(s) = (s^3-s^2+2)$$

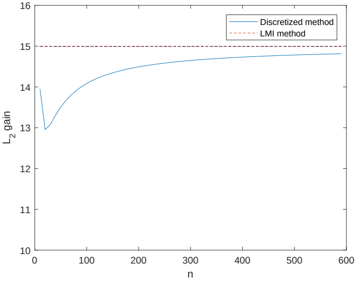

Figure: Compare with Discretization $(d = 1)$

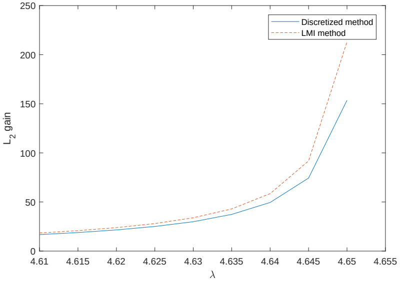

Figure: H_∞ Gain as a function of λ $(d = 1)$

PIE Formulation of the Controller Synthesis Problem

Write the ODE-PDE System as

$$\mathcal{T}\dot{\mathbf{x}}_f(t) = \mathcal{A}\mathbf{x}_f + \mathcal{B}_1 w(t) + \mathcal{B}_2 u(t) \qquad \mathcal{T}\mathbf{x}_f(t) = \mathbf{x}_p$$
$$z(t) = \mathcal{C}_1\mathbf{x}_f(t) + \mathcal{D}_1 w(t) + \mathcal{D}_2 u(t), \quad u(t) = \mathcal{K}\mathbf{x}(t)$$

Example: Time-Delay System

$$\mathcal{T} := \mathcal{P}\left\{\begin{smallmatrix} I, & 0 \\ I, & \{0,0,-I\}\end{smallmatrix}\right\} \quad \mathcal{A} := \mathcal{P}\left\{\begin{smallmatrix} A_0 + A_1, & -A_1 \\ 0, & \{I,0,0\}\end{smallmatrix}\right\} \quad \mathcal{C}_1 := \mathcal{P}\left\{\begin{smallmatrix} C_{10} + C_{11}, & -C_{11} \\ \emptyset, & \{\emptyset\}\end{smallmatrix}\right\}$$

$$\mathcal{B}_i := \mathcal{P}\left\{\begin{smallmatrix} B_i, & \emptyset \\ 0, & \{\emptyset\}\end{smallmatrix}\right\}, \quad \mathcal{D}_i := \mathcal{P}\left\{\begin{smallmatrix} D_i, & \emptyset \\ \emptyset, & \{\emptyset\}\end{smallmatrix}\right\}, \qquad \mathcal{K} := \mathcal{P}\left\{\begin{smallmatrix} K_1, & K_2 \\ \emptyset, & \{\emptyset\}\end{smallmatrix}\right\}$$

Theorem 3.

Suppose there exist operators $\mathcal{P} = \mathcal{P}\left\{\begin{smallmatrix} P, & Q \\ Q^T, & \{R_i\}\end{smallmatrix}\right\} > 0 : \mathbb{R}^n \times L_2^n \to \mathbb{R}^n \times L_2^n$ *and* $\mathcal{Z} = \mathcal{P}\left\{\begin{smallmatrix} Z_1, & Z_2 \\ \emptyset, & \{\emptyset\}\end{smallmatrix}\right\} : \mathbb{R}^n \times L_2^n \to \mathbb{R}^q$ *such that*

$$\begin{bmatrix} -\gamma I & D_1 & (\mathcal{C}\mathcal{P} + \mathcal{D}_2\mathcal{Z})\mathcal{T}^* \\ D_1^T & -\gamma I & (\mathcal{T}\mathcal{B}_1)^* \\ \mathcal{T}(\mathcal{C}\mathcal{P} + \mathcal{D}_2\mathcal{Z})^* & \mathcal{T}\mathcal{B}_1 & (\mathcal{A}\mathcal{P} + \mathcal{B}_2\mathcal{Z})\mathcal{T}^* + \mathcal{T}(\mathcal{A}\mathcal{P} + \mathcal{B}_2\mathcal{Z})^* \end{bmatrix} < 0$$

on $\mathbb{R}^{r+p+n} \times L_2^n[-\tau, 0]$, *where* $\mathcal{T}, \mathcal{A}, \mathcal{B}, \mathcal{C}_1, \mathcal{D}_1, \mathcal{C}_2, \mathcal{D}_2$ *are as defined above. Then if* $u(t) = \mathcal{K} = \mathcal{Z}\mathcal{P}^{-1}\mathbf{x}_f(t)$, *solutions satisfy* $\|z\|_{L_2} \le \gamma\|\omega\|_{L_2}$.

The Inverse of a PIE Operator is a PIE Operator!

