Delay Compensation in Control Systems

Michael Malisoff

Background and Motivation

Background and Motivation

Delay compensating controllers are needed because of destabilizing effects of potentially long input delays.

Background and Motivation

Delay compensating controllers are needed because of destabilizing effects of potentially long input delays.

The linear matrix inequalities used to study time-invariant linear delayed systems are inadequate for nonlinear systems.

Background and Motivation

Delay compensating controllers are needed because of destabilizing effects of potentially long input delays.

The linear matrix inequalities used to study time-invariant linear delayed systems are inadequate for nonlinear systems.

Traditional Lyapunov functions are replaced by Razumikhin functions or Lyapunov-Krasovskii functionals.

Background and Motivation

Delay compensating controllers are needed because of destabilizing effects of potentially long input delays.

The linear matrix inequalities used to study time-invariant linear delayed systems are inadequate for nonlinear systems.

Traditional Lyapunov functions are replaced by Razumikhin functions or Lyapunov-Krasovskii functionals.

Lyapunov-Krasovskii functionals can often be built from Lyapunov functions for corresponding undelayed systems.

Background and Motivation

Delay compensating controllers are needed because of destabilizing effects of potentially long input delays.

The linear matrix inequalities used to study time-invariant linear delayed systems are inadequate for nonlinear systems.

Traditional Lyapunov functions are replaced by Razumikhin functions or Lyapunov-Krasovskii functionals.

Lyapunov-Krasovskii functionals can often be built from Lyapunov functions for corresponding undelayed systems.

M. Jankovic, M. Krstic, Z. Lin, SI. Niculescu, P. Pepe, A. Teel,...

Stability Definitions

Input-to-state stability (ISS, Sontag, ’89) generalizes uniform global asymptotic stability by quantifying effects of uncertainties.

Stability Definitions

Input-to-state stability (ISS, Sontag, ’89) generalizes uniform global asymptotic stability by quantifying effects of uncertainties.

$$x'(t) = \mathcal{G}(t, x(t), x(t-\tau)), \quad x(t) \in \mathcal{X} \tag{Σ}$$

$$|x(t)| \le \gamma_1 \left(e^{t_0 - t}\gamma_2(|x|_{[t_0-\tau, t_0]})\right) \tag{UGAS}$$

γ_i’s are 0 at 0, strictly increasing, and unbounded. $\gamma_i \in \mathcal{K}_\infty$.

Stability Definitions

Input-to-state stability (ISS, Sontag, '89) generalizes uniform global asymptotic stability by quantifying effects of uncertainties.

$$x'(t) = \mathcal{G}(t, x(t), x(t-\tau)), \quad x(t) \in \mathcal{X} \qquad (\Sigma)$$

$$|x(t)| \le \gamma_1 \left(e^{t_0 - t} \gamma_2(|x|_{[t_0-\tau, t_0]}) \right) \qquad \text{(UGAS)}$$

γ_i's are 0 at 0, strictly increasing, and unbounded. $\gamma_i \in \mathcal{K}_\infty$.

$$x'(t) = \mathcal{G}\big(t, x(t), x(t-\tau), \delta(t)\big), \quad x(t) \in \mathcal{X} \qquad (\Sigma_{\text{pert}})$$

$$|x(t)| \le \gamma_1 \left(e^{t_0 - t} \gamma_2(|x|_{[t_0-\tau, t_0]}) \right) + \gamma_3(|\delta|_{[t_0, t]}) \qquad \text{(ISS)}$$

Stability Definitions

Input-to-state stability (ISS, Sontag, '89) generalizes uniform global asymptotic stability by quantifying effects of uncertainties.

$$x'(t) = \mathcal{G}(t, x(t), x(t-\tau)), \quad x(t) \in \mathcal{X} \tag{Σ}$$

$$|x(t)| \le \gamma_1 \left(e^{t_0 - t}\gamma_2(|x|_{[t_0-\tau, t_0]})\right) \tag{UGAS}$$

γ_i's are 0 at 0, strictly increasing, and unbounded. $\gamma_i \in \mathcal{K}_\infty$.

$$x'(t) = \mathcal{G}\big(t, x(t), x(t-\tau), \delta(t)\big), \quad x(t) \in \mathcal{X} \tag{Σ_{pert}}$$

$$|x(t)| \le \gamma_1 \left(e^{t_0 - t}\gamma_2(|x|_{[t_0-\tau, t_0]})\right) + \gamma_3(|\delta|_{[t_0, t]}) \tag{ISS}$$

Find γ_i's by building Lyapunov-Krasovskii functionals (LKFs).

