Taming Delays in Cyber-Physical Systems

Naijun Zhan

SKLCS, Institute of Software, Chinese Academy Sciences, Beijing, China

The Algebra and Logic Seminar

By Algebra and Logic Department, IMI-BAS

Sept. 9, 2022

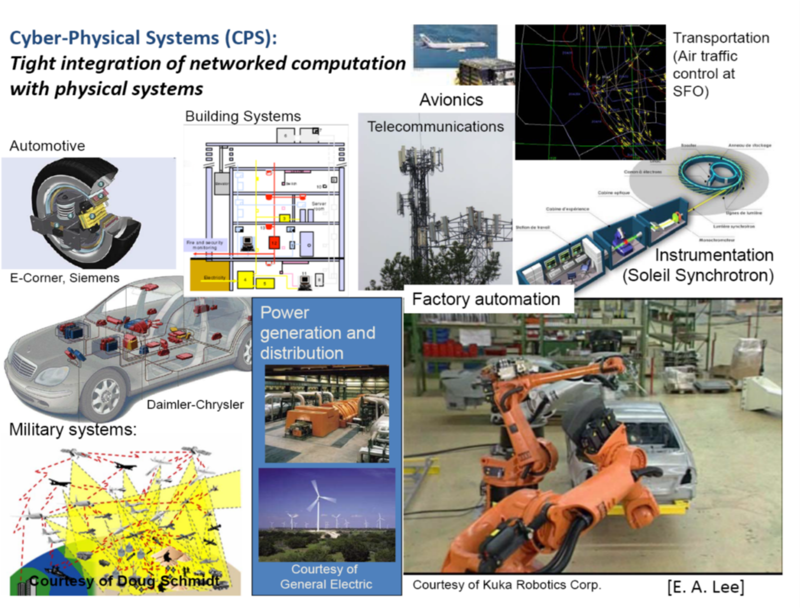
Cyber-Physical Systems

"*The term* **Cyber-Physical Systems (CPS)** *refers a new generation of systems with* integrated computational and physical *capabilities that can* interact with humans *through many new modalities.*

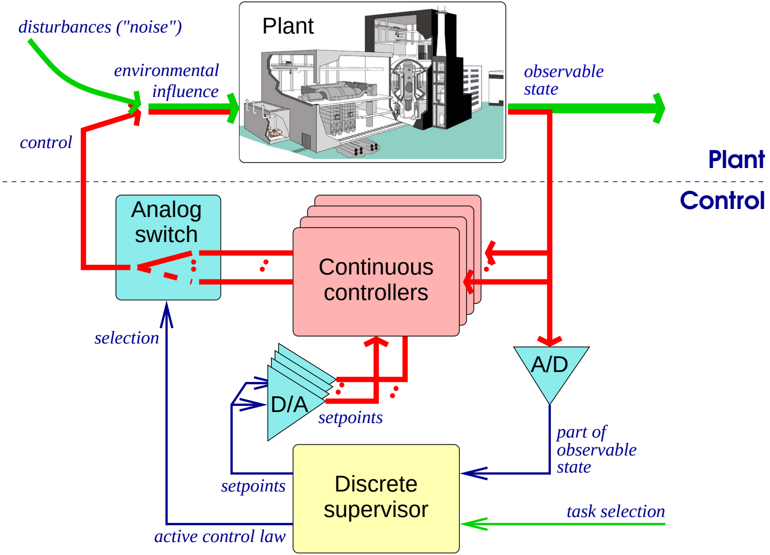
The ability to interact with, and expand the capabilities of, the physical world through **computation**, **communicaiton**, *and* **control** *is a key enabler for future technology developments.* "

— Helen Gill and Kisan Baheti NSF. IEEE Impact of Control Technology. Available at www.ieeecss.org

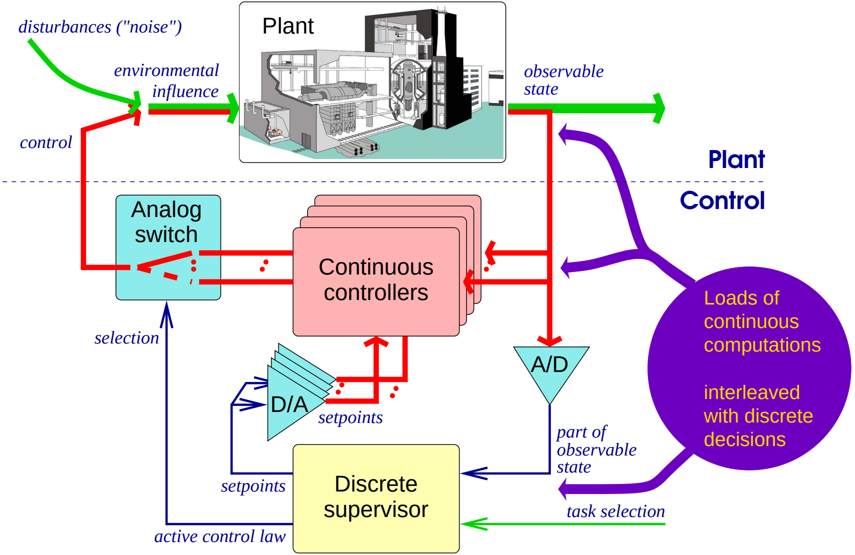
Cyber-Physical Systems



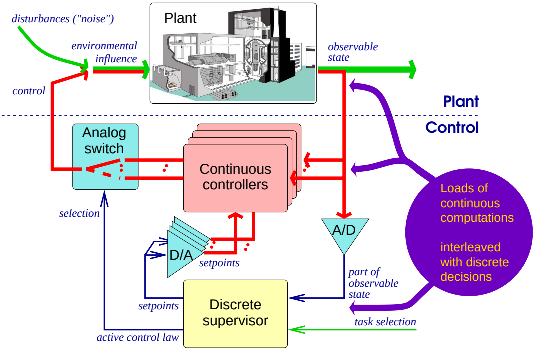
Hybrid Systems – A Common CPS Model



Hybrid Systems – A Common CPS Model



Hybrid Systems – A Common CPS Model



Crucial question:

- How do the controller and the plant interact?

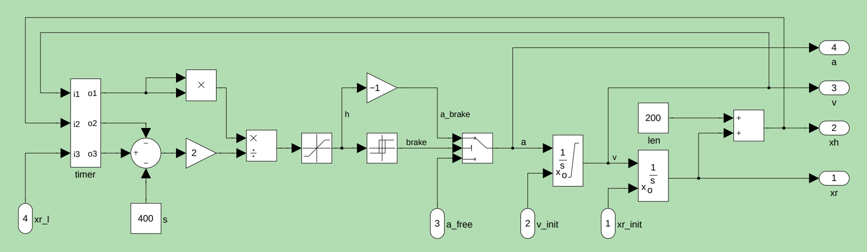
Traditional answer:

- Coupling assumed to be (or at least modeled as) delay-free.

⇒ Mode dynamics is covered by the conjunction of the individual ODEs.

⇒ Switching btw. modes is an immediate reaction to environmental conditions.

Instantaneous Coupling



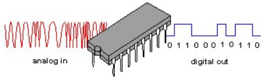
Following the tradition, above (rather typical) Simulink model assumes

- delay-free coupling between all components,
- instantaneous feed-through within all functional blocks.

Central questions:

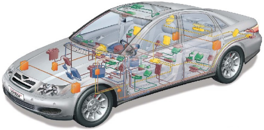
1. Is this realistic?
2. If not, does it have observable effect on control performance?
3. May that effect be detrimental or even harmful?

Q1: Is Instantaneous Coupling Realistic?



Digital control needs A/D and D/A conversion, which induces latency in signal forwarding.