Result from Keqin Gu

How to find (Note $R_1 = R_2$)

$$\mathcal{K} = \mathcal{P}\left\{\begin{smallmatrix} Z_1, & Z_2 \\ \emptyset, & \{\emptyset\}\end{smallmatrix}\right\}\mathcal{P}\left\{\begin{smallmatrix} & P, & Q \\ Q^T, & \{S, & R, & R\}\end{smallmatrix}\right\}^{-1}?$$

Assume Q and R are polynomial

Extract Polynomial Coefficients: $Q(s) = HZ(s)$ and $R(s,\theta) = Z(s)^T \Gamma Z(\theta)$.

Then $\mathcal{P}\left\{\begin{smallmatrix} & P, & Q \\ Q^T, & \{S, & R, & R\}\end{smallmatrix}\right\}^{-1} = \mathcal{P}\left\{\begin{smallmatrix} & \hat{P}, & \hat{Q} \\ \hat{Q}^T, & \{\hat{S}, & \hat{R}, & \hat{R}\}\end{smallmatrix}\right\}$ where

$$\hat{P} = \left(I - \hat{H} V H^T\right) P^{-1}, \qquad \hat{Q}(s) = \frac{1}{\tau}\hat{H} Z(s) S(s)^{-1}$$

$$\hat{S}(s) = \frac{1}{\tau^2} S(s)^{-1} \qquad \hat{R}(s,\theta) = \frac{1}{\tau} S(s)^{-1} Z(s)^T \hat{\Gamma} Z(\theta) S(\theta)^{-1},$$

where

$$\hat{H} = P^{-1} H \left(V H^T P^{-1} H - I - V\Gamma\right)^{-1}$$

$$\hat{\Gamma} = -(\hat{H}^T H + \Gamma)(I + V\Gamma)^{-1},$$

$$V = \int_{-\tau}^{0} Z(s) S(s)^{-1} Z(s)^T ds$$

Boring Numerical Controller Synthesis Examples

$$\dot{x}(t) = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} -1 & -1 \\ 0 & -.9 \end{bmatrix} x(t-\tau) + \begin{bmatrix} 1 \\ 1 \end{bmatrix} w(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ .1 \end{bmatrix} u(t)$$

d	1	2	3	Padé	Fridman 2003	Li 1997
$\gamma_{\min}(\tau = .999)$	.10001	.10001	.10001	.1000	.22844	1.8822
$\gamma_{\min}(\tau = 2)$	1.43	1.36	1.341	1.340	∞	∞
CPU sec	.478	.879	2.48	2.78	N/A	N/A

$$\dot{x}(t) = \begin{bmatrix} 2 & 1 \\ 0 & -1 \end{bmatrix} x(t) + \begin{bmatrix} -1 & 0 \\ -1 & 1 \end{bmatrix} x(t-\tau) + \begin{bmatrix} -.5 \\ 1 \end{bmatrix} w(t) + \begin{bmatrix} 3 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & -.5 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

d	1	2	3	Padé
$\gamma_{\min}(\tau = .3)$	.3953	.3953	.3953	.3953
CPU sec	.655	1.248	2.72	2.91

H_∞-Optimal Observer Synthesis in the PIE Framework

Nominal System:

$$\mathcal{T}\dot{\mathbf{x}}_f(t) = \mathcal{A}\mathbf{x}_f + \mathcal{B}w(t)$$
$$y(t) = \mathcal{C}_2\mathbf{x}_f(t) + \mathcal{D}_2 w(t), \qquad z(t) = \mathcal{C}_1\mathbf{x}_f(t) + \mathcal{D}_1 w(t)$$

Observer Structure using 4-PIE Operators:

$$\mathcal{T}\dot{\hat{\mathbf{x}}}_f(t) = \mathcal{A}\hat{\mathbf{x}}_f + \mathcal{L}(\hat{y} - y) = (\mathcal{A} + \mathcal{L}\mathcal{C}_2)\hat{\mathbf{x}}_f - \mathcal{L}\mathcal{C}_2\mathbf{x}_f - \mathcal{L}\mathcal{D}_2 w$$
$$\hat{y}(t) = \mathcal{C}_2\hat{\mathbf{x}}_f(t) \quad \hat{z}(t) = \mathcal{C}_1\hat{\mathbf{x}}_f(t)$$

where the observer gains are

$$\mathcal{L} := \mathcal{P}\left\{\begin{smallmatrix} L_1, & \emptyset \\ L_2, & \{\emptyset\} \end{smallmatrix}\right\}$$

Error Dynamics and the LMI for H_∞-optimal Observers

Define

$$\mathbf{e}_p = \hat{\mathbf{x}}_p - \mathbf{x}_p, \quad y_e(t) = \hat{y}(t) - y(t).$$

