Towards Delays in Dynamical and Control Systems

Simulation-Based Verification & Game Theory

Mingshuai Chen

State Key Lab. of Computer Science, Institute of Software, Chinese Academy of Sciences
Joint work with Martin Fränzle, Yangjia Li, Naijun Zhan and Peter N. Mosaad

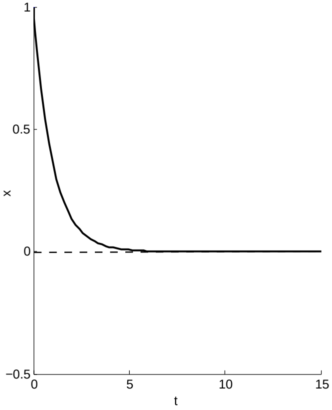
Saarbrücken, July 2016

Outline

Outline

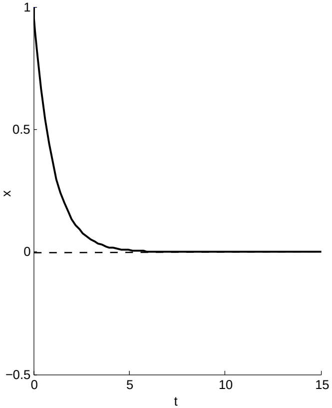
Delayed Differential Dynamics

$$\begin{cases} \dot{x}(t) = -x(t) \\ x(0) = 1 \end{cases}$$

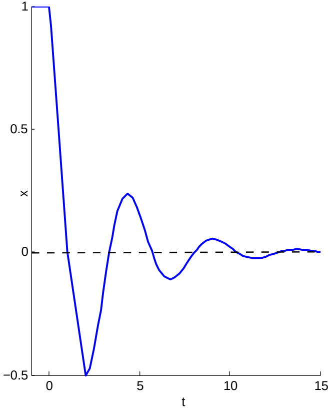


Delayed Differential Dynamics

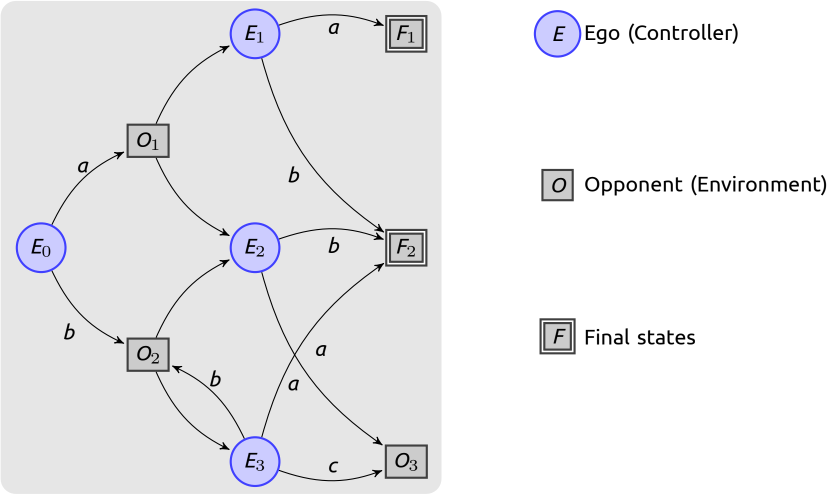
$$\begin{cases} \dot{x}(t) = -x(t) \\ x(0) = 1 \end{cases}$$



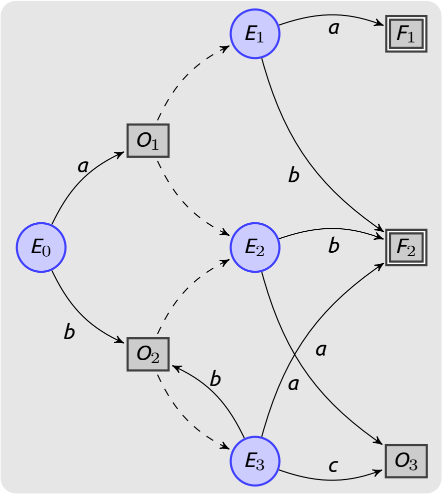
$$\begin{cases} \dot{x}(t) = -x(t-1) \\ x([-1,0]) \equiv 1 \end{cases}$$