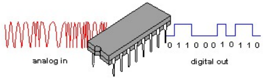
Digital signal processing, especially in complex sensors like CV, needs processing time, adding signal delays.



Networked control introduces communication latency into the feedback control loop.

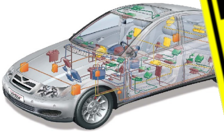
Harvesting, fusing, and forwarding data through sensor networks enlarge the latter by orders of magnitude.

Q1: Is Instantaneous Coupling Realistic? No.



Digital control needs A/D and D/A conversion, which induces latency in signal forwarding.

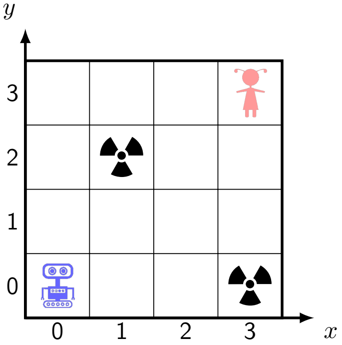
Digital sig specially in complex
essing time, adding



communication
ol loop.

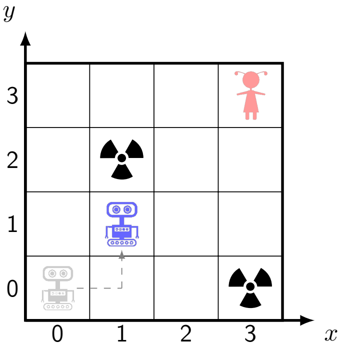
Harvesting, fusing, and forwarding data through sensor networks enlarge the latter by orders of magnitude.

Q2: Do Delays Have Observable Effect?



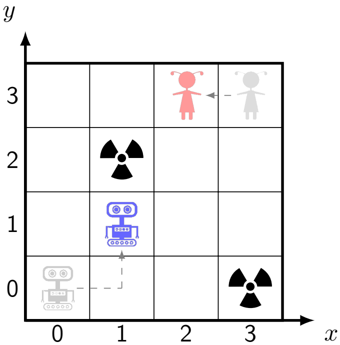
A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

Q2: Do Delays Have Observable Effect?



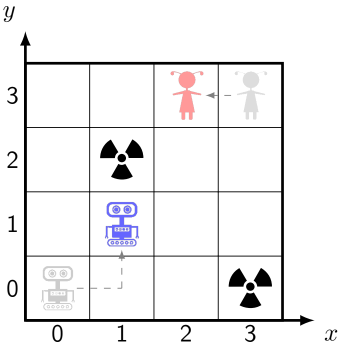
A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

Q2: Do Delays Have Observable Effect?



A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

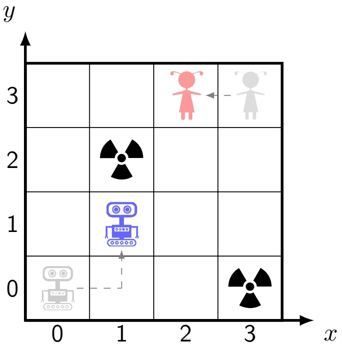
Q2: Do Delays Have Observable Effect?



A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

No delay:

Q2: Do Delays Have Observable Effect?

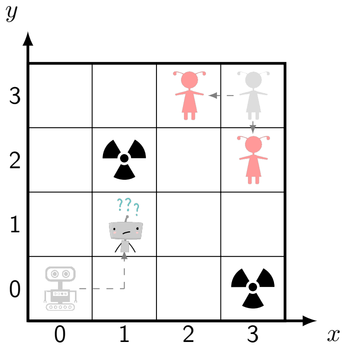


A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

No delay:

Robot always wins by circling around the obstacle at (1,2).

Q2: Do Delays Have Observable Effect?



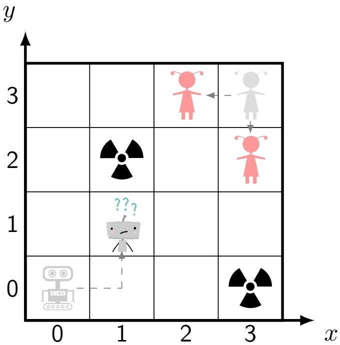
A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

No delay:

Robot always wins by circling around the obstacle at (1,2).

1 step delay:

Q2: Do Delays Have Observable Effect?



A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

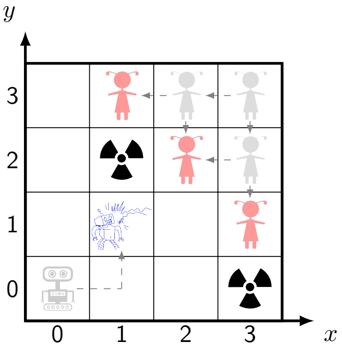
No delay:

Robot always wins by circling around the obstacle at (1,2).

1 step delay:

Robot wins by 1-step pre-decision.

Q2: Do Delays Have Observable Effect?



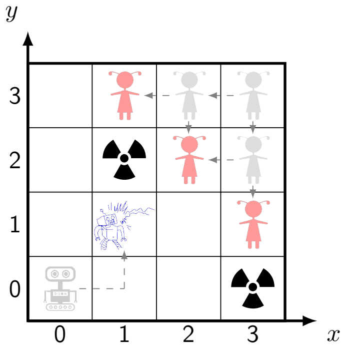
A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

No delay:
Robot always wins by circling around the obstacle at (1,2).

1 step delay:
Robot wins by 1-step pre-decision.

2 steps delay:

Q2: Do Delays Have Observable Effect?



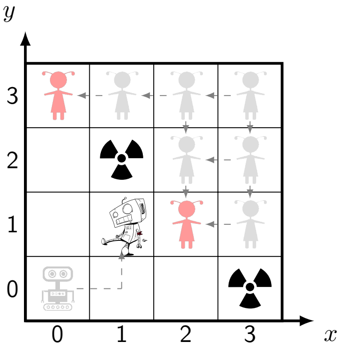
A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

No delay:
Robot always wins by circling around the obstacle at (1,2).

1 step delay:
Robot wins by 1-step pre-decision.

2 steps delay:
Robot still wins, yet extra memory is needed.

Q2: Do Delays Have Observable Effect?



A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

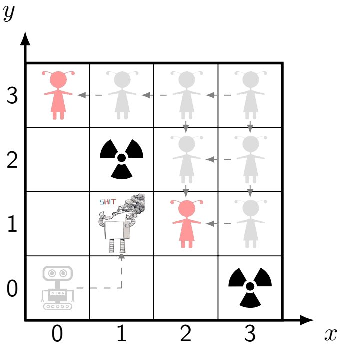
No delay:
Robot always wins by circling around the obstacle at (1,2).

1 step delay:
Robot wins by 1-step pre-decision.

2 steps delay:
Robot still wins, yet extra memory is needed.

3 steps delay:

Q2: Do Delays Have Observable Effect?



A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

No delay:

Robot always wins by circling around the obstacle at (1,2).

1 step delay:

Robot wins by 1-step pre-decision.

2 steps delay:

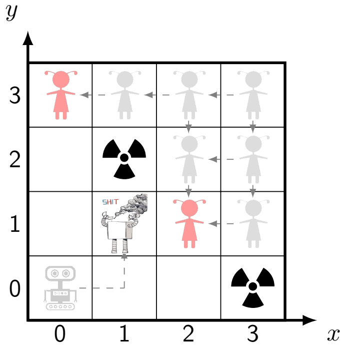
Robot still wins, yet extra memory is needed.

3 steps delay:

Robot is unwinnable (uncontrollable) anymore.

Q2: Do Delays Have Observable Effect?

