Concurrent Stochastic Lossy Channel Games

Daniel Stan (EPITA)

October 6, 2023

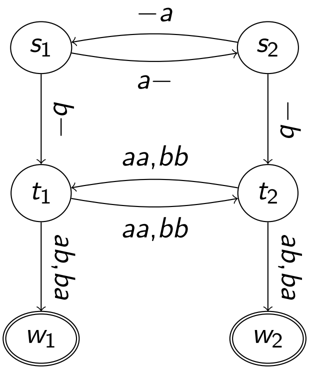
joint work with **Anthony W. Lin** (MPI-SWS, RPTU),
Muhammad Najib (Heriot-Watt) and
Parosh Abdullah (Uppsala University)

Outline

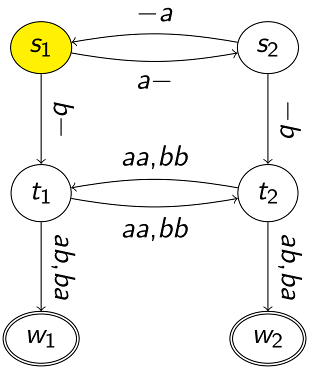
1. Finite Concurrent Games
2. Lossy Channel Systems
3. Infinite Concurrent Games

Stochastic Concurrent Finite Games

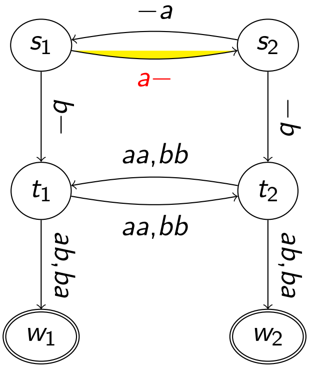
Concurrent Game on a Finite graph



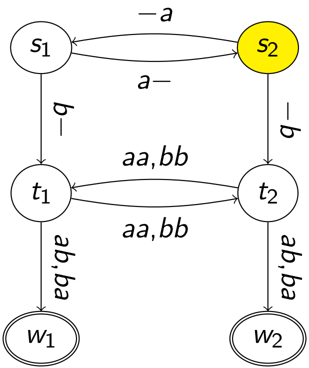
Concurrent Game on a Finite graph



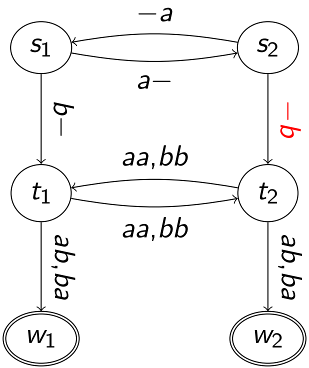
Concurrent Game on a Finite graph



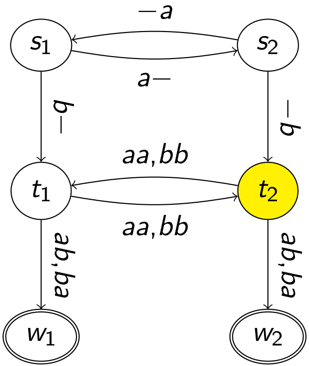
Concurrent Game on a Finite graph



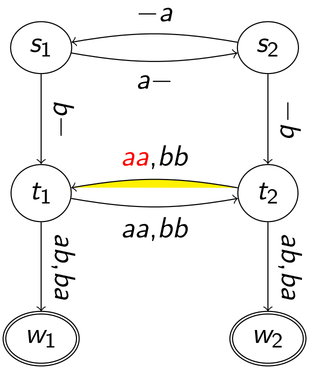
Concurrent Game on a Finite graph



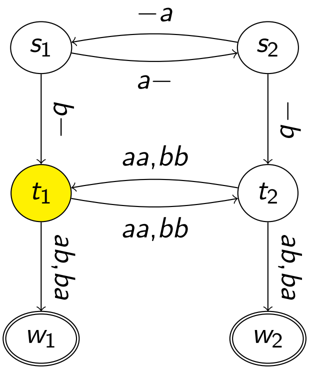
Concurrent Game on a Finite graph



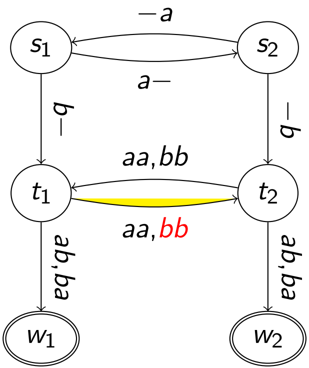
Concurrent Game on a Finite graph



