Reduction of discrete-time 2-channel delayed systems

Mattia Mattioni*, Salvatore Monaco* & Dorothée Normand-Cyrot**

*Dipt. Ingegneria Informatica, Automatica e Gestionale A. Ruberti
Università di Roma La Sapienza - Rome - It

**Laboratoire des Signaux et Systèmes
CNRS - Gif-sur-Yvette - Fr

57^{th} IEEE-CDC 2018

Problem settlement

$$x(k+1) = F(x(k), u_1(k - N_1), u_2(k - N_2))$$

- $F(\cdot) : \mathbb{R}^n \times \mathbb{R}^2 \to \mathbb{R}^n$ analytic in its arguments
- $x = 0$ is the equilibrium to be stabilized, $F(0,0,0) = F_0(0) = 0$
- $u_i \in \mathbb{R}$, $i = 1, 2$ affected by different time delays $N_1, N_2 \in \mathbb{N}$
- the delays N_1, N_2 are known

$\Rightarrow$ Difficulties to directly apply the single input methodologies

Motivations

- Nonlinear discrete-time design with delays
- Interconnected and network systems
- Sampled-data context
  - Sampling is instrumental for stabilizing input delayed continuous-time dynamics
  - From an infinite dimensional problem to a finite dimensional one
  - A straight connivence between sampled-data dynamics and dynamics with input delays

---

I. Karafyllis and M. Krstic, Nonlinear stabilization under sampled and delayed measurements, and with inputs subject to delay and zero-order hold, IEEE TAC, 2012.

F. Mazenc, M. Malisoff, and T. N. Dinh, Robustness of nonlinear systems with respect to delay and sampling of the controls, Automatica, 2013.

P. Pepe, Stabilization in the Sample-and-Hold Sense of Nonlinear Retarded Systems, SIAM J. Control Optim., 2014.

E. Fridman, A.Seuret, J.P. Richard, Robust sampled-data stabilization of linear systems: An input delay approach, Automatica, 2004.

Outline of the talk

- Review on the single input case (or uniform delay case)
- Focus on two methodologies
  - The Immersion and Invariance (I&I) approach
  - The reduction method
- The main result
- Conclusions
- An example

Exploit the extended cascade representation

Extended delay free form $v^i(k) = u(k - N - 1 + i); i = 1, ..., N$:

$$x(k+1) = F(x(k), v^1(k))$$
$$v^1(k+1) = v^2(k)$$
$$\dots$$
$$v^N(k+1) = u(k)$$

- Backstepping design (pure feedback form) $\Rightarrow$ a top down strategy
- Feedforwarding design $\Rightarrow$ a bottom up strategy
- Immersion & Invariance techniques $\Rightarrow$ a natural framework for systems with delays

---

V. Tanasa, S.Monaco and D. Normand-Cyrot, Backstepping control under multi-rate sampling. IEEE TAC, 2016

M. Mattioni, S.Monaco and D. Normand-Cyrot. Lyapunov stabilization of discrete-time feedforward dynamics, Proc. 56th IEEE-CDC, 2017.

Immersion & Invariance- I&I

⇒ A very natural framework to stabilize input delayed dynamics in discrete time

A. Astolfi and R. Ortega, Immersion and invariance: a new tool for stabilization and adaptive control of nonlinear systems, IEEE TAC, 2003.

Y. Yalcin and A. Astolfi. Discrete time immersion and invariance adaptive control for systems in strict feedback form. Proc 50th IEEE-CDC /EUCA-ECC, 2011.

I &I stability in discrete time - the main ideas

$$x(k+1) = F(x(k), u(k)) \qquad \text{over} \quad \mathbb{R}^n \qquad \text{with} \qquad x_* = F(x_*, 0)$$

H$_1$) Existence of a target dynamics on $\mathbb{R}^p; p < n$ with ξ_* GAS

$$\xi(k+1) = \alpha(\xi(k))$$

H$_2$) Immersion mapping $\pi(\cdot) : \mathbb{R}^p \to \mathbb{R}^n$ and invariance under $c(\cdot) : \mathbb{R}^p \to \mathbb{R}$

$$\pi(\alpha(\xi)) = F\Big(\pi(\xi), c(\xi)\Big) \qquad x_* = \pi(\xi_*)$$

H$_3$) Implicit manifold description

$$\{x \in \mathbb{R}^n | \phi(x) = 0\} \equiv \{x \in \mathbb{R}^n | x = \pi(\xi)\} \qquad \phi(\cdot) : \mathbb{R}^n \to \mathbb{R}^{n-p}$$