– **Yes, they have.**



A robot escape game in a 4×4 room, with
$\Sigma_r = \{\mathtt{RU}, \mathtt{UR}, \mathtt{LU}, \mathtt{UL}, \mathtt{RD}, \mathtt{DR}, \mathtt{LD}, \mathtt{DL}, \epsilon\}$,
$\Sigma_k = \{\mathtt{R}, \mathtt{L}, \mathtt{U}, \mathtt{D}\}$.

No delay:

Robot always wins by circling around the obstacle at (1,2).

1 step delay:

Robot wins by 1-step pre-decision.

2 steps delay:

Robot still wins, yet extra memory is needed.

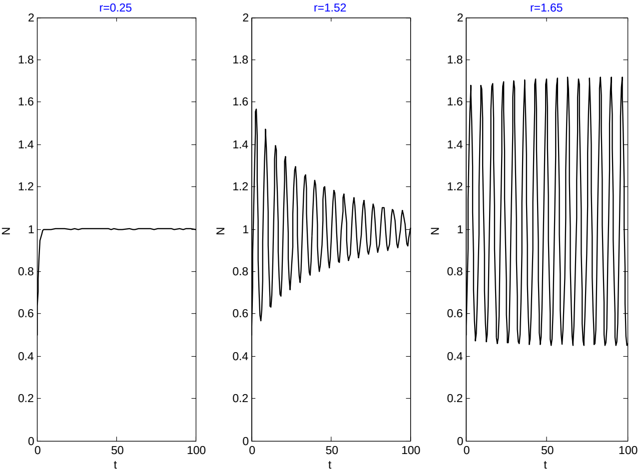
3 steps delay:

Robot is unwinnable (uncontrollable) anymore.

Q3: May the Effects be Harmful?

- Delayed logistic equation [G. Hutchinson, 1948]:

$$\frac{\mathrm{d}}{\mathrm{d}t}N(t) = N(t)[1 - N(t-r)]$$

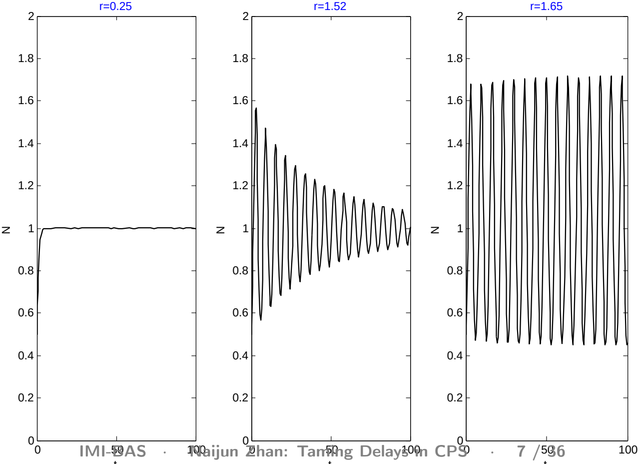


Q3: May the Effects be Harmful?

– Yes, delays may well annihilate control performance.

- Delayed logistic equation [G. Hutchinson, 1948]:

$$\frac{\mathrm{d}}{\mathrm{d}t}N(t) = N(t)[1 - N(t-r)]$$



DDE — The History

Historical motivation (predating digital control):

> *"Despite [...] very satisfactory state of affairs as far as [ordinary] differential equations are concerned, we are nevertheless forced to turn to the study of more complex equations. Detailed studies of the real world impel us, albeit reluctantly, to take account of the fact that the rate of change of physical systems depends not only on their present state, but also on their past history."*

[Richard Bellman and Kenneth L. Cooke, 1963]

DDE — The History

Historical motivation (predating digital control):

> *"Despite [...] very satisfactory state of affairs as far as [ordinary] differential equations are concerned, we are nevertheless forced to turn to the study of more complex equations. Detailed studies of the real world impel us, albeit reluctantly, to take account of the fact that the rate of change of physical systems depends not only on their present state, but also on their past history."*

[Richard Bellman and Kenneth L. Cooke, 1963]

Mathematical form:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = f(\mathbf{x}(t), \mathbf{x}(t-\delta_1), \ldots, \mathbf{x}(t-\delta_n)), \text{ with } \delta_n > \ldots > \delta_1 > 0,$$

DDE — The History

Historical motivation (predating digital control):

> *"Despite [...] very satisfactory state of affairs as far as [ordinary] differential equations are concerned, we are nevertheless forced to turn to the study of more complex equations. Detailed studies of the real world impel us, albeit reluctantly, to take account of the fact that the rate of change of physical systems depends not only on their present state, but also on their past history."*

[Richard Bellman and Kenneth L. Cooke, 1963]

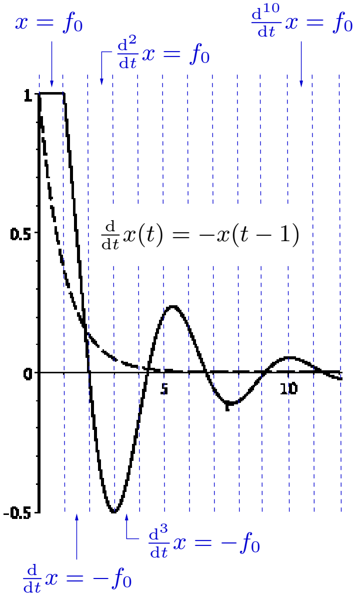
Mathematical form:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = f(\mathbf{x}(t), \mathbf{x}(t-\delta_1), \ldots, \mathbf{x}(t-\delta_n)), \text{ with } \delta_n > \ldots > \delta_1 > 0,$$

Simplest instance (which we will mostly concentrate on in the remainder):

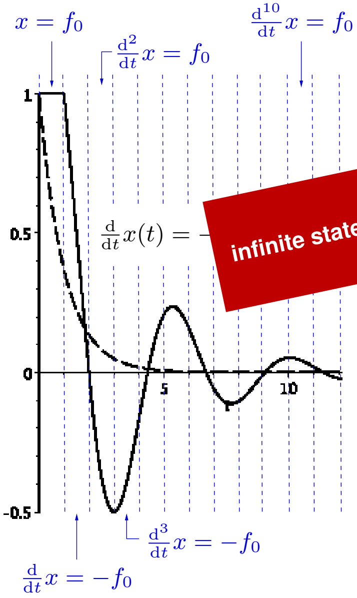
$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = f(\mathbf{x}(t-\delta))$$

DDE — Why They are Hard(er)



DDE constitute a model of system dynamics beyond "state snapshots":

- They feature "functional state" instead of state in the $\mathbb{R}^n$.
- Thus providing rather infallible, infinite-dimensional memory of the past.



Try only if infinite state no longer is scary enough to you!

- ...model of system dy-... state snapshots":
- They feature "functional state" instead of state in the $\mathbb{R}^n$.
- Thus providing rather infallible, infinite-dimensional memory of the past.

N.B.: More complex transformations may be applied to the initial segment f_0 according to the DDE's right-hand side. f_0 will nevertheless hardly ever vanish from the state space.

Conclusioin

- Delays in feedback control loops are ubiquitous, give difficulties.
- They may well invalidate the safety/stability/...certificates obtained by verifying delay-free abstractions of the feedback control system.

Automatic verification/synthesis methods addressing feedback delays in hybrid systems should therefore abound!

Conclusioin

- Delays in feedback control loops are ubiquitous, give difficulties.
- They may well invalidate the safety/stability/... certificates obtained by verifying delay-free abstractions of the feedback control system.

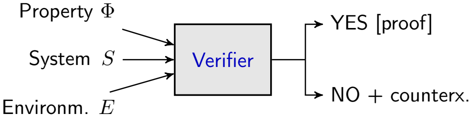
Automatic verification/synthesis methods addressing feedback delays in hybrid systems should therefore abound!