Definition in Control-Affine Case

$$\dot{x}(t) = f(t, x(t)) + g(t, x(t))[u_s(t, x(t - \tau)) + \delta(t)]. \qquad (\Sigma_{\mathrm{d}})$$

Definition in Control-Affine Case

$$\dot{x}(t) = f(t, x(t)) + g(t, x(t))[u_s(t, x(t - \tau)) + \delta(t)]. \qquad (\Sigma_\mathrm{d})$$

Assume: f and g are locally Lipschitz and grow linearly in x,
$u_s \in C^1$, $|(\partial u_s/\partial x)(t, x)|$ bounded.

Definition in Control-Affine Case

$$\dot{x}(t) = f(t, x(t)) + g(t, x(t))[u_s(t, x(t - \tau)) + \delta(t)]. \qquad (\Sigma_{\mathrm{d}})$$

Assume: f and g are locally Lipschitz and grow linearly in x, $u_s \in C^1$, $|(\partial u_s/\partial x)(t, x)|$ bounded. $x_t(s) = x(t + s)$, $-\tau \le s \le 0$.

Definition in Control-Affine Case

$$\dot{x}(t) = f(t, x(t)) + g(t, x(t))[u_s(t, x(t-\tau)) + \delta(t)]. \qquad (\Sigma_{\mathrm{d}})$$

Assume: f and g are locally Lipschitz and grow linearly in x, $u_s \in C^1$, $|(\partial u_s/\partial x)(t, x)|$ bounded. $x_t(s) = x(t+s)$, $-\tau \le s \le 0$.

Definition: A function $U : [0, \infty) \times \mathcal{C}_n([-\tau, 0]) \to [0, \infty)$ is an ISS-LKF for (Σ_d) provided there are $\alpha_i \in \mathcal{K}_\infty$ such that for all solutions $x(t)$ of (Σ_d), $U(t, x_t)$ is absolutely continuous in t and

(i) $\alpha_1(|\phi(0)|) \le U(t, \phi) \le \alpha_2(|\phi|_{[-\tau,0]})$ and

(ii) $\frac{d}{dt}\big(U(t, x_t)\big) \le -\alpha_3(U(t, x_t)) + \alpha_4(|\delta|_{[t_o,t]})$

hold for all $\phi \in \mathcal{C}_n([-\tau, 0])$ and almost all $t \ge t_o$ and all $t_0 \ge 0$.

Ways We Built Delay-Tolerant Feedback Controls

Ways We Built Delay-Tolerant Feedback Controls

First Approach: Emulation

Ways We Built Delay-Tolerant Feedback Controls

First Approach: Emulation

1. Solve the stabilization problem with the delays set to zero, by building a strict LF for the undelayed closed-loop system.

Ways We Built Delay-Tolerant Feedback Controls

First Approach: Emulation

1. Solve the stabilization problem with the delays set to zero, by building a strict LF for the undelayed closed-loop system.
2. Transform the LF into a Lyapunov-Krasovkii functional (LKF) for the delayed system by adding double integrals.

Ways We Built Delay-Tolerant Feedback Controls

First Approach: Emulation

1. Solve the stabilization problem with the delays set to zero, by building a strict LF for the undelayed closed-loop system.
2. Transform the LF into a Lyapunov-Krasovkii functional (LKF) for the delayed system by adding double integrals.
3. Use the LKF to compute upper bounds on the delays that the feedback can tolerate, and use strictness to prove ISS.