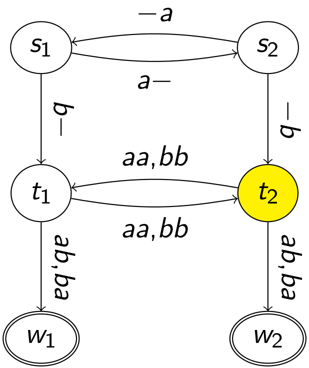
Concurrent Game on a Finite graph



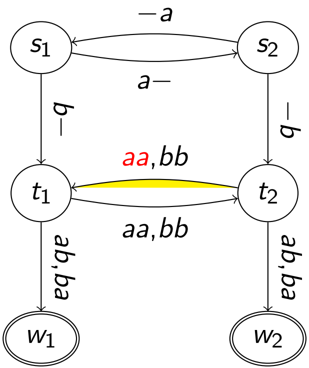
Concurrent Game on a Finite graph



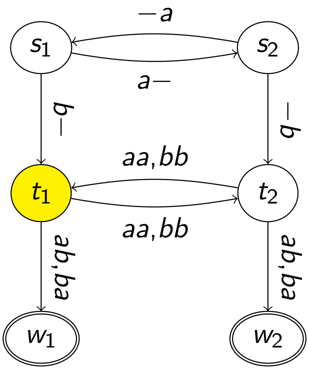
Concurrent Game on a Finite graph



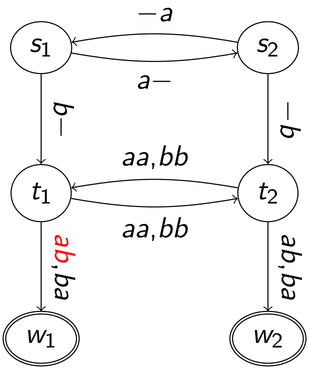
Concurrent Game on a Finite graph



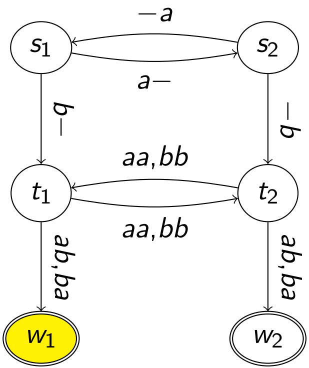
Concurrent Game on a Finite graph



Concurrent Game on a Finite graph

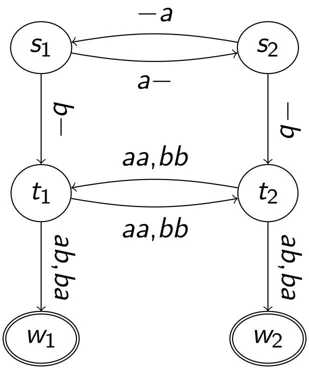


Concurrent Game on a Finite graph



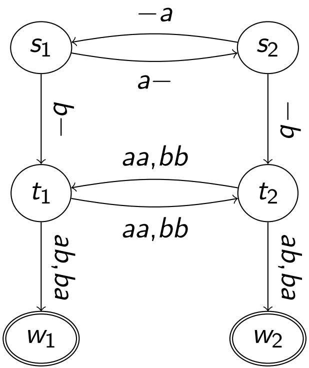
- Game on graph
- Several agents
- For each state $s \in S$, each $i \in \mathrm{Agt}$, set of $\mathrm{Act}_i(s)$ actions
- Transitions played concurrently

Concurrent Game on a Finite graph



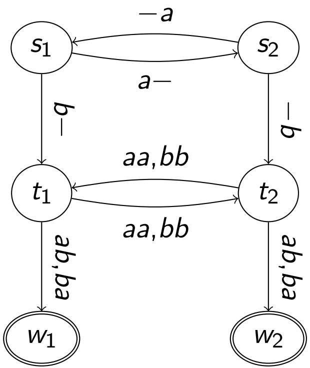
- Game on graph
- Several agents
- For each state $s \in S$, each $i \in \mathrm{Agt}$, set of $\mathrm{Act}_i(s)$ actions
- Transitions played concurrently
- Also: stochastic transitions (players and environment)