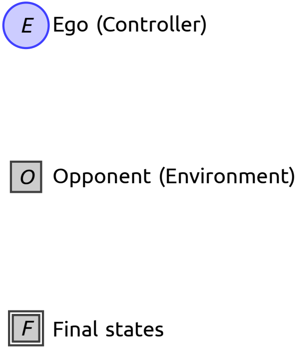


Games with Perfect Information



Games with Delayed Information





Outline

Delayed Dynamical Systems

Delayed Dynamical Systems

$$\begin{cases} \dot{\mathbf{x}}(t) & = & \boldsymbol{f}(\mathbf{x}(t), \mathbf{x}(t-r_1), \ldots, \mathbf{x}(t-r_k)), \quad t \in [0, \infty) \\ \mathbf{x}(t) & \equiv & \mathbf{x}_0 \in \Theta, \quad t \in [-r_k, 0] \end{cases}$$

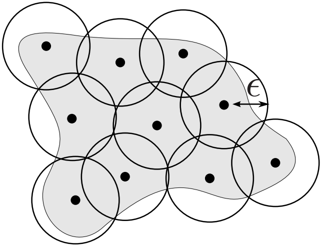
The unique *solution* (*trajectory*) : $\xi_{\mathbf{x}_0}(t) : [-r_k, \infty) \mapsto \mathbb{R}^n$.

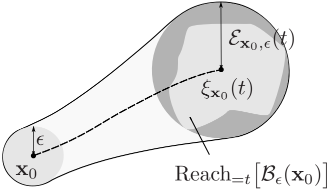
Safety Verification Problem

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

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

Basic Idea [1]





1. Figures are taken from [A. DonzDonzé and O. Maler, HSCC'07].

Verification Algorithm

Algorithm 1: Simulation-based Verification for Delayed Dynamical Systems

input : The dynamics $\boldsymbol{f}(\mathbf{x}, \mathbf{u})$, delay term r, initial set $\mathcal{X}_0$, unsafe set $\mathcal{U}$, time bound T, precision ϵ.

`/* initialization */`

$S \leftarrow \mathcal{X}_0;\ \mathcal{R} \leftarrow \emptyset;\ \delta \leftarrow dia(\mathcal{X}_0)/2;\ \tau \leftarrow \tau_0;$
while $S \neq \emptyset$ **do**
    $\mathcal{X} \leftarrow \delta$-Partition$(S)$;
    if $\delta < \epsilon$ **then**
        return (UNKNOWN, $\mathcal{R}$);
    for $\mathcal{B}_\delta(\mathbf{x}_0) \in \mathcal{X}$ **do**
        $\langle \mathbf{t}, \mathbf{y}, \mathbf{d} \rangle \leftarrow$ Simulation$(\mathcal{B}_\delta(\mathbf{x}_0), \boldsymbol{f}(\mathbf{x}, \mathbf{u}), r, \tau, T)$;
        $\mathcal{T} \leftarrow \bigcup_{n=0}^{N-1} conv(B_{\mathbf{d}_n}(\mathbf{y}_n) \cup B_{\mathbf{d}_{n+1}}(\mathbf{y}_{n+1}))$;
        if $\mathcal{T} \cap \mathcal{U} = \emptyset$ **then**
            $S \leftarrow S \backslash \mathcal{B}_\delta(\mathbf{x}_0);\quad \mathcal{R} \leftarrow \mathcal{R} \cup \mathcal{T}$;
        else if $\exists i.\ \mathcal{B}_{\mathbf{d}_i}(\mathbf{y}_i) \subseteq \mathcal{U}$ **then**
            return (UNSAFE, $\mathcal{T}$);
        else
            $S \leftarrow S \backslash \mathcal{B}_\delta(\mathbf{x}_0);\quad \delta \leftarrow \delta/2$;
return (SAFE, $\mathcal{R}$);

Local Error Bounds

$$E(t) = \begin{cases} d_0, & \text{if } t = 0, \\ E(t_i) + (t - t_i)e_{i+1}, & \text{if } t \in [t_i, t_{i+1}]. \end{cases}$$

Local Error Bounds

$$E(t) = \begin{cases} d_0, & \text{if } t = 0, \\ E(t_i) + (t - t_i)e_{i+1}, & \text{if } t \in [t_i, t_{i+1}]. \end{cases}$$