Ways We Built Delay-Tolerant Feedback Controls

First Approach: Emulation

1. Solve the stabilization problem with the delays set to zero, by building a strict LF for the undelayed closed-loop system.
2. Transform the LF into a Lyapunov-Krasovkii functional (LKF) for the delayed system by adding double integrals.
3. Use the LKF to compute upper bounds on the delays that the feedback can tolerate, and use strictness to prove ISS.

Mazenc, F., M. Malisoff, and Z. Lin, "Further results on input-to-state stability for nonlinear systems with delayed feedbacks," *Automatica*, 44(9):2415-2421, 2008.

Ways We Built Delay-Tolerant Feedback Controls

First Approach: Emulation

1. Solve the stabilization problem with the delays set to zero, by building a strict LF for the undelayed closed-loop system.
2. Transform the LF into a Lyapunov-Krasovkii functional (LKF) for the delayed system by adding double integrals.
3. Use the LKF to compute upper bounds on the delays that the feedback can tolerate, and use strictness to prove ISS.

Malisoff, M., and F. Zhang, "Robustness of adaptive control under time delays for three-dimensional curve tracking," *SIAM Journal on Control and Optimization*, 53(4):2203-2236, 2015.

First Approach: Emulation

Assumption L: There are $\sigma \in \mathcal{K}_\infty$ such that $\sigma(r) \leq r$ for all $r \geq 0$; constants $K_1 \geq 1$ and $K_i \geq 0$ for $i = 2, 3, 4$; and a C^1 uniformly proper and positive definite $V : [0, \infty) \times \mathbb{R}^n \to [0, \infty)$ such that for all $x \in \mathbb{R}^n$, $q \in \mathbb{R}^n$, $l \geq 0$, and $t \geq 0$, we have

- H1 $V_t(t, x) + V_x(t, x)[f(t, x) + g(t, x)u_s(t, x)] \leq -\sigma^2(|x|)$;
- H2 $|V_x(t, x)g(t, x)| \leq K_1\sigma(|x|)$, $\left|\frac{\partial u_s}{\partial x}(t, x)f(l, x)\right|^2 \leq K_2\sigma(|x|)^2$;
- H3 $\left|\frac{\partial u_s}{\partial x}(t, x)g(l, x)\right|^2 \leq K_3(\sigma(|x|) + 1)$; and
- H4 $\left[\left|\frac{\partial u_s}{\partial x}(t, x)g(l, x)\right| |u_s(l, q)|\right]^2 \leq K_4[\sigma^2(|x|) + \sigma^2(|q|)]$.

First Approach: Emulation

Assumption L: There are $\sigma \in \mathcal{K}_\infty$ such that $\sigma(r) \leq r$ for all $r \geq 0$; constants $K_1 \geq 1$ and $K_i \geq 0$ for $i = 2, 3, 4$; and a C^1 uniformly proper and positive definite $V : [0, \infty) \times \mathbb{R}^n \to [0, \infty)$ such that for all $x \in \mathbb{R}^n$, $q \in \mathbb{R}^n$, $l \geq 0$, and $t \geq 0$, we have

H1 $V_t(t, x) + V_x(t, x)[f(t, x) + g(t, x)u_s(t, x)] \leq -\sigma^2(|x|)$;

H2 $|V_x(t, x)g(t, x)| \leq K_1\sigma(|x|)$, $\left|\frac{\partial u_s}{\partial x}(t, x)f(l, x)\right|^2 \leq K_2\sigma(|x|)^2$;

H3 $\left|\frac{\partial u_s}{\partial x}(t, x)g(l, x)\right|^2 \leq K_3(\sigma(|x|) + 1)$; and

H4 $\left[\left|\frac{\partial u_s}{\partial x}(t, x)g(l, x)\right| |u_s(l, q)|\right]^2 \leq K_4[\sigma^2(|x|) + \sigma^2(|q|)]$.