Surprisingly, they don't:

1. S. Prajna, A. Jadbabaie: *Meth. f. safety verification of time-delay syst.* (CDC'05)
2. L. Zou, M. Fränzle, ZNJ, P.N. Mosaad: *Autom. verific. of stabil. and safety* (CAV '15)
3. Z. Huang, C. Fan, S. Mitra: *Bounded invariant verification for time-delayed nonlinear networked dynamical systems* (NAHS '16)
4. M. Chen, M. Fränzle, Y. Li, P.N. Mosaad, ZNJ: *Validat. simul.-based verific.* (FM '16)
5. E. Goubault, S. Putot, and L. Sahlmann: Inner and outer approximating flowpipes for delay differential equations (CAV '18)
6. S. Feng, M. Chen, ZNJ et al.: Taming delays in dynamical systems: Unbounded verification of DDEs (CAV '19)
7. Y. Bai, T. Gan, L. Jiao, B. Xia, B. Xue and ZNJ : Switching Controller Synthesis for Time-delayed Hybrid Systems under Perturbation (HSCC'21)
8. M. Chen, M. Fränzle, Y. Li, P. Mosad and ZNJ: Indecision and delays are the parents of failure Taming them algorithmically by synthesizing delay-resilient control (Acta Informatica'21)
9. B. Xue, Q. Wang, S. Feng and ZNJ: Over- and Under-Approximating Reach Sets for Perturbed Delay Differential Equations (IEEE TAC'21)
10. ...

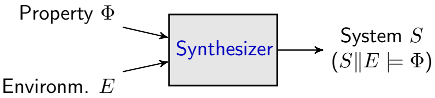
The Agenda

❶ Verification of delay differential equations



- Bounded verification
- Unbounded verification

❷ Controller synthesis for time-delayed systems



- Controller synthesis by reduction to playing safety games in the setting of discrete time
- Safety switching controller synthesis of delay hybrid systems by invariant generation and constraint solving

❸ Summary

Solving Delay Differential Equations (DDE)

A formal model of delayed feedback control

Safety Problem

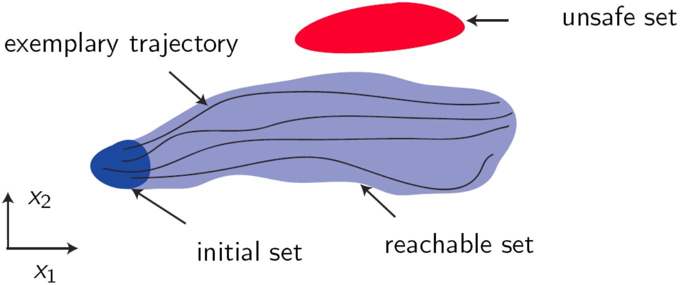
Given $T \in \mathbb{R}$, $\mathcal{X}_0 \subseteq \mathcal{X}$, $\mathcal{U} \subseteq \mathbb{R}^n$, weather

$$\forall \mathbf{x}_0 \in \mathcal{X}_0 : \quad \left(\bigcup_{t \leq T} \boldsymbol{\xi}_{\mathbf{x}_0}(t)\right) \cap \mathcal{U} = \emptyset \quad ?$$

Safety Problem

Given $T \in \mathbb{R}$, $\mathcal{X}_0 \subseteq \mathcal{X}$, $\mathcal{U} \subseteq \mathbb{R}^n$, weather

$$\forall \mathbf{x}_0 \in \mathcal{X}_0 : \quad \left(\bigcup_{t \leq T} \boldsymbol{\xi}_{\mathbf{x}_0}(t)\right) \cap \mathcal{U} = \emptyset \quad ?$$



- System is safe, if no trajectory enters the unsafe set.

Verification of DDE

Bounded verification

Verification goal: given a time-bound T show that the solutions to the DDE *on time interval* $[0, T]$ satisfy a given invariance property.

Simulation-Based Verification

- partition the initial set into a finitely smaller sets;

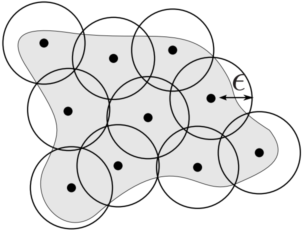


Figure: A finite ϵ-cover of the initial set of states.

- do numerical simulation on a (sufficiently dense) sample of each partitioned initial set;

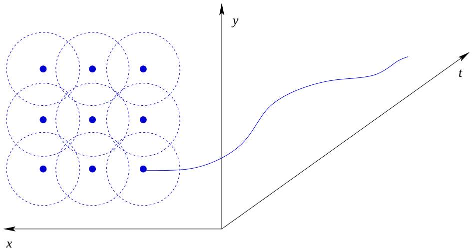


Figure: An Over-approximation of the reachable set by bloating the simulation.

Simulation-Based Verification

- partition the initial set into a finitely smaller sets;

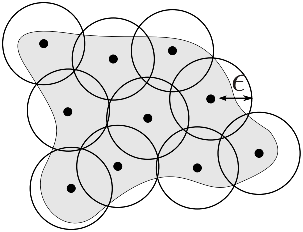


Figure: A finite ϵ-cover of the initial set of states.

- add (pessimistic) error analysis and sensitivity analysis;

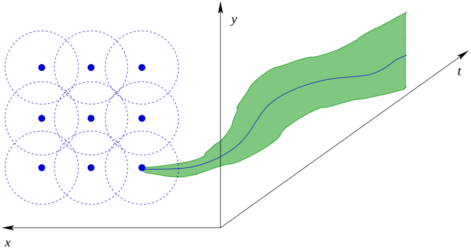


Figure: An Over-approximation of the reachable set by bloating the simulation.

- Details can be found in [FM'16].

Simulation-Based Verification

- partition the initial set into a finitely smaller sets;

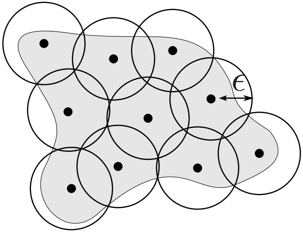


Figure: A finite ϵ-cover of the initial set of states.

- "bloat" the resulting trajectories accordingly.

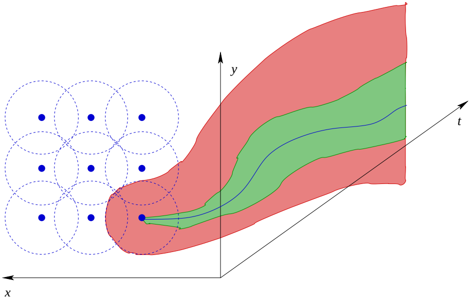


Figure: An Over-approximation of the reachable set by bloating the simulation.

- Details can be found in [FM'16].

Simulation-Based Verification

- partition the initial set into a finitely smaller sets;

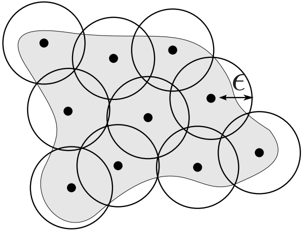


Figure: A finite ϵ-cover of the initial set of states.

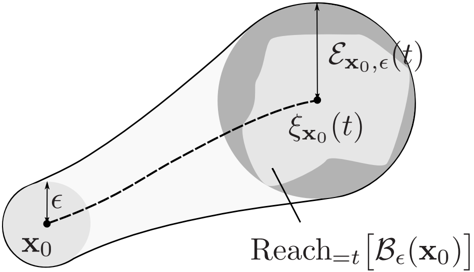


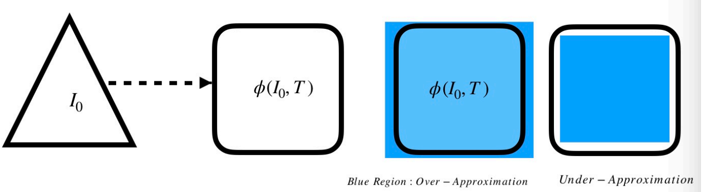
Figure: An Over-approximation of the reachable set by bloating the simulation.