Validation Property :

$$\xi_{\mathbf{x}_0}(t) \in \mathcal{B}_{E(t)}\left(\frac{(t - t_i)\mathbf{y}_i + (t_{i+1} - t)\mathbf{y}_{i+1}}{t_{i+1} - t_i}\right), \text{ for each } t \in [t_i, t_{i+1}].$$

Simulation Algorithm

Algorithm 2: Simulation: a validated DDE solver producing rigorous bounds

input : The initial set $\mathcal{B}_\delta(\mathbf{x}_0)$, dynamics $\boldsymbol{f}(\mathbf{x}, \mathbf{u})$, delay term r, step size τ, time bound T.

output: A triple $\langle \mathbf{t}, \mathbf{y}, \mathbf{d} \rangle$, where the components represent lists, with the same length, respectively for the time points, numerical approximations (possibly multi-dimensional), and the rigorous local error bounds.

```
/* initializing the lists, whose indices start from -1 */
```
$\mathbf{t} \leftarrow [\![-\tau, 0]\!]$; $\mathbf{y} \leftarrow [\![\mathbf{x}_0, \mathbf{x}_0]\!]$; $\mathbf{d} \leftarrow [\![0, \delta]\!]$;

```
/* r has to be divisible by τ (in FP numbers) */
```
$n \leftarrow 0$; $m \leftarrow r/\tau$;

while $\mathbf{t}_n < T$ **do**

- $t_{n+1} \leftarrow \mathbf{t}_n + \tau$;
- `/* approximating` y_{n+1} `using` *`forward Euler method`* `*/`
- $y_{n+1} \leftarrow \mathbf{y}_n + \boldsymbol{f}(\mathbf{y}_n, \mathbf{y}_{n-m}) * \tau$;
- `/* computing error slope by constrained optimization */`
- $e_n \leftarrow$ **Find** minimum e s.t.

$$\begin{cases} \|\boldsymbol{f}(\mathbf{x} + t * \mathbf{f}, \mathbf{u} + t * \mathbf{g}) - \boldsymbol{f}(\mathbf{y}_n, \mathbf{y}_{n-m})\| \leq e - \sigma, \text{ for} \\ \forall t \in [0, \tau] \\ \forall \mathbf{x} \in \mathcal{B}_{\mathbf{d}_n}(\mathbf{y}_n) \\ \forall \mathbf{u} \in \mathcal{B}_{\mathbf{d}_{n-m}}(\mathbf{y}_{n-m}) \\ \forall \mathbf{f} \in \mathcal{B}_e(\boldsymbol{f}(\mathbf{y}_n, \mathbf{y}_{n-m})) \\ \forall \mathbf{g} \in \mathcal{B}_{e_{n-m}}(\boldsymbol{f}(\mathbf{y}_{n-m}, \mathbf{y}_{n-2m})); \end{cases} \tag{9}$$

- $d_{n+1} \leftarrow \mathbf{d}_n + \tau e_n$;
- `/* updating the lists by appending the extrapolation */`
- $\mathbf{t} \leftarrow [\![\mathbf{t}, t_{n+1}]\!]$; $\mathbf{y} \leftarrow [\![\mathbf{y}, y_{n+1}]\!]$; $\mathbf{d} \leftarrow [\![\mathbf{d}, d_{n+1}]\!]$;
- $n \leftarrow n + 1$;

return $\langle \mathbf{t}, \mathbf{y}, \mathbf{d} \rangle$;

Solving the Optimization by HySAT-II

$$\text{find } \min\{e \geq 0 \mid \forall x : \phi(x, e) \implies \psi(x, e)\}$$

Solving the Optimization by HySAT-II

$$\text{find } \min\{e \geq 0 \mid \forall x : \phi(x, e) \implies \psi(x, e)\}$$

$$\text{find } \max\{e \geq 0 \mid \exists x : \phi(x, e) \wedge \neg\psi(x, e)\}$$

Simulation Algorithm

Theorem (Correctness)

Suppose the maximum index of the lists is N, then $\forall t \in [0, T]$ *and* $\forall \mathbf{x} \in B_\delta(\mathbf{x}_0)$,

$$\xi_{\mathbf{x}}(t) \subseteq \bigcup_{n=0}^{N-1} conv(\mathcal{B}_{\mathbf{d}_n}(\mathbf{y}_n) \cup \mathcal{B}_{\mathbf{d}_{n+1}}(\mathbf{y}_{n+1})).$$