Concurrent Game on a Finite graph



- Game on graph
- Several agents
- For each state $s \in S$, each $i \in \mathrm{Agt}$, set of $\mathrm{Act}_i(s)$ actions
- Transitions played concurrently
- Also: stochastic transitions (players and environment)
- Zero sum if two players with **opposite** objectives.
  (Example here:
)

Concurrent Game on a Finite graph



- Game on graph
- Several agents
- For each state $s \in S$, each $i \in \mathrm{Agt}$, set of $\mathrm{Act}_i(s)$ actions
- Transitions played concurrently
- Also: stochastic transitions (players and environment)
- Zero sum if two players with **opposite** objectives.
  (Example here: $\Diamond w_1 \not\equiv \neg \Diamond w_2$ so not zero-sum)

Strategy Classes

A strategy for player i is:

$$\sigma_i : S^+ \rightarrow \mathrm{Dist}(\mathrm{Act})$$

- **D**etermistic: only one action with probability 1;

$$\forall h \in S^+, \exists \alpha : \sigma_i(h)[\alpha] = 1$$

Strategy Classes

A strategy for player i is:

$$\sigma_i : S^+ \to \mathrm{Dist}(\mathrm{Act})$$

- **D**etermistic: only one action with probability 1;

$$\forall h \in S^+, \exists \alpha : \sigma_i(h)[\alpha] = 1$$

- **P**ositional: depends only on the current state;

$$\forall h \in S^*, \forall s \in S^+, \sigma_i(h \cdot s) = \sigma_i(s)$$

Strategy Classes

A strategy for player i is:

$$\sigma_i : S^+ \to \mathrm{Dist}(\mathrm{Act})$$

- **D**etermistic: only one action with probability 1;

$$\forall h \in S^+, \exists \alpha : \sigma_i(h)[\alpha] = 1$$

- **P**ositional: depends only on the current state;

$$\forall h \in S^*, \forall s \in S^+, \sigma_i(h \cdot s) = \sigma_i(s)$$

Strategy Classes

A strategy for player i is:

$$\sigma_i : S^+ \to \mathrm{Dist}(\mathrm{Act})$$

- **D**etermistic: only one action with probability 1;

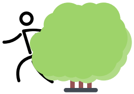
$$\forall h \in S^+, \exists \alpha : \sigma_i(h)[\alpha] = 1$$

- **P**ositional: depends only on the current state;

$$\forall h \in S^*, \forall s \in S^+, \sigma_i(h \cdot s) = \sigma_i(s)$$

Can't we just play with **DP** strategies only?

Skirmish Game [dAHK07]



$A_1(s_0) = \{h, r\}$

$A_2(s_0) = \{s, w\}$

Played:

Skirmish Game [dAHK07]

$A_1(s_0) = \{h, r\}$

$A_2(s_0) = \{s, w\}$

Played: hw

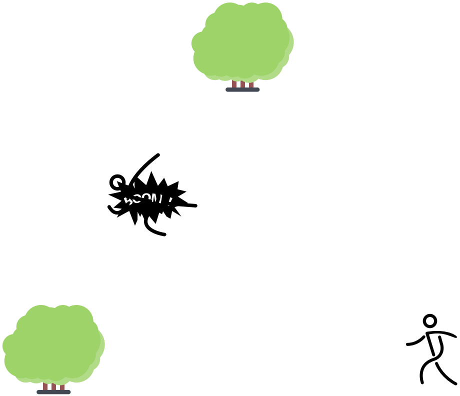
Skirmish Game [dAHK07]

$A_1(s_0) = \{h, r\}$

$A_2(s_0) = \{s, w\}$

Played: rs

Skirmish Game [dAHK07]



$A_1(s_0) = \{h, r\}$ $A_2(s_0) = \{s, w\}$

Played: rs

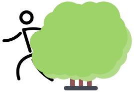
Skirmish Game [dAHK07]

$A_1(s_0) = \{h, r\}$

$A_2(s_0) = \{s, w\}$

Played: rs

Skirmish Game [dAHK07]