Exponentially stable $\dot{x}(t) = (A(t) + B(t)K(t))x(t)$ with $\sigma(s) = s$

First Approach: Emulation

Thm 1: (M-Mazenc-Lin, '08) If Assumption L holds, then

$$\dot{x}(t) = f(t, x(t)) + g(t, x(t))[u_s(t, x(t - \tau)) + \delta(t)] \qquad (\Sigma_{\mathrm{d}})$$

with any constant feedback delay $\tau \in (0, \bar{\tau}]$ where

$$\bar{\tau} = \frac{1}{4K_1\sqrt{3K_2+3K_4+1}}$$

admits the ISS-LKF

$$U(t, x_t) = V(t, x(t)) + \frac{1}{8\bar{\tau}}\int_{t-2\bar{\tau}}^{t}\left(\int_r^t \sigma^2(|x(p)|)\mathrm{d}p\right)\mathrm{d}r$$

and therefore is ISS. □

First Approach: Emulation

Thm 1: (M-Mazenc-Lin, '08) If Assumption L holds, then

$$\dot{x}(t) = f(t, x(t)) + g(t, x(t))[u_s(t, x(t-\tau)) + \delta(t)] \qquad (\Sigma_{\mathrm{d}})$$

with any constant feedback delay $\tau \in (0, \bar{\tau}]$ where

$$\bar{\tau} = \frac{1}{4K_1\sqrt{3K_2+3K_4+1}}$$

admits the ISS-LKF

$$U(t, x_t) = V(t, x(t)) + \frac{1}{8\bar{\tau}} \int_{t-2\bar{\tau}}^{t} \left(\int_r^t \sigma^2(|x(p)|)\mathrm{d}p \right) \mathrm{d}r$$

and therefore is ISS. □

Delay need not be known.

First Approach: Emulation

Thm 1: (M-Mazenc-Lin, '08) If Assumption L holds, then

$$\dot{x}(t) = f(t, x(t)) + g(t, x(t))[u_s(t, x(t-\tau)) + \delta(t)] \qquad (\Sigma_{\mathrm{d}})$$

with any constant feedback delay $\tau \in (0, \bar{\tau}]$ where

$$\bar{\tau} = \frac{1}{4K_1\sqrt{3K_2+3K_4+1}}$$

admits the ISS-LKF

$$U(t, x_t) = V(t, x(t)) + \tfrac{1}{8\bar{\tau}} \int_{t-2\bar{\tau}}^{t} \left(\int_r^t \sigma^2(|x(p)|)\mathrm{d}p\right) \mathrm{d}r$$

and therefore is ISS. □

Delay need not be known. Can drop delay bound in many cases using reduction or prediction or scaling of controls.

Ways We Built Delay-Tolerant Feedback Controls

Second Approach: Reduction Model

Ways We Built Delay-Tolerant Feedback Controls

Second Approach: Reduction Model

1. Find controls depending on inputs along a continuum of times by solving integral equations, for any constant delay.

Ways We Built Delay-Tolerant Feedback Controls

Second Approach: Reduction Model

1. Find controls depending on inputs along a continuum of times by solving integral equations, for any constant delay.
2. They globally stabilize linear time-varying systems, which can arise from linearizing along a reference trajectory.

Ways We Built Delay-Tolerant Feedback Controls

Second Approach: Reduction Model

1. Find controls depending on inputs along a continuum of times by solving integral equations, for any constant delay.
2. They globally stabilize linear time-varying systems, which can arise from linearizing along a reference trajectory.
3. We can also prove local stabilization for time-varying nonlinear systems with basin of attraction computations.

Ways We Built Delay-Tolerant Feedback Controls

Second Approach: Reduction Model

1. Find controls depending on inputs along a continuum of times by solving integral equations, for any constant delay.
2. They globally stabilize linear time-varying systems, which can arise from linearizing along a reference trajectory.
3. We can also prove local stabilization for time-varying nonlinear systems with basin of attraction computations.

Mazenc, F., M. Malisoff, and S.-I. Niculescu, "Reduction model approach for linear time-varying systems with delays," *IEEE Transactions on Automatic Control*, 59(8):2068-2082, 2014.