Simulation Algorithm

Theorem (Correctness)

Suppose the maximum index of the lists is N, then $\forall t \in [0, T]$ *and* $\forall \mathbf{x} \in B_\delta(\mathbf{x}_0)$,

$$\xi_{\mathbf{x}}(t) \subseteq \bigcup_{n=0}^{N-1} conv(\mathcal{B}_{\mathbf{d}_n}(\mathbf{y}_n) \cup \mathcal{B}_{\mathbf{d}_{n+1}}(\mathbf{y}_{n+1})).$$

Theorem (Completeness)

Suppose the function $\boldsymbol{f}$ *is continuously differentiable in both arguments and the dynamical system is solvable for time interval* $[0, T]$, *then for any* $\varepsilon > 0$, *there exists* δ, τ *and* σ *such that the optimization problem (9) has a solution* e_n *for all* $n \leq \frac{T}{\tau}$, *and moreover* $\mathbf{d}_n \leq \varepsilon$.

Simulation Algorithm

Theorem (Correctness)

Suppose the maximum index of the lists is N, then $\forall t \in [0, T]$ *and* $\forall \mathbf{x} \in B_\delta(\mathbf{x}_0)$,

$$\xi_{\mathbf{x}}(t) \subseteq \bigcup_{n=0}^{N-1} conv(\mathcal{B}_{\mathbf{d}_n}(\mathbf{y}_n) \cup \mathcal{B}_{\mathbf{d}_{n+1}}(\mathbf{y}_{n+1})).$$

Theorem (Completeness)

Suppose the function $\boldsymbol{f}$ *is continuously differentiable in both arguments and the dynamical system is solvable for time interval* $[0, T]$, *then for any* $\varepsilon > 0$, *there exists* δ, τ *and* σ *such that the optimization problem (9) has a solution* e_n *for all* $n \leq \frac{T}{\tau}$, *and moreover* $\mathbf{d}_n \leq \varepsilon$.

Further extension to simulations with variable stepsize.

Delayed Logistic Equation

$$\dot{N}(t) = N(t)[1 - N(t - r)]$$

Delayed Logistic Equation

$$\dot{N}(t) = N(t)[1 - N(t - r)]$$

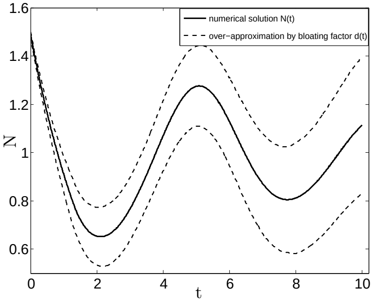


Figure : $\mathcal{X}_0 = \mathcal{B}_{0.01}(1.49)$, $r = 1.3$, $\tau_0 = 0.01$, $T = 10$s.

Delayed Logistic Equation

$$\dot{N}(t) = N(t)[1 - N(t - r)]$$

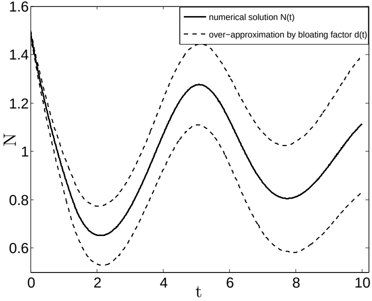


Figure : $\mathcal{X}_0 = \mathcal{B}_{0.01}(1.49)$, $r = 1.3$, $\tau_0 = 0.01$, $T = 10$s.

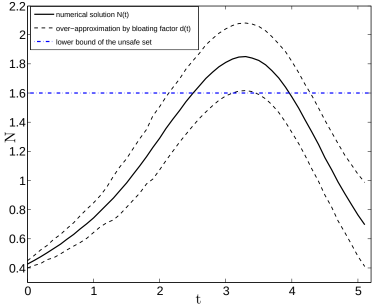
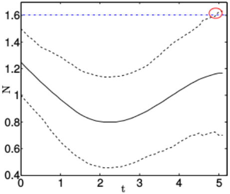
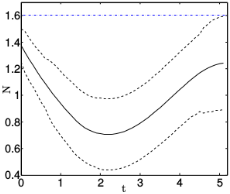


Figure : Over-approximation rigorously proving unsafe, with $r = 1.7$, $\mathcal{X}_0 = \mathcal{B}_{0.025}(0.425)$, $\tau_0 = 0.1$, $T = 5$s, $\mathcal{U} = \{N | N > 1.6\}$.