H$_4$) I&I stability $\equiv$ manifold attractivity

Setting : $z = \phi(x)$ Design : $u = \psi(x, z)$ $\psi(\cdot) : \mathbb{R}^{n \times (n-p)} \to \mathbb{R}$

$$\lim_{k \to \infty} z(k) = 0 \quad \text{and} \quad \psi(\pi(\xi), 0) = c(\xi)$$

$\Rightarrow$ x_* is GAS for the closed loop x-dynamics (provided full trajectories boundedness)

$$x(k+1) = F(x(k), \psi(x(k), \phi(x(k))))$$

Ex: I&I stabilization of DT dynamics $N = 1$

$$x(k+1) = F(x(k), v(k))$$
$$v(k+1) = u(k)$$

A1: *there exists a stabilizing delay free feedback* $u = \gamma(x)$

Th. A1 $\Rightarrow$ I&I stabilizability of the input delayed dynamics with :

- target dynamics $x(k+1) = F(x(k), \gamma(x(k)))$
- immersion mapping $\pi(x) = (x^T, \gamma(x))^T$
- invariant manifold $v - \gamma(x) = 0$ and $z = v - \gamma(x)$; *a prediction error*.

$\Rightarrow$ Any feedback achieving $\lim_{k\to\infty} z(k) = 0$ with boundedness of the trajectories over $\mathbb{R}^{n+2}$

$$x(k+1) = F(x(k), v(k))$$
$$v(k+1) = \psi(x(k), v(k), z(k))$$
$$z(k+1) = \psi(x(k), v(k), z(k)) - \gamma(F(x(k), v(k)))$$

achieves GAS of the equilibrium $(0, 0)$.

Some more comments

$$u = \psi(x, v, z) = \gamma(F(x, \gamma(x) + z)) + K_0 z; \qquad |K_0| < 1$$

Easy manipulations in $O(z^2)$ give the computable solution :

$$u_{ap} = \underbrace{\gamma(F(x, \gamma(x))}_{\text{predictor-based control}} + \underbrace{K_0(x) z}_{\text{feedback on the off-manifold variable}}$$

with suitably chosen $K_0(x)$ so that u_{ap} remains bounded and :

$$\left| K_0(x) - \frac{\partial \gamma}{\partial x}\Big|_{x=F(x,\gamma(x))} \frac{\partial F}{\partial v}\Big|_{v=\gamma(x)} \right| < 1.$$

⇒ I& I applies to multi-input equal delays

⇒ I& I improves the predictor based feedback

S. Monaco, D. Normand-Cyrot and M. Mattioni (2017) Sampled-data stabilization of nonlinear dynamics with input delays through immersion and invariance, *IEEE TAC*

About reduction based methods

Given

$$x(k+1) = F(x(k), u(k-N))$$

with $x \in \mathbb{R}^n$, $u \in \mathbb{R}^p$, possessing an equilibrium at the origin and affected by a discrete delay $N \geq 0$

How to "reduce the delay" through suitable state transformations ?

- the state Predictor
- the proposed Reduction variable

Predictor based feedback

$$x(k+1) = F(x(k), u(k-N))$$

- Set $z(k) = x(k+N)$ so reducing the delay on the z-dynamics

$$z(k+1) = F(z(k), u(k))$$

Any feedback $u(k) = \gamma(z(k)) = \gamma(x(k+N))$ stabilizing the delay free dynamics (assumption **A1**) stabilizes the delayed one N steps ahead.

- Computation of the predicted state $z(k) = x(k+N)$ is possible through finite composition of functions:

$$x(k+N) = F(\cdot, u(k-1)) \circ \cdots \circ F(x(k), u(k-N)).$$

Reduction based feedback

$$x(k+1) = F(x(k), u(k-N))$$

Assuming $F_0(\cdot)$ invertible $F_0^{-1}(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$, set

$$\eta(k) = F_0^{-N}(x(k+N)) = F_0^{-N} \circ F(\cdot, u(k-1)) \circ \cdots \circ F(x(k), u(k-N))$$

$$\Downarrow$$

$$\eta(k+1) = F_{r(N)}(\eta(k), u(k)) = F_0^{-N}(\cdot) \circ F(\cdot, u(k)) \circ F_0^N(\eta(k))$$

the η-dynamics is delay free with unchanged drift $F_{r(N)}(\eta, 0) = F_0(\eta)$

Any feedback $u(k) = \alpha_N(\eta(k))$ stabilizing the η-dynamics stabilizes the delayed one.

$$\begin{aligned} \eta(k+1) =& F_{r(N)}(\eta(k), \alpha_N(\eta(k))) \\ x(k+1) =& F(x(k), \alpha_N(F_0^{-N}(x(k)) = \eta(k-N)) \end{aligned}$$