The closed-loop error system dynamics are

$$\mathcal{T}\dot{\mathbf{e}}_f(t) = (\mathcal{A} + \mathcal{L}\mathcal{C}_2)\mathbf{e}_f - (\mathcal{B} + \mathcal{L}\mathcal{D}_2)w(t) \quad \mathcal{T}\mathbf{e}_f(t) = \mathbf{e}_p(t)$$
$$z_e(t) = \mathcal{C}_1\mathbf{e}_f(t) - \mathcal{D}_1 w(t)$$

An LOI for H_∞-Optimal Observer Design

Error Dynamics:

$$\mathcal{T}\dot{\mathbf{e}}_f(t) = (\mathcal{A}+\mathcal{L}\mathcal{C}_2)\mathbf{e}_f - (\mathcal{B}+\mathcal{L}\mathcal{D}_2)w(t), \qquad z_e(t) = \mathcal{C}_1\mathbf{e}_f(t) - \mathcal{D}_1 w(t)$$

Theorem 4.

Suppose there exist operators $\mathcal{P} = \mathcal{P}\left\{\begin{smallmatrix} P, & Q \\ Q^T, & \{R_i\}\end{smallmatrix}\right\} \geq 0 : \mathbb{R}^n \times L_2^n \to \mathbb{R}^n \times L_2^n$ *and* $\mathcal{Z} = \mathcal{P}\left\{\begin{smallmatrix} Z_1, & \emptyset \\ Z_2, & \{\emptyset\}\end{smallmatrix}\right\} : \mathbb{R}^q \to \mathbb{R}^n \times L_2^n$ *such that*

$$\begin{bmatrix} -\gamma I & -\mathcal{D}_1^* & -(\mathcal{P}\mathcal{B}+\mathcal{Z}\mathcal{D}_2)^*\mathcal{T} \\ -\mathcal{D}_1 & -\gamma I & \mathcal{C}_1 \\ -\mathcal{T}^*(\mathcal{P}\mathcal{B}+\mathcal{Z}\mathcal{D}_2) & \mathcal{C}_1^* & (\mathcal{P}\mathcal{A}+\mathcal{Z}\mathcal{C}_2)^*\mathcal{T}+\mathcal{T}^*(\mathcal{P}\mathcal{A}+\mathcal{Z}\mathcal{C}_2) \end{bmatrix} < 0$$

on $\mathbb{R}^{r+p+n} \times L_2^n[-\tau, 0]$*, where* $\mathcal{T}, \mathcal{A}, \mathcal{B}, \mathcal{C}_1, \mathcal{D}_1, \mathcal{C}_2, \mathcal{D}_2$ *are as defined previously. Then if* $\mathcal{L} = \mathcal{P}^{-1}Z$*, solutions satisfy* $\|z_e\|_{L_2} \leq \gamma \|\omega\|_{L_2}$.

Lemma (Structure of $\mathcal{L}$): Suppose

$$\mathcal{P}\left\{\begin{smallmatrix} \hat{P}, & \hat{Q} \\ \hat{Q}^T, & \{\hat{R}_i\}\end{smallmatrix}\right\} = \mathcal{P}\left\{\begin{smallmatrix} P, & Q \\ Q^T, & \{R_i\}\end{smallmatrix}\right\}^{-1}$$

Then

$$\mathcal{L} := \mathcal{P}\left\{\begin{smallmatrix} \hat{P}, & \hat{Q} \\ \hat{Q}^T, & \{\hat{R}_i\}\end{smallmatrix}\right\}\mathcal{P}\left\{\begin{smallmatrix} Z_1, & \emptyset \\ Z_2, & \{\emptyset\}\end{smallmatrix}\right\} = \mathcal{P}\left\{\begin{smallmatrix} L_1, & \emptyset \\ L_2, & \{\emptyset\}\end{smallmatrix}\right\}$$

Proof **Choose Lyapunov function as**

$$V(\mathbf{e}_p) = \langle \mathbf{e}_p, \mathcal{P}\{{}_{Q^T,}^{P,}{}_{\{R_i\}}^{Q}\}\mathbf{e}_p \rangle$$

Define $v_e = \frac{1}{\gamma} z_e$. Then

$$\dot{V}(\mathbf{e}_p) - \gamma w^T w + \frac{1}{\gamma} z_e^T z_e =$$

$$\left\langle \begin{bmatrix} w \\ v_e \\ \mathbf{e}_f \end{bmatrix}, \begin{bmatrix} -\gamma I & -\mathcal{D}_1^* & -(\mathcal{P}\mathcal{B}+\mathcal{Z}\mathcal{D}_2)^*\mathcal{T} \\ -\mathcal{D}_1 & -\gamma I & \mathcal{C}_1 \\ -\mathcal{T}^*(\mathcal{P}\mathcal{B}+\mathcal{Z}\mathcal{D}_2) & \mathcal{C}_1^* & (\mathcal{P}\mathcal{A}+\mathcal{Z}\mathcal{C}_2)^*\mathcal{T} + \mathcal{T}^*(\mathcal{P}\mathcal{A}+\mathcal{Z}\mathcal{C}_2) \end{bmatrix} \begin{bmatrix} w \\ v_e \\ \mathbf{x}_f \end{bmatrix} \right\rangle < 0$$