- Details can be found in [FM'16].

Set-boundary Based Over/Under-Approximation

Basic idea

- Make use of the homeomorphism property to perform reachability analysis only on the initial set's boundary
- Prove that there exists a class of DDEs whose delays are small than a threshold w.r.t. their initial sets, satisfying the homemorphoism property
- Make use of sensitivity analysis to perform reachability on a subset of the initial set's boundary



- Tools: **IraPhy** (https://github.com/JianqiangDing/irafhy)
- Details can be found in ([IEEE TAC'21],[Xue et al., CAV'16], [FORMATS'17])

Unbounded Verification of DDE

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = f(\mathbf{x}(t), \mathbf{x}(t-\delta_1), \ldots, \mathbf{x}(t-\delta_n))$$

Verification goal: show that the solutions to the DDE satisfy a given invariance property (the trajectory could be infinite).

Stability of Linear Dynamics by Spectral Analysis

For linear DDEs:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = A\mathbf{x}(t) + B\mathbf{x}(t-r)$$

Stability of Linear Dynamics by Spectral Analysis

For linear DDEs:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = A\mathbf{x}(t) + B\mathbf{x}(t-r)$$

The characteristic equation:

$$\det\left(\lambda I - A - B\mathrm{e}^{-r\lambda}\right) = 0$$

Stability of Linear Dynamics by Spectral Analysis

For linear DDEs:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}\left(t\right) = A\mathbf{x}\left(t\right) + B\mathbf{x}\left(t - r\right)$$

The characteristic equation:

$$\det\left(\lambda I - A - B\mathrm{e}^{-r\lambda}\right) = 0$$

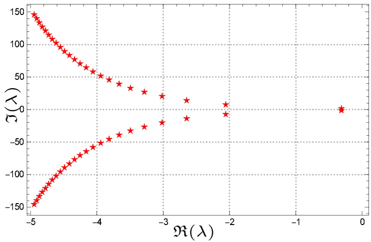
Stability of Linear Dynamics by Spectral Analysis

For linear DDEs:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = A\mathbf{x}(t) + B\mathbf{x}(t-r)$$

The characteristic equation:

$$\det\left(\lambda I - A - B\mathrm{e}^{-r\lambda}\right) = 0$$



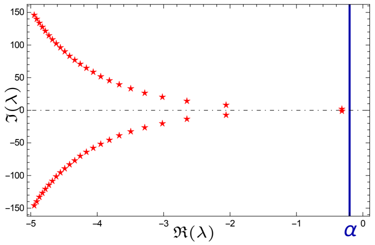
Stability of Linear Dynamics by Spectral Analysis

For linear DDEs:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = A\mathbf{x}(t) + B\mathbf{x}(t-r)$$

The characteristic equation:

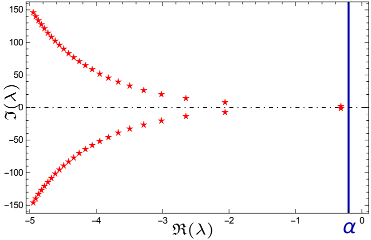
$$\det\left(\lambda I - A - B\mathrm{e}^{-r\lambda}\right) = 0$$



Stability of Linear Dynamics by Spectral Analysis

For linear DDEs:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = A\mathbf{x}(t) + B\mathbf{x}(t-r)$$



The characteristic equation:

$$\det\left(\lambda I - A - B\mathrm{e}^{-r\lambda}\right) = 0$$

Globally exponentially stable if $\forall\lambda\colon \Re(\lambda) < 0$, i.e.,

$$\exists K > 0.\, \exists\alpha < 0\colon\ \|\boldsymbol{\xi}_{\boldsymbol{\phi}}(t)\| \leq K\,\|\boldsymbol{\phi}\|\,\mathrm{e}^{\alpha t}, \quad \forall t \geq 0,\ \forall\boldsymbol{\phi}\in\mathcal{C}_r$$

Reducing Unbounded V. to Bounded V.

1. Linearize a non-linear DDE to a linear one.
2. Identify the rightmost real part of the eigenvalues (and hence α), then construct K and δ.
3. Compute T^*, as well as T' (by bounded verifiers) s.t. $\|\Omega\| < \delta$ within T'.
4. Reduce to bounded verification, i.e., $\forall T > T' + T^*$, ∞-safe $\Longleftrightarrow$ T-safe.

- Details can be found in [CAV '19].

Reducing Unbounded V. to Bounded V.

1. Linearize a non-linear DDE to a linear one.
2. Identify the rightmost real part of the eigenvalues (and hence α), then construct K and δ.
3. Compute T^*, as well as T' (by bounded verifiers) s.t. $\|\Omega\| < \delta$ within T'.
4. Reduce to bounded verification, i.e., $\forall T > T' + T^*$, ∞-safe $\Longleftrightarrow$ T-safe.

- Details can be found in [CAV '19].

Reducing Unbounded V. to Bounded V.

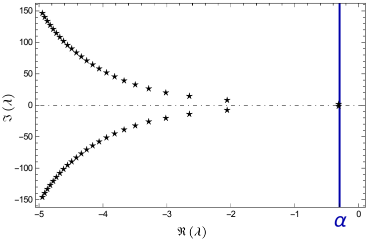
1. Linearize a non-linear DDE to a linear one.
2. Identify the rightmost real part of the eigenvalues (and hence α), then construct K and δ.
3. Compute T^*, as well as T' (by bounded verifiers) s.t. $\|\Omega\| < \delta$ within T'.
4. Reduce to bounded verification, i.e., $\forall T > T' + T^*$, ∞-safe $\iff$ T-safe.

- Details can be found in [CAV '19].

Exemplifying by Delayed Logistic Equation

Consider the safety problem over $[-r, \infty)$ with $\mathcal{X} = [-0.2, 0.2]$, $\mathcal{U} = \{u \mid |u| > 0.6\}$, under a constant delay $r = 1$.

1. Let $u = N - 1$, then $\frac{\mathrm{d}}{\mathrm{d}t}N(t) = N(t)[1 - N(t-r)] \implies \frac{\mathrm{d}}{\mathrm{d}t}u(t) = -u(t-1), \quad t \geq 0$.
2. In the second step, $\alpha = -0.3$, and $K = 3.28727$,
3. In the third step, $\delta = 0.00351678$, $T^* = 0$s and $T = 15.5$s,
4. So, the safety is guaranteed by verifying Ω over $[-1, 15.5]$ is disjoint with $\mathcal{U}$.



$$\delta = \min\left\{\delta_\epsilon, \delta_\epsilon / \left(\hat{K}\mathrm{e}^{-r\alpha}\left(1 + \|B\| \int_0^r \mathrm{e}^{-\alpha\tau}\,\mathrm{d}\tau\right)\right)\right\}$$

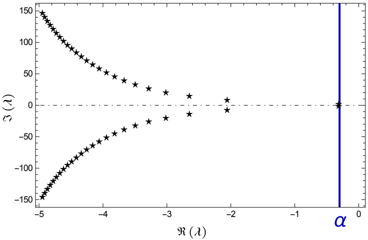
$$\delta_\epsilon = \hat{K}\mathrm{e}^{-r\alpha}\left(1 + \|B\| \int_0^r \mathrm{e}^{-\alpha\tau}\,\mathrm{d}\tau\right)\|\boldsymbol{\phi}\|\,\mathrm{e}^{\epsilon\hat{K}\mathrm{e}^{-r\alpha}t + \alpha t}$$

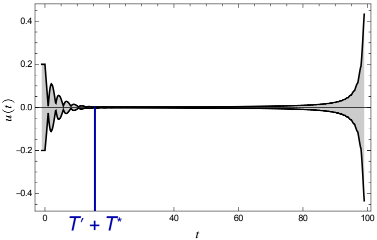
$$\epsilon \leq -\alpha/(2\hat{K}\mathrm{e}^{-r\alpha})$$

Exemplifying by Delayed Logistic Equation

Consider the safety problem over $[-r, \infty)$ with $\mathcal{X} = [-0.2, 0.2]$, $\mathcal{U} = \{u \mid |u| > 0.6\}$, under a constant delay $r = 1$.