The closed loop η-dynamics and x-dynamics are $\equiv$ through $x = F_0^N(\eta)$

Extension to multi-input equal delays is possible

Reduction versus prediction

- Computational advantages

$$\eta(0) = x(0) \quad \text{when} \quad u(-N) = ... = u(-1) = 0$$

$$\begin{aligned}\eta(k) &= F_0^{-N} \circ F(\cdot, u(k-1)) \circ \cdots \circ F(x(k), u(k-N)) \\ &= x(k) + O(u(k-1), ..., u(k-N))\end{aligned}$$

- Approximations can be performed
- More possibilities for the design
- The design is directly performed on the reduced η-dynamics which depends on the delay
- The knowledge of a stabilizing feedback for the delay free dynamics A1 is relaxed

---

M. Mattioni, S.Monaco and D. Normand-Cyrot (2018) Nonlinear discrete-time systems with delayed control: a reduction, *Systems and Control Letters*

M. Mattioni, S.Monaco and D. Normand-Cyrot (2017) Sampled-data reduction of nonlinear input-delayed dynamics; *L-CSS, IEEE Control Systems Letters*

The difficulties of the non uniform delays

Let Σ_{NuD} (non uniform delay)

$$x(k+1) = F(x(k), u_1(k-N_1), u_2(k-N_2))$$

$N_2 - N_1 = N > 0$ $(convention)$ $F_0(\cdot) := F(\cdot, 0, 0)$ invertible over $\mathbb{R}^n$

- Predictor, I&I or reduction methods do require $N_2 = N_1$!
- How to prevent from a large state extension $N_1 + N_2$?
- How to prevent from an extra usefulness prediction over $N_2 > N1$?

The proposed strategy

Combine the two methodologies into 3 steps:

1. Introduce a dynamical extension over u_2 so to compensate the mismatch among the two input delays
2. Apply the Reduction method to design the feedback over u_1 so reducing the lowest delay
3. Apply the Immersion and Invariance method to design the complete feedback over $(u_1, u_2) \Rightarrow$ so improving the remaining mismatch prediction

The transformed dynamics

1. Let $\xi := (\xi_1, \dots, \xi_N)^\top$ with $\xi_i(k) = u_2(k - N + i - 1)$ for $i = 1, \dots, N = N_2 - N_1$

$$\begin{aligned} x(k+1) &= F(x(k), u_1(k - N_1), \xi_1(k - N_1)) \\ \xi_1(k+1) &= \xi_2(k) \\ &\cdots \\ \xi_N(k+1) &= u_2(k) \end{aligned}$$

$\Rightarrow$ A cascade structure over R^{n+N} with state and input delays of the same length N_1

The transformed dynamics

1. Let $\xi := (\xi_1, \dots, \xi_N)^\top$ with $\xi_i(k) = u_2(k - N + i - 1)$ for $i = 1, \dots, N = N_2 - N_1$

$$\begin{aligned} x(k+1) =& F(x(k), u_1(k-N_1), \xi_1(k-N_1)) \\ \xi_1(k+1) =& \xi_2(k) \\ &\dots \\ \xi_N(k+1) =& u_2(k) \end{aligned}$$

$\Rightarrow$ A cascade structure over R^{n+N} with state and input delays of the same length N_1

2. Let $\eta(k) := F_0^{-N_1}(x(k+N_1))$ the reduction variable over the N_1-delayed x-dynamics

$$\begin{aligned} \eta(k+1) =& F_{r(N_1)}(\eta(k), u_1(k), \xi_1(k)) = F_0^{-N_1}(\cdot) \circ F(\cdot, u_1(k), \xi_1(k)) \circ F_0^{N_1}(\eta(k)) \\ \xi_1(k+1) =& \xi_2(k) \\ &\dots \\ \xi_N(k+1) =& u_2(k). \end{aligned}$$

$\Rightarrow$ An *extended reduced dynamics* over R^{n+N}, delay free with respect to (u_1, u_2)
$\Rightarrow$ A cascade with connection variable ξ_1.