Ways We Built Delay-Tolerant Feedback Controls

Second Approach: Reduction Model

1. Find controls depending on inputs along a continuum of times by solving integral equations, for any constant delay.
2. They globally stabilize linear time-varying systems, which can arise from linearizing along a reference trajectory.
3. We can also prove local stabilization for time-varying nonlinear systems with basin of attraction computations.

Mazenc, F., and M. Malisoff, "Local stabilization of nonlinear systems through the reduction model approach," *IEEE Transactions on Automatic Control*, 59(11):3033-3039, 2014.

Second Approach: Reduction Model

$$\dot{x}(t) = M(t)x(t) + N(t)u(t-\tau) + \delta(t). \tag{1}$$

Second Approach: Reduction Model

$$\dot{x}(t) = M(t)x(t) + N(t)u(t-\tau) + \delta(t). \qquad (1)$$

Thm 2: (MMN'14) If there is a bounded continuous K such that

$$\dot{z}(t) = \big[M(t) + \lambda(t, t+\tau)N(t+\tau)K(t)\big]z(t) \qquad (2)$$

is UGAS, where λ is the fundamental matrix for M, then there are functions $\gamma_i \in \mathcal{K}_\infty$ such that all trajectories of (1) with

$$u(t) = K(t)\left[x(t) + \int_{t-\tau}^{t} \lambda(t, r+\tau)N(r+\tau)u(r)\mathrm{d}r\right] \qquad (3)$$

satisfy

$$|x(t)| + |u|_{[t-\tau,t]} \leq \gamma_1\Big(\gamma_2\big(|x(t_0)| + |u|_{[t_0-\tau,t_0]}\big)e^{t_0-t}\Big) + \gamma_3\big(|\delta|_{[t_0,t]}\big)$$

for all initial times $t_0 \geq 0$ and all $t \geq t_0$.

Ways We Built Delay-Tolerant Feedback Controls

Third Approach: Sequential Predictors

Ways We Built Delay-Tolerant Feedback Controls

Third Approach: Sequential Predictors

1. They allow arbitrarily long time-varying delays and provide controls that are free of distributed terms.

Ways We Built Delay-Tolerant Feedback Controls

Third Approach: Sequential Predictors

1. They allow arbitrarily long time-varying delays and provide controls that are free of distributed terms.
2. They use dynamic ODE controllers that include copies of the original system running at different time scales.

Ways We Built Delay-Tolerant Feedback Controls

Third Approach: Sequential Predictors

1. They allow arbitrarily long time-varying delays and provide controls that are free of distributed terms.
2. They use dynamic ODE controllers that include copies of the original system running at different time scales.
3. They apply under input and measurement delays, sampling, outputs, and uncertainties in the plant and measurements.

Ways We Built Delay-Tolerant Feedback Controls

Third Approach: Sequential Predictors

1. They allow arbitrarily long time-varying delays and provide controls that are free of distributed terms.
2. They use dynamic ODE controllers that include copies of the original system running at different time scales.
3. They apply under input and measurement delays, sampling, outputs, and uncertainties in the plant and measurements.

Mazenc, F., and M. Malisoff, "Stabilization and robustness analysis for time-varying systems with time-varying delays using a sequential predictors approach," *Automatica*, 82:118-127, 2017.

Ways We Built Delay-Tolerant Feedback Controls

Third Approach: Sequential Predictors

1. They allow arbitrarily long time-varying delays and provide controls that are free of distributed terms.
2. They use dynamic ODE controllers that include copies of the original system running at different time scales.
3. They apply under input and measurement delays, sampling, outputs, and uncertainties in the plant and measurements.

Mazenc, F., and M. Malisoff, "Stabilization of nonlinear time-varying systems through a new prediction based approach," *IEEE Transactions on Automatic Control*, 62(6):2908-2915, 2017.