1. Let $u = N - 1$, then $\frac{\mathrm{d}}{\mathrm{d}t}N(t) = N(t)[1 - N(t-r)] \quad \Longrightarrow$ $\frac{\mathrm{d}}{\mathrm{d}t}u(t) = -u(t-1), \quad t \geq 0.$
2. In the second step, $\alpha = -0.3$, and $K = 3.28727$,
3. In the third step, $\delta = 0.00351678$, $T^* = 0$s and $T = 15.5$s,
4. So, the safety is guaranteed by verifying Ω over $[-1, 15.5]$ is disjoint with $\mathcal{U}$.

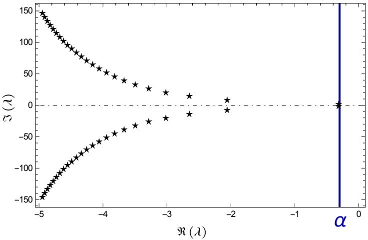


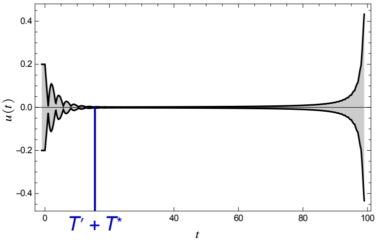


Exemplifying by Delayed Logistic Equation

Consider the safety problem over $[-r, \infty)$ with $\mathcal{X} = [-0.2, 0.2]$, $\mathcal{U} = \{u \mid |u| > 0.6\}$, under a constant delay $r = 1$.

1. Let $u = N - 1$, then $\frac{\mathrm{d}}{\mathrm{d}t}N(t) = N(t)[1 - N(t - r)] \quad \Longrightarrow$ $\frac{\mathrm{d}}{\mathrm{d}t}u(t) = -u(t-1), \quad t \geq 0.$
2. In the second step, $\alpha = -0.3$, and $K = 3.28727$,
3. In the third step, $\delta = 0.00351678$, $T^* = 0$s and $T = 15.5$s,
4. So, the safety is guaranteed by verifying Ω over $[-1, 15.5]$ is disjoint with $\mathcal{U}$.

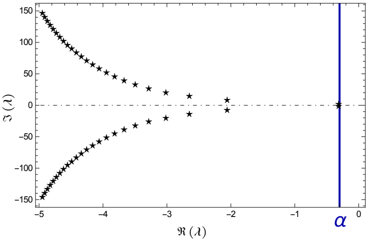


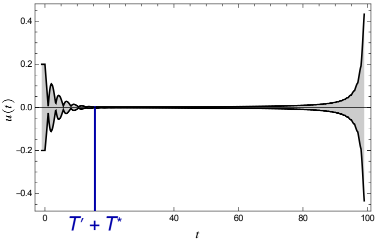


Exemplifying by Delayed Logistic Equation

Consider the safety problem over $[-r,\infty)$ with $\mathcal{X}=[-0.2,0.2]$, $\mathcal{U}=\{u \mid |u|>0.6\}$, under a constant delay $r=1$.

1. Let $u=N-1$, then $\frac{\mathrm{d}}{\mathrm{d}t}N(t)=N(t)[1-N(t-r)] \quad \Longrightarrow$ $\frac{\mathrm{d}}{\mathrm{d}t}u(t)=-u(t-1), \quad t\geq 0.$
2. In the second step, $\alpha=-0.3$, and $K=3.28727$,
3. In the third step, $\delta=0.00351678$, $T^*=0$s and $T=15.5$s,
4. So, the safety is guaranteed by verifying Ω over $[-1,15.5]$ is disjoint with $\mathcal{U}$.





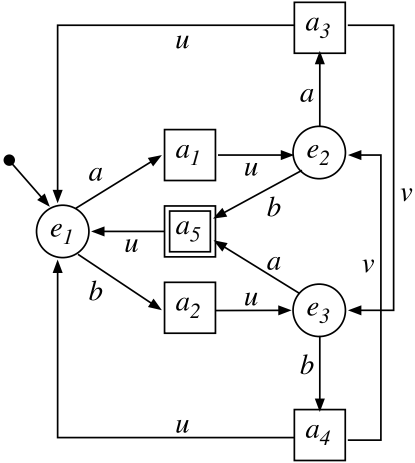
Controller Synthesis for Time-delayed Systems

Goal: Given an environment E and system specification Φ, to synthesize a system S such that $E \| S \models \Phi$

Controller Synthesis for Discrete Time-delayed Systems by Reduction to Discrete Safety Games

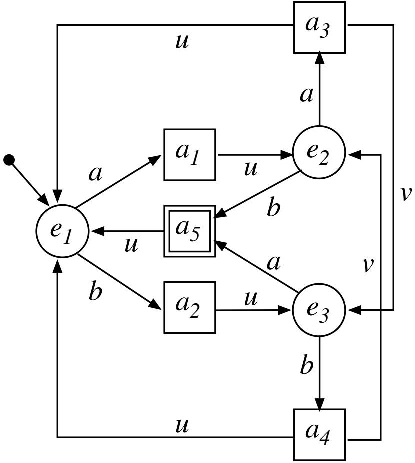
Goal: Given an environment E and system specification Φ, to synthesize a system S such that $E \| S \models \Phi$

A Trivial Safety Game



Goal: Avoid $\boxed{\boxed{a_5}}$ by appropriate actions of player e.

A Trivial Safety Game



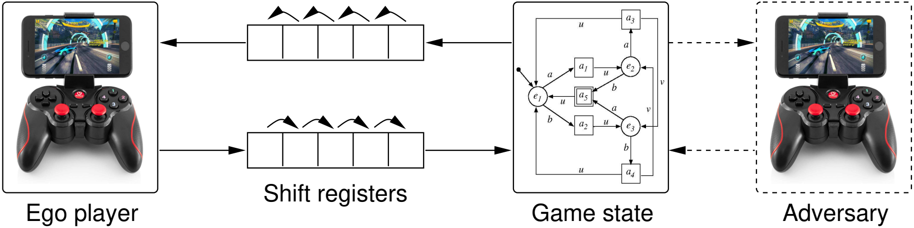
Goal: Avoid $\boxed{a_5}$ by appropriate actions of player e.

Strategy: May always play "a" except in e_3:

$$\begin{array}{lcl} e_1, e_2 & \mapsto & a \\ e_3 & \mapsto & b \end{array}$$

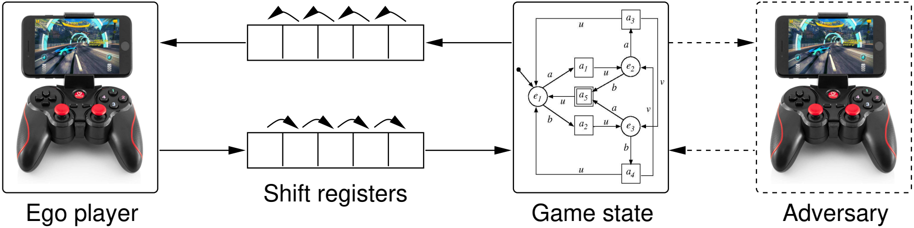
Proterties: Determinacy and memoryless.