The complete Reduction + I&I design

Given the extended reduced (η, ξ) -dynamics

$$\begin{aligned} \eta(k+1) =& F_{r(N_1)}(\eta(k), u_1(k), \xi_1(k)) \\ \xi_1(k+1) =& \xi_2(k) \\ & \dots \\ \xi_N(k+1) =& u_2(k). \end{aligned}$$

1. Compute a feedback ($u_1 = \gamma_1(\eta)$, $\xi_1 = \gamma_2(\eta)$) which makes GAS the equilibrium of the η-dynamics
2. The (η, ξ)-dynamics is I&I stabilizable with target dynamics

$$\eta(k+1) = F_{r(N_1)}(\eta(k), \gamma_1(\eta(k)), \gamma_2(\eta(k)))$$

3. Compute the I&I stabilizing feedback:

$$\begin{aligned} u_1(k) =& \gamma_1(\eta(k)) \\ u_2(k) =& \gamma_2(\eta(k+N)) + Lz(k) \end{aligned}$$

with $z \in \mathbb{R}^N; z_i = \xi_i - \gamma_2(\eta(k+i-1))$ and suitable gain matrix L so achieving GAS of the equilibrium of the given x-delayed dynamics.

step 1 $\equiv$ stabilization of the x-dynamics without delay mismatch $N_1 = N_2$

Conclusions and future work

Conclusions

- the proposed solution combines the two methodologies Reduction plus I&I
- the reduction avoids extra state extension which reduces to the mismatch between the delays - idem in the multi-input case
- I&I further improves the predictor based feedback
- extensions to cascade connected dynamics with both input and transmission delays are possible

Ongoing research

- $\Rightarrow$ quantify the robustness improvements !
- the delays have to be known $\Rightarrow$ relax the assumption !

Thanks for your attention

Companion papers :

M. M,S.M, DNC (2018) Reduction-based stabilization of time-delay continuous-time nonlinear dynamics; Proc. 57th IEEE CDC, Miami, USA

M. M,S.M, DNC (2017) Sampled-data reduction of nonlinear input-delayed dynamics;

L-CSS, IEEE Control Systems Letters

The example - PVTOL

$$\begin{aligned}
\dot{x}_1(t) =& x_2(t)\\
\dot{x}_2(t) =& u_1(t-\tau_1)\\
\dot{x}_3(t) =& x_5(t)\\
\dot{x}_4(t) =& x_6(t)\\
\dot{x}_5(t) =& u_2(t-\tau_2)\\
\dot{x}_6(t) =& -x_3(t)(1+u_1(t-\tau_1))
\end{aligned}$$

Setting

- $x = \mathrm{col}(x_1, \ldots, x_6)$
- $u_i(t) = Cst = u_i(k)$ for $t \in [k\delta, (k+1)\delta[$ and $x(k) := x(t = k\delta)$
- $\tau_1 = N_1\delta = \delta$ and $\tau_2 = N_2\delta = 2\delta$

$\Rightarrow$ a finitely computable discrete-time equivalent

The discrete-time equivalent dynamics

$$x(k+1) = A^{\delta}x(k) + B_0^{\delta}(u_1(k-1), u_2(k-2)) + B_1^{\delta}(u_1(k-1))x(k)$$

$$A^{\delta} = \begin{pmatrix} 1 & \delta & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & \delta & 0 \\ 0 & 0 & -\frac{\delta^2}{2} & 1 & -\frac{\delta^3}{6} & \delta \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -\delta & 0 & -\frac{\delta^2}{2} & 1 \end{pmatrix};$$

$$B_0^{\delta}(u_1, u_2) = \begin{pmatrix} \frac{\delta^2 u_1}{2} \\ \delta u_1 \\ \frac{\delta^2 u_2}{2} \\ \frac{-\delta^4 (1+u_1) u_2}{24} \\ \delta u_2 \\ \frac{-\delta^3 (1+u_1) u_2}{6} \end{pmatrix}$$

$$B_1^{\delta}(u_1) = \begin{pmatrix} & & & \mathbf{0}_{3\times 6} & & \\ 0 & 0 & -\frac{\delta^2}{2}u_1 & 0 & -\frac{\delta^3}{6}u_1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\delta u_1 & 0 & -\frac{\delta^2}{2}u_1 & 0 \end{pmatrix}.$$

The example - PVTOL

A multirate stabilizing design of order 2 on u_1 and 4 on u_2

$$\begin{aligned} u_1^i(k) =& \gamma_1^i(\eta(k)) \\ u_2^j(k) =& \gamma_2^j(\eta(k+1)) + \ell_j z^j(k), \quad |\ell_j| < 1. \end{aligned}$$

for $i = 1, 2$ and $j = 1, \ldots, 4$, with $z^j(k) = \xi^j(k) - \gamma_2^j(\eta(k))$.

- Deadbeat maneuver from the origin to final configuration $x_d^\top = (10, 0, 0, 10, 0, 0)$; $\ell_j = 0$ for $j = 1, 2, 3, 4$.
- Red lines : evolution of the target and the controls under uniform delay $N_1 = N_2 = 1$
- Blue lines : evolution with $N_1 = 1$ and $N_2 = 2$.

Simulations - $\delta = 10$ seconds and $x_d = (10, 0, 0, 10, 0, 0)$