Third Approach: Sequential Predictors

$$\dot{x}(t) = A(t)x(t) + B(t)u(t - h(t)), \quad x(t) \in \mathbb{R}^n. \qquad \text{(LTV)}$$

Third Approach: Sequential Predictors

$$\dot{x}(t) = A(t)x(t) + B(t)u(t - h(t)), \quad x(t) \in \mathbb{R}^n. \qquad \text{(LTV)}$$

Assumption 1: The functions A and B are bounded and continuous, and there is a known bounded continuous function $K : [0, \infty) \to \mathbb{R}^{\ell \times n}$ such that $\dot{x}(t) = [A(t) + B(t)K(t)]x(t)$ is uniformly globally exponentially stable to 0.

Third Approach: Sequential Predictors

$$\dot{x}(t) = A(t)x(t) + B(t)u(t - h(t)), \quad x(t) \in \mathbb{R}^n. \qquad \text{(LTV)}$$

Assumption 1: The functions A and B are bounded and continuous, and there is a known bounded continuous function $K : [0, \infty) \to \mathbb{R}^{\ell \times n}$ such that $\dot{x}(t) = [A(t) + B(t)K(t)]x(t)$ is uniformly globally exponentially stable to 0.

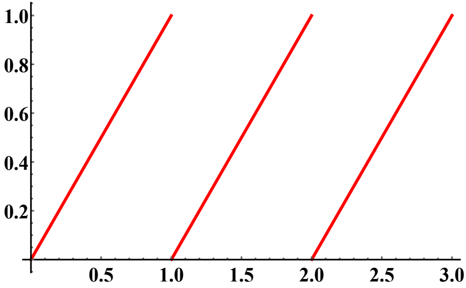
Assumption 2: The function $h : \mathbb{R} \to [0, \infty)$ is C^1 and bounded from above by a constant $c_h > 0$. Also, its derivative $\dot{h}$ is bounded from below, and $\dot{h}$ is bounded from above by a constant $l_h \in (0, 1)$, and $\dot{h}$ has a global Lipschitz constant $n_h > 0$.

Third Approach: Sequential Predictors

Sawtooth wave delay represents sampling in control.

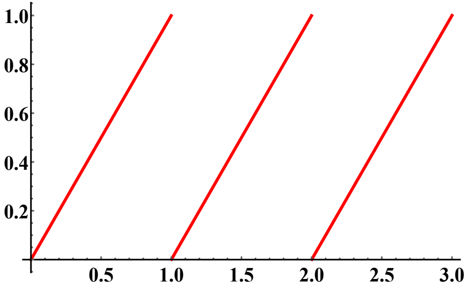
Third Approach: Sequential Predictors

Sawtooth wave delay represents sampling in control.



Third Approach: Sequential Predictors

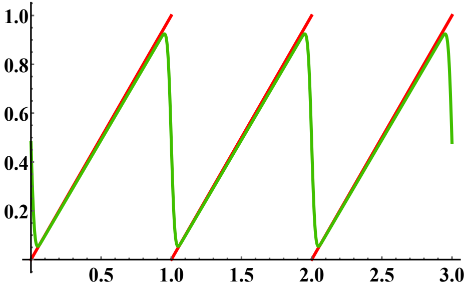
Sawtooth wave delay represents sampling in control.



Gaussian smoothing and interpolation. 1000 interpolation points and standard deviation 0.2 of smoothing on $[0, 1]$. Scale by 0.98.

Third Approach: Sequential Predictors

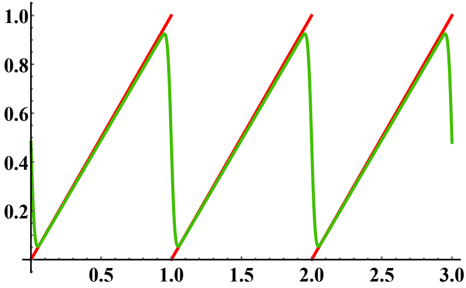
Sawtooth wave delay represents sampling in control.



Gaussian smoothing and interpolation. 1000 interpolation points and standard deviation 0.2 of smoothing on $[0,1]$. Scale by 0.98.

Third Approach: Sequential Predictors

Sawtooth wave delay represents sampling in control.