Almost Complete Matlab Code:

```
pvar s th gam; opvar T A B C1 C2 D1 D2;
A=...;B=...;C1=...;C2=...;D1=...;D2=...;T=...;X=[-tau,0];
prog = sosprogram([s;th],gam)
[prog, P] = sos_posopvar(prog,[n n],X,s,th,[d1 d2]);
[prog, Z] = sos_opvar(prog,[n q;n 0],X,s,th,[d1 d2]);
D=[-gam*eye(nw)      D'                 -(P*B+Z*D2)'*T;
   D               -gam*eye(ny)       C1;
   -T'*(P*B+Z*D2)    C1'                T'*(P*A+Z*C2)+(P*A+Z*C2)*T];
[prog, N] = sosposop_RL2RL_noR0(prog,D.dim(:,2),X,s,th,[d1 d2]);
[prog, gN] = sosposop_RL2RL_noR0(prog,D.dim(:,2),X,s,th,[d1 d2]);
prog = sosopeq(prog,D+N+gN);prog = sossetobj(prog, gamma); prog =
sossolve(prog);
```

State Estimation Numerical Example

Distributed State Estimation:

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \lambda \begin{bmatrix}1 & 0.3\\ 0.1 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}x_{1ss}\\ x_{2ss}\end{bmatrix} + \begin{bmatrix}s-s^2\\ 0\end{bmatrix} w(t),$$

$$y = \int^{b} \begin{bmatrix}0 & 1\end{bmatrix}\begin{bmatrix}x_1(s)\\ x_2(s)\end{bmatrix} \mathsf{d}s, z(t) = \int^{b} \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}x_1(s)\\ x_2(s)\end{bmatrix} \mathsf{d}s.$$

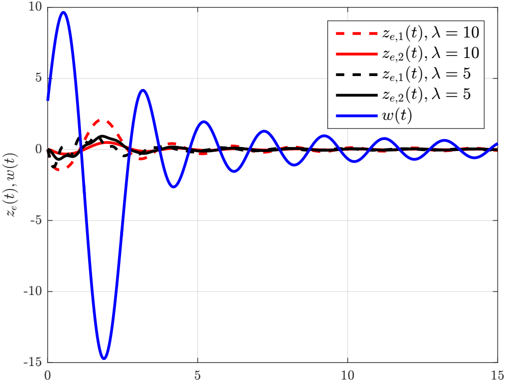

Figure: Time evolution of $z_e(t)$ and $w(t)$ for $\lambda \overset{t}{=} 5, 10$ where $w(t)$ is generated by damped sinusoidal functions.

Conclusion and Extensions (Thanks to ONR #N000014-17-1-2117)

$\mathcal{P}_{\{N_0,N_1,N_2\}}$ Framework extends LMI techniques to PDEs.

- $A^TP + PA < 0$ becomes

$$\underbrace{\mathcal{P}^*_{\{H_0,H_1,H_2\}}}_{A^T}\underbrace{\mathcal{P}_{\{N_0,N_1,N_2\}}}_{P}\mathcal{P}_{\{G_0,G_1,G_2\}}+\mathcal{P}^*_{\{G_0,G_1,G_2\}}\underbrace{\mathcal{P}_{\{N_0,N_1,N_2\}}}_{P}\underbrace{\mathcal{P}_{\{H_0,H_1,H_2\}}}_{A} \le 0$$

Conclusions:

PROs:

- Computationally Efficient
- A more rational treatment of boundary conditions.
- No Conservatism (Almost N+S)
- Easily Extended to New Problems
  - e.g. higher order derivatives
  - e.g. distributed dynamics

CONs:

- Requires $n_2 + 2n_3$ BCs to be clearly specified
- PDE Must be Stable in all States

Extensions:

- Input-Output Properties (ACC, 2019)
  - H_∞ Gain
  - passivity
- ODEs coupled with PDEs (CDPS, 2019)
- Optimal Estimator Synthesis
- Optimal Controller Synthesis

Solvable (in order of difficulty)

- Extension to 3D
- Duality (Stability of $\mathcal{A}^*$)
- Inversion of the $\mathcal{P}_{\{N_0,N_1,N_2\}}$ Operator
  - Want an Analytic Formula