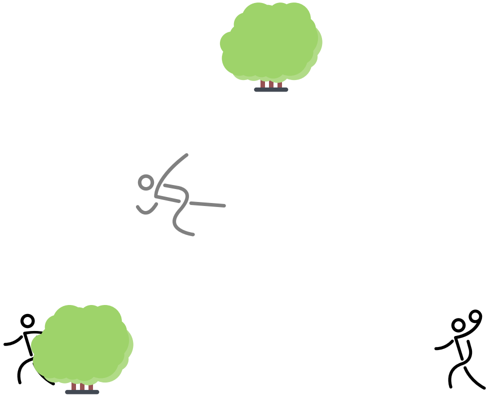


$A_1(s_0) = \{h, r\}$

$A_2(s_0) = \{s, w\}$

Played: rw

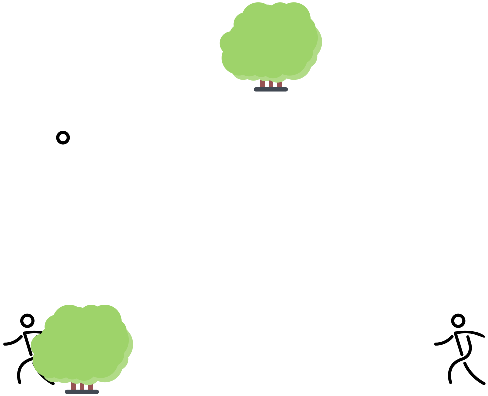
Skirmish Game [dAHK07]



$A_1(s_0) = \{h, r\}$ $A_2(s_0) = \{s, w\}$

Played: rw

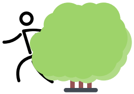
Skirmish Game [dAHK07]



$A_1(s_0) = \{h, r\}$ $A_2(s_0) = \{s, w\}$

Played: rw

Skirmish Game [dAHK07]



$A_1(s_0) = \{h, r\}$ $A_2(s_0) = \{s, w\}$

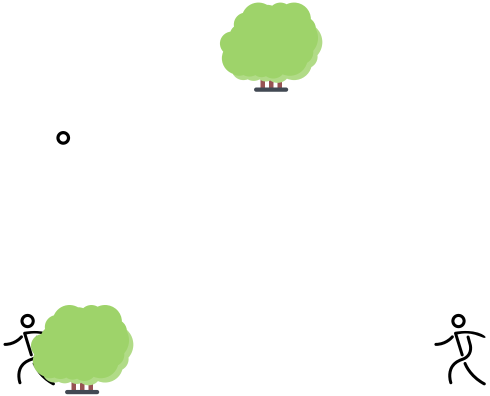
Played: hs

Skirmish Game [dAHK07]

$A_1(s_0) = \{h, r\}$ $A_2(s_0) = \{s, w\}$

Played: hs

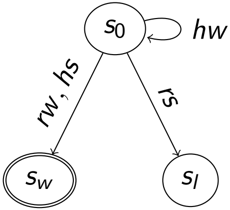
Skirmish Game [dAHK07]



$A_1(s_0) = \{h, r\}$ $A_2(s_0) = \{s, w\}$

Played: hs

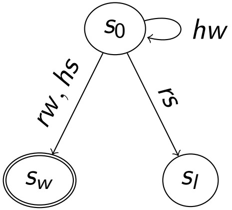
Skirmish Game: Analysis



P1: **reach with probability** 1
aka $\mathrm{AS}(\Diamond R)$
P2: **stay away of** R **with pp**
aka $\mathrm{NZ}(\Box \overline{R})$.

- $\forall \epsilon > 0$; Player 1 can win with probability $1 - \epsilon$,

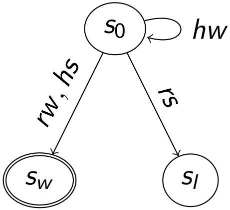
Skirmish Game: Analysis



P1: **reach with probability** 1
aka $\mathrm{AS}(\Diamond R)$
P2: **stay away of** R **with pp**
aka $\mathrm{NZ}(\Box\overline{R})$.

- $\forall \epsilon > 0$; Player 1 can win with probability $1 - \epsilon$,
- For a given σ_1, player 2 can enforce s_l **with pp.**;

Skirmish Game: Analysis

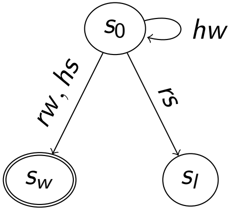


P1: **reach with probability** 1
aka $\mathrm{AS}(\Diamond R)$
P2: **stay away of** R **with pp**
aka $\mathrm{NZ}(\Box \overline{R})$.