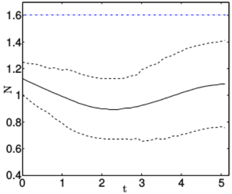
Delayed Logistic Equation



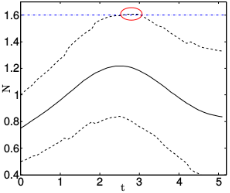
(a) An initial over-approximaion of trajectories starting from $\mathcal{B}_{0.225}(1.25)$. It overlaps with the unsafe set (s. circle). Initial set is consequently split (cf. Figs. 3b, 3c).



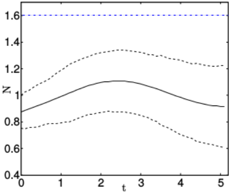
(b) All trajectories starting from $\mathcal{B}_{0.125}(1.375)$ are proven safe within the time bound, as the over-approximation does not intersect with the unsafe set.



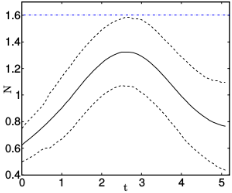
(c) Initial state set $\mathcal{B}_{0.125}(1.125)$ is verified to be safe as well.



(d) $\mathcal{B}_{0.25}(0.75)$ yields overlap w. unsafe; the ball is partitioned again (Figs. 3e, 3f).



(e) All trajectories originating from $\mathcal{B}_{0.125}(0.875)$ are provably safe.

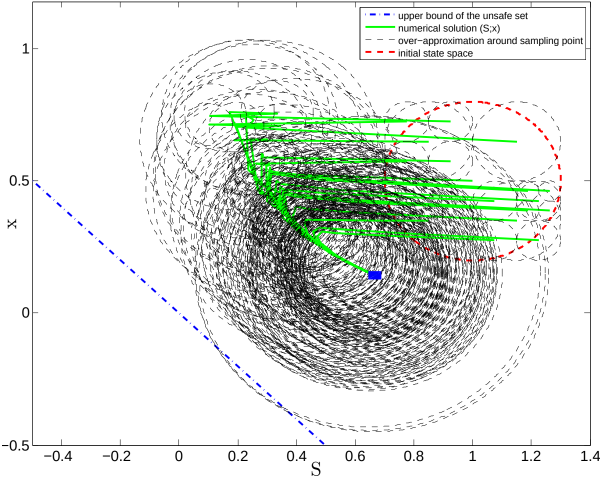


(f) All trajectories originating from $\mathcal{B}_{0.125}(0.625)$ are provably safe as well.

Fig. 3: The logistic system (13) is proven safe through 6 rounds of simulation with base stepsize $\tau_0 = 0.1$. Delay $r = 1.3$, initial state set $\mathcal{X}_0 = \{N | N \in [0.5, 1.5]\}$, time bound $T = 5$s, unsafe set $\{N | N > 1.6\}$.

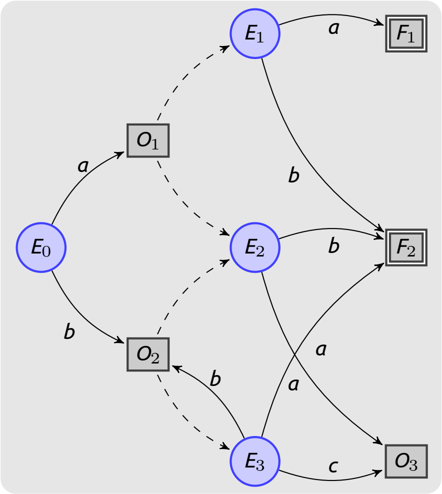
Delayed Microbial Growth

$$\begin{cases} \dot{S}(t) = 1 - S(t) - f(S(t))x(t) \\ \dot{x}(t) = e^{-r} f(S(t-r))x(t-r) - x(t) \end{cases}$$