Gaussian smoothing and interpolation. 1000 interpolation points and standard deviation 0.2 of smoothing on $[0, 1]$. Scale by 0.98.

Assumption 2 holds with $c_h = 0.924$, $l_h = 0.98$, and $n_h = 592.72$.

Third Approach: Sequential Predictors

We use an $p n$-dimensional dynamic extension to build our delay compensating control for any number of predictors

$$p > \max\left\{2, 4\left(\frac{b_1}{\sqrt{2}} + b_2\right)\frac{c_h}{1-l_h}\right\}, \tag{LB}$$

where

$$b_1 = \left[1 + \left(1 + \frac{u_c}{p}\right)^p |A|_\infty\right]\left(1 + \frac{u_c}{p}\right)^p |A|_\infty,$$

$$b_2 = \left[1 + \left(1 + \frac{u_c}{p}\right)^p |A|_\infty\right]^2\left(1 + \frac{u_c}{p}\right), \text{ and } u_c = \frac{c_h n_h}{(1-l_h)^2} + \frac{l_h}{1-l_h}.$$

Third Approach: Sequential Predictors

We use an pn-dimensional dynamic extension to build our delay compensating control for any number of predictors

$$p > \max\left\{2, 4\left(\frac{b_1}{\sqrt{2}} + b_2\right)\frac{c_h}{1-l_h}\right\}, \qquad \text{(LB)}$$

where

$$b_1 = \left[1 + \left(1 + \frac{u_c}{p}\right)^p |A|_\infty\right]\left(1 + \frac{u_c}{p}\right)^p |A|_\infty,$$

$$b_2 = \left[1 + \left(1 + \frac{u_c}{p}\right)^p |A|_\infty\right]^2\left(1 + \frac{u_c}{p}\right), \quad \text{and} \quad u_c = \frac{c_h n_h}{(1-l_h)^2} + \frac{l_h}{1-l_h}.$$

p sequential predictors for $\dot{x}(t) = A(t)x(t) + B(t)u(t - h(t))$

Third Approach: Sequential Predictors

We use an pn-dimensional dynamic extension to build our delay compensating control for any number of predictors

$$p > \max\left\{2, 4\left(\frac{b_1}{\sqrt{2}} + b_2\right)\frac{c_h}{1-l_h}\right\}, \tag{LB}$$

where

$$b_1 = \left[1 + \left(1 + \frac{u_c}{p}\right)^p |A|_\infty\right]\left(1 + \frac{u_c}{p}\right)^p |A|_\infty,$$

$$b_2 = \left[1 + \left(1 + \frac{u_c}{p}\right)^p |A|_\infty\right]^2\left(1 + \frac{u_c}{p}\right), \quad \text{and} \quad u_c = \frac{c_h n_h}{(1-l_h)^2} + \frac{l_h}{1-l_h}.$$

p sequential predictors for $\dot{x}(t) = A(t)x(t) + B(t)u(t - h(t))$

$\Omega_i(t) = t - (i/p)h(t)$ and $\theta_i(t) = \Omega^{-1}_{p-i+1}(\Omega_{p-i}(t))$ for $i \in \{0, ..., p\}$

Third Approach: Sequential Predictors

We use an pn-dimensional dynamic extension to build our delay compensating control for any number of predictors

$$p > \max\left\{2, 4\left(\frac{b_1}{\sqrt{2}} + b_2\right)\frac{c_h}{1-l_h}\right\}, \qquad \text{(LB)}$$

where

$$b_1 = \left[1 + \left(1 + \frac{u_c}{p}\right)^p |A|_\infty\right]\left(1 + \frac{u_c}{p}\right)^p |A|_\infty,$$

$$b_2 = \left[1 + \left(1 + \frac{u_c}{p}\right)^p |A|_\infty\right]^2 \left(1 + \frac{u_c}{p}\right), \text{ and } u_c = \frac{c_h n_h}{(1-l_h)^2} + \frac{l_h}{1-l_h}.$$

p sequential predictors for $\dot{x}(t) = A(t)x(t) + B(t)u(t - h(t))$

$\Omega_i(t) = t - (i/p)h(t)$ and $\theta_i(t) = \Omega_{p-i+1}^{-1}(\Omega_{p-i}(t))$ for $i \in \{0, ..., p\}$

$R_1(t) = \dot{\theta}_1(t)$, $R_i(t) = \dot{\theta}_i(t)R_{i-1}(\theta_i(t))$ for $i > 1$.