- $\forall \epsilon > 0$; Player 1 can win with probability $1 - \epsilon$,
- For a given σ_1, player 2 can enforce s_l **with pp.**;
- For any **finite memory** strategy σ_2, player 1 can go to s_w **almost-surely**.

Still,

Skirmish Game: Analysis



P1: **reach with probability** 1
aka $\text{AS}(\Diamond R)$
P2: **stay away of** R **with pp**
aka $\text{NZ}(\Box\overline{R})$.

- $\forall \epsilon > 0$; Player 1 can win with probability $1 - \epsilon$,
- For a given σ_1, player 2 can enforce s_l **with pp.**;
- For any **finite memory** strategy σ_2, player 1 can go to s_w **almost-surely**.

Still, **Player 2** wins this game but with an **infinite memory strategy**.

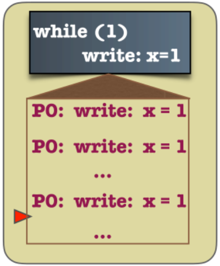
$$\forall n, \sigma_2(s_0^n)[s] = \left(\frac{1}{2}\right)^{2^{-n}}$$

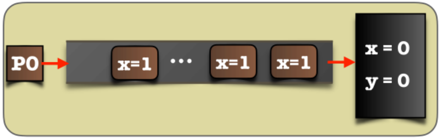
Channel Systems

Channel Systems (FIFO): Motivations

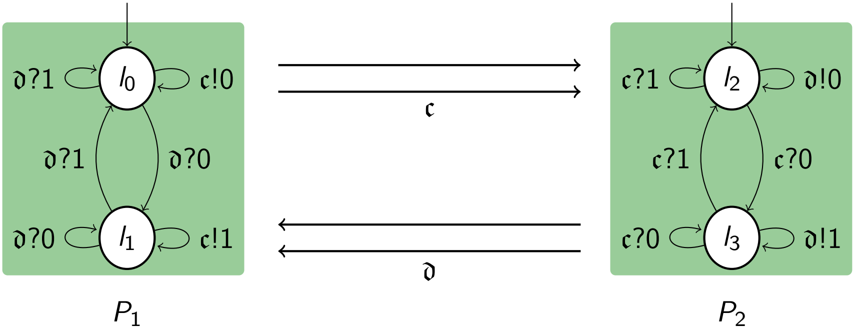
Modelisation and verification of systems with:

- Network transmissions;
- Transactional operations;
- TSO semantics [AABN18]

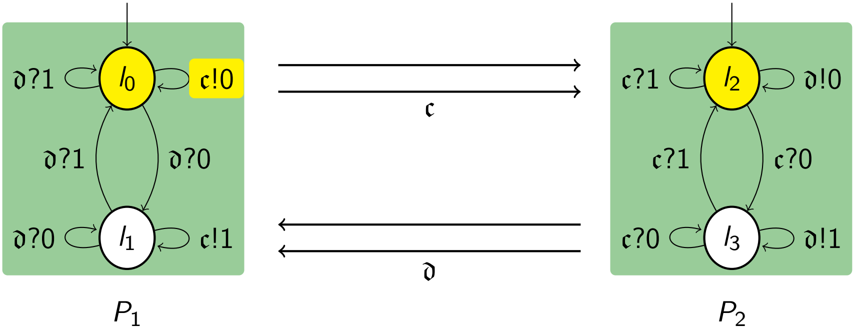




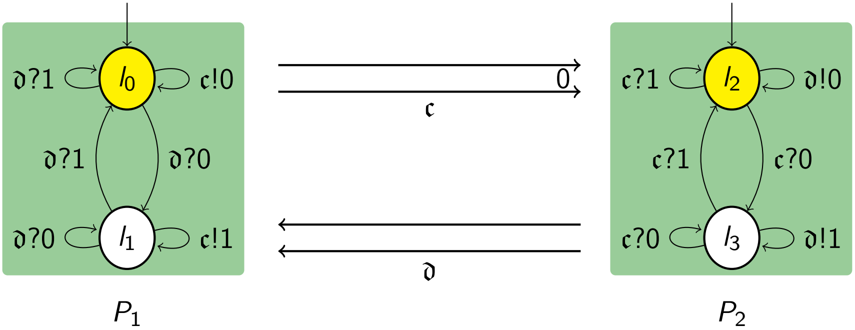
Channel Systems (FIFO): Motivations