Playing Safety Game Subject to Discrete Delay



Observation: It doesn't make an observable difference for the joint dynamics whether delay occurs in perception, actuation, or both.

Playing Safety Game Subject to Discrete Delay

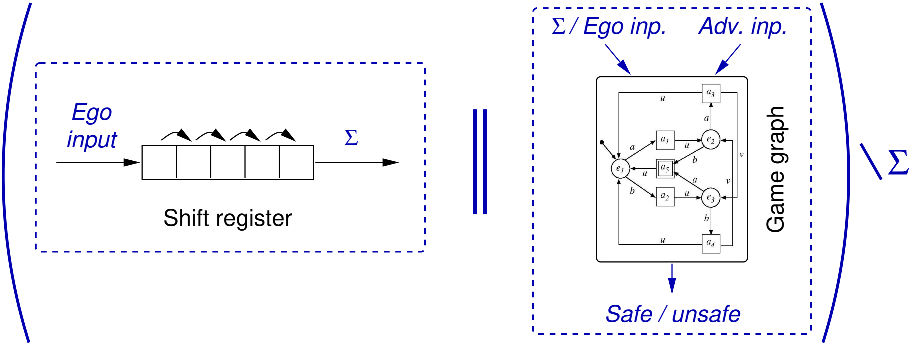


Observation: It doesn't make an observable difference for the joint dynamics whether delay occurs in perception, actuation, or both.

Consequence: There is an[1] obvious reduction to a safety game of perfect information.

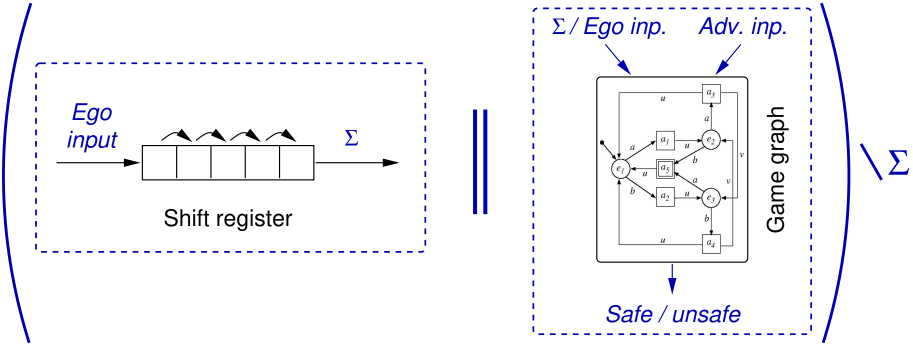
Reduction to Delay-Free Games

from Ego-Player Perspective



Reduction to Delay-Free Games

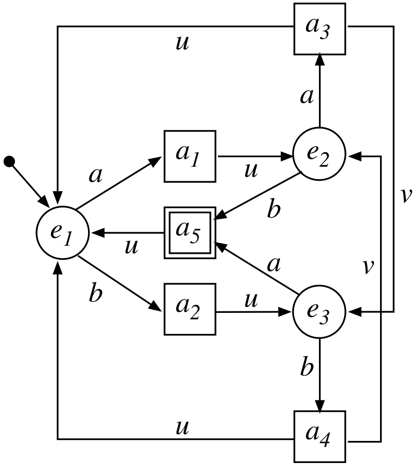
from Ego-Player Perspective



- 🙂 Safety games w. delay can be solved algorithmically ([M. Zimmermann. LICS'18, GandALF'17], [F. Klein & M. Zimmermann. ICALP'15, CSL'15]).
- ☹ Game graph incurs blow-up by factor $|\text{Alphabet(ego)}|^{\text{delay}}$.
- 🙂 A more efficient algorithm is presented in [ATVA'18, Acta Informatica'21]

The Simple Safety Game

...but with Delay

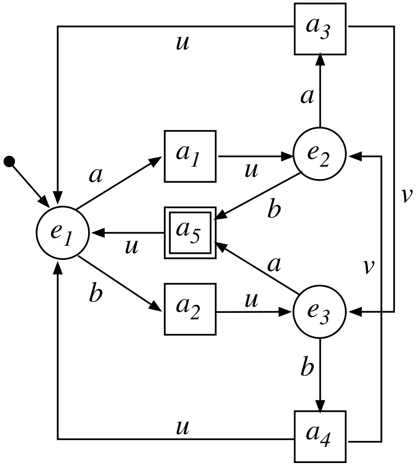


No delay:

$$\begin{aligned} e_1, e_2 &\mapsto a \\ e_3 &\mapsto b \end{aligned}$$

1 step delay: Strategy?

$$\begin{aligned} a_1, a_4 &\mapsto a \\ a_2, a_3 &\mapsto b \end{aligned}$$



No delay:

$$\begin{aligned} e_1, e_2 &\mapsto a \\ e_3 &\mapsto b \end{aligned}$$

1 step delay: Strategy?

$$\begin{aligned} a_1, a_4 &\mapsto a \\ a_2, a_3 &\mapsto b \end{aligned}$$

2 steps delay: Strategy?

$$e_1 \mapsto \begin{cases} a & \text{if 2 steps back an "}a\text{" was issued,} \\ b & \text{if 2 steps back a "}b\text{" was issued.} \end{cases}$$

$$\begin{aligned} e_2 &\mapsto b \\ e_3 &\mapsto a \end{aligned}$$

Need memory!

Incremental Synthesis of Delay-Tolerant Strategies

Observation: A winning strategy for delay $k' > k$ can always be utilized for a safe win under delay k.

Consequence: That a position is winning for delay k is a necessary condition for it being winning under delay $k' > k$.

Incremental Synthesis of Delay-Tolerant Strategies

Observation: A winning strategy for delay $k' > k$ can always be utilized for a safe win under delay k.

Consequence: That a position is winning for delay k is a necessary condition for it being winning under delay $k' > k$.

Idea: Incrementally filter out loss states & incrementally synthesize winning strategy for the remaining:

1. Synthesize winning strategy for underlying delay-free safety game.
2. For each winning state, lift strategy from delay k to $k + 1$.
3. Remove states where this does not succeed.
4. Repeat from 2 until either delay-resilience suffices or initial state turns lossy.

- Details can be found in [ATVA '18, Acta Informatica'21].

Safety Switching Controller Synthesis for Delay Hybrid Systems by Invariant Generation and Constraint Solving

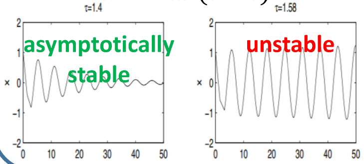
Goal: Given an environment E and system specification Φ, to synthesize a system S such that $E \| S \models \Phi$

Delay Hybrid Automata

◆ **Two kinds of delay** occur in CPS.

Delay in continuous dynamics

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



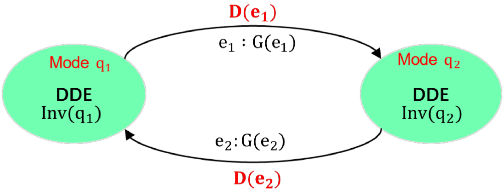
Delay in discrete dynamics

Delay Hybrid Automata

Definition (Delay Hybrid Automaton, DHA)

A DHA is a tuple $\mathcal{H} = (Q, X, U, Inv, X_0, F, E, D, G, R)$

- U: a set of continuous functionals;
- Inv: an invariant Inv(q) for each mode q $\in$ Q ;
- R: E $\times$ $X_D \rightarrow$ U : reset functions;
- ..