Third Approach: Sequential Predictors

Thm 3: (M-Mazenc, '17) Let Assumptions 1-2 hold and p satisfy (LB). Then if we use the control $u(t) = K(\Omega_p^{-1}(t))z_p(t)$ in (LTV), where z_p is the last n components of the system

$$\begin{aligned} \dot{z}_1(t) &= R_1(t)A(\theta_1(t))z_1(t) + R_1(t)B(\theta_1(t))u(\Omega_{p-1}(t)) \\ &\quad + L_1(t)[z_1(\theta_1^{-1}(t)) - x(t)] \\ \dot{z}_i(t) &= R_i(t)A(G_i(t))z_i(t) + R_i(t)B(G_i(t))u(\Omega_{p-i}(t)) \\ &\quad + L_i(t)[z_i(\theta_i^{-1}(t)) - z_{i-1}(t)],\ i \in \{2, \dots, p\} \end{aligned} \tag{4}$$

where $L_i(t) = -I_n - R_i(t)A(G_i(t))$ and $G_i = \Omega_p^{-1} \circ \Omega_{p-i}$, then the dynamics for $(x, \mathcal{E})$ are globally exponentially stable to 0, where $\mathcal{E}(t) = (z_1(t) - x(\theta_1(t)), z_2(t) - z_1(\theta_2(t)), \dots, z_p(t) - z_{p-1}(\theta_p(t)))$.

Conclusions

Conclusions

- Delays are prevalent in engineering systems.

Conclusions

- ▷ Delays are prevalent in engineering systems.
- ▷ Controls for undelayed systems might not be delay-tolerant.

Conclusions

- ▷ Delays are prevalent in engineering systems.
- ▷ Controls for undelayed systems might not be delay-tolerant.
- ▷ Reduction model methods compensate any positive delay.

Conclusions

- ▷ Delays are prevalent in engineering systems.
- ▷ Controls for undelayed systems might not be delay-tolerant.
- ▷ Reduction model methods compensate any positive delay.
- ▷ Distributed terms can produce implementation challenges.

Conclusions

- ▷ Delays are prevalent in engineering systems.
- ▷ Controls for undelayed systems might not be delay-tolerant.
- ▷ Reduction model methods compensate any positive delay.
- ▷ Distributed terms can produce implementation challenges.
- ▷ They can sometimes be overcome by sequential predictors.

Conclusions

- ▷ Delays are prevalent in engineering systems.
- ▷ Controls for undelayed systems might not be delay-tolerant.
- ▷ Reduction model methods compensate any positive delay.
- ▷ Distributed terms can produce implementation challenges.
- ▷ They can sometimes be overcome by sequential predictors.
- ▷ Sequential predictors allow outputs, sampling and uncertainty.

Conclusions

- ▷ Delays are prevalent in engineering systems.
- ▷ Controls for undelayed systems might not be delay-tolerant.
- ▷ Reduction model methods compensate any positive delay.
- ▷ Distributed terms can produce implementation challenges.
- ▷ They can sometimes be overcome by sequential predictors.
- ▷ Sequential predictors allow outputs, sampling and uncertainty.
- ▷ We are developing analogs for ODE-PDE cascades.

Conclusions

▷ Delays are prevalent in engineering systems.

▷ Controls for undelayed systems might not be delay-tolerant.

▷ Reduction model methods compensate any positive delay.

▷ Distributed terms can produce implementation challenges.

▷ They can sometimes be overcome by sequential predictors.

▷ Sequential predictors allow outputs, sampling and uncertainty.

▷ We are developing analogs for ODE-PDE cascades.

Thank you for your attention!