Outline

Observation-Based Subset-Construction

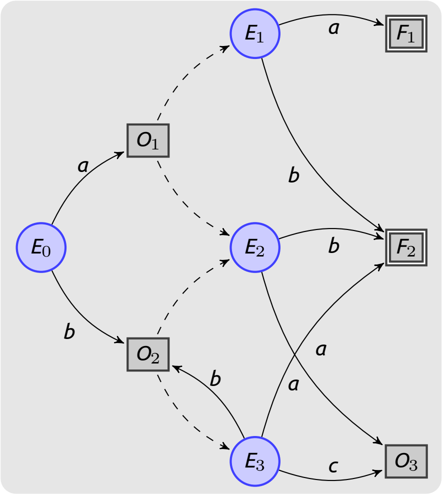


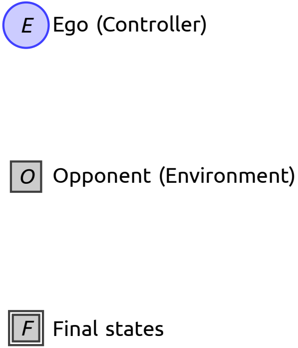
Ego (Controller)

Opponent (Environment)

Final states

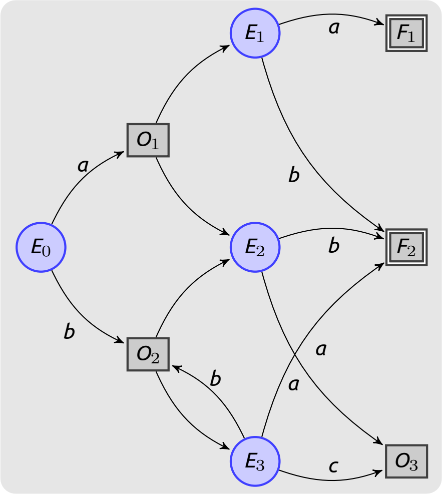
Observation-Based Subset-Construction



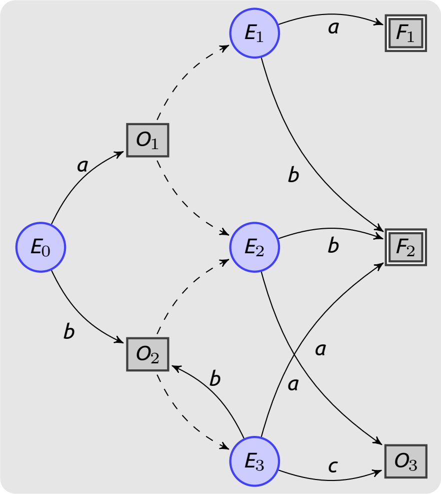


Is it possible to avoid the explosion of the state space ?

Lift the Step of Delays from $0, 1, \ldots, k$ by Labelling

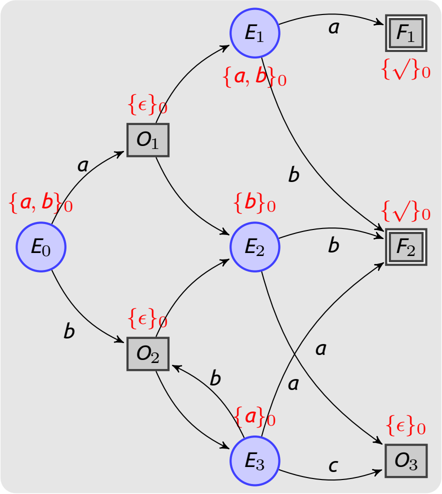


$delay = 0$

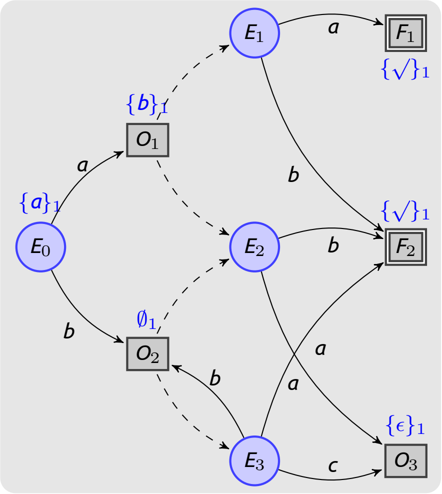


$delay = 1$

Lift the Step of Delays from $0, 1, \ldots, k$ by Labelling



$delay = 0$



$delay = 1$

Outline

Concluding Remarks

- Verifying delayed differential dynamics by validated simulation : a combination of numerical methods with SMT solvers.
- A sketch idea on synthesizing controllers in games of delayed information ; non-deterministic strategy & almost winning ?