Synthesis Problem

- Given a DHA $\mathcal{H} = (\mathrm{Q}, \mathrm{X}, \mathrm{U}, \mathrm{Inv}, \mathrm{X}_0, \mathrm{F}, \mathrm{E}, \mathrm{D}, \mathrm{G}, \mathrm{R})$
- a safety property

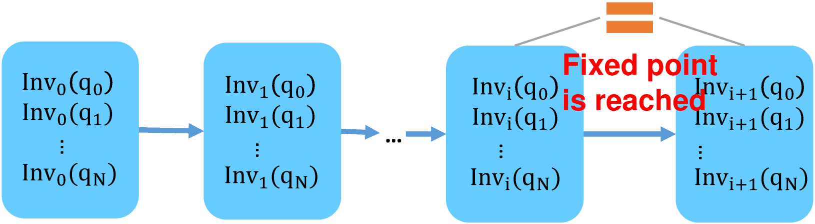
a safe switching controller synthesis problem is to synthesize a new DHA $\mathcal{H}^* = (\mathrm{Q}, \mathrm{X}, \mathrm{U}^*, \mathrm{Inv}^*, \mathrm{X}_0^*, \mathrm{F}, \mathrm{E}, \mathrm{D}, \mathrm{G}^*, \mathrm{R})$ such that

- $\mathcal{H}^*$ is safe;
- $\mathcal{H}^*$ is a refinement of $\mathcal{H}$;
- $\mathcal{H}^*$ is non-blocking.

- Details can be found in [HSCC'21, SCM 51(1)].

Invariant Generation

- Generate a global invariant for delay hybrid system by computing a fixed point.
  - Generate a strengthened differential invariant for each mode
  - Generate a strengthened guarded for each transition



Differential Invariant Generation

Linear DDE: $\dot{\boldsymbol{x}}(t) = A\,\boldsymbol{x}(t) + B\boldsymbol{x}(t-r) + C\boldsymbol{w}(t)$

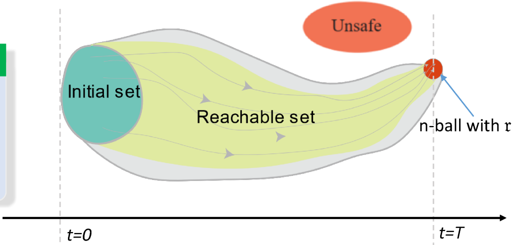
- Reduce to T-invariant, i.e., ∀T > T^*, ∞-invariant ⇔T-invariant
- Compute a safe over-approximate reachable set in T

Exponentially convergent to a ball:

if there exist a constant $\gamma > 0$ and a non-decreasing function $\kappa(\cdot)$ such that

$$\left\|\xi_{\phi}^{w}(t)\right\|_{\infty} \leq r + \kappa(\|\phi\|_{\infty})e^{-\gamma t}, \qquad \forall\, t \geq 0$$

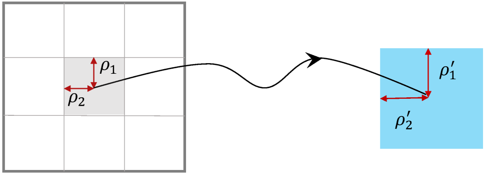
holds for all $\phi \in C$, $\|w(t)\|_{\infty} \leq \overline{w} \ \ \forall\, t \geq 0.$



Differential Invariant Generation

Linear DDE: $\dot{\boldsymbol{x}}(t) = A\,\boldsymbol{x}(t) + B\boldsymbol{x}(t-r) + C\boldsymbol{w}(t)$

- ☐Reduce to T-invariant, i.e., ∀T > T^*, ∞-invariant ⇔T-invariant
- ☐Compute a safe over-approximate reachable set in T



Differential Invariant Generation

Non-linear DDE: $\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{x}(t-r), \boldsymbol{w}(t))$

linearize

Linear DDE: $\dot{\boldsymbol{x}}(t) = A\,\boldsymbol{x}(t) + B\boldsymbol{x}(t-r) + C\boldsymbol{w}(t) + \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{x}(t-r))$

□Reduce to T-invariant, i.e., ∀T > T^*, ∞-invariant ⇔T-invariant

Locally exponentially convergent to a ball:

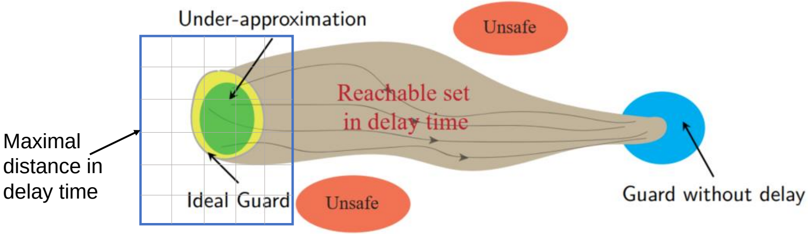
if there exist a constant $\gamma > 0, l > 0$ and a non-decreasing function $\kappa(\cdot)$ such that

$$\|\phi(t)\|_\infty \le l \Rightarrow \left\|\xi_\phi^w(t)\right\|_\infty \le r + \kappa(\|\phi\|_\infty)e^{-\gamma t}, \qquad \forall\, t \ge 0$$

holds for all $\phi \in C$, $\|\boldsymbol{w}(t)\|_\infty \le \bar{w}$ $\forall\, t \ge 0$.

Guard Synthesis under Delay

- Synthesize guard condition without delay using invariant;
- Synthesize guard condition under delay by backward reachable set computation.



Summing Up

Summary

Problem: We face

- increasingly wide-spread use of networked distributed sensing and ctrl.,
- substantial feedback delay thus affecting hybrid control schemes,
- delays impact controllability and control performance in both the discrete and the continuous parts.

Summary

Problem: We face

- increasingly wide-spread use of networked distributed sensing and ctrl.,
- substantial feedback delay thus affecting hybrid control schemes,
- delays impact controllability and control performance in both the discrete and the continuous parts.

Status: Uncovered algorithms

- for veryfing continuous differential dynamics represented as a DDE with a single, constant or multiple small delays,
- for efficient control synthesis for discrete safety games under delay,
- for controller synthesis for delay hybrid systems based on invariant generation and constraint solving.

Summary

Problem: We face

- increasingly wide-spread use of networked distributed sensing and ctrl.,
- substantial feedback delay thus affecting hybrid control schemes,
- delays impact controllability and control performance in both the discrete and the continuous parts.

Status: Uncovered algorithms

- for veryfing continuous differential dynamics represented as a DDE with a single, constant or multiple small delays,
- for efficient control synthesis for discrete safety games under delay,
- for controller synthesis for delay hybrid systems based on invariant generation and constraint solving.

Future Work:

- DDE exhibiting state-dependent or/and stochastic delay,
- **Invariant generation for time-delayed systems** (on-going)
  - Some first try was done by Prajna&Jadbabaie [CDC'05], and recently by Ames *et al.* [ACC'19,ACC'21] and by us [SCIS'21]
  - But more general invariant generation for DDE is still challenging.

I would like to thank all collaborators on this topic, including Prof. Martin Fränzle, Prof. Bican Xia, Prof. Li Jiao, Dr. Liang Zou, Dr. Mingshuai Chen, Dr. Peter Nazier Mosaad, Dr. Xue Bai, Dr. Yangjia Li, Dr. Yunjun Bai, Dr. Ting Gan, Mr. Shenghua Feng and Mr. Wenyou Liu.

Thanks for your attention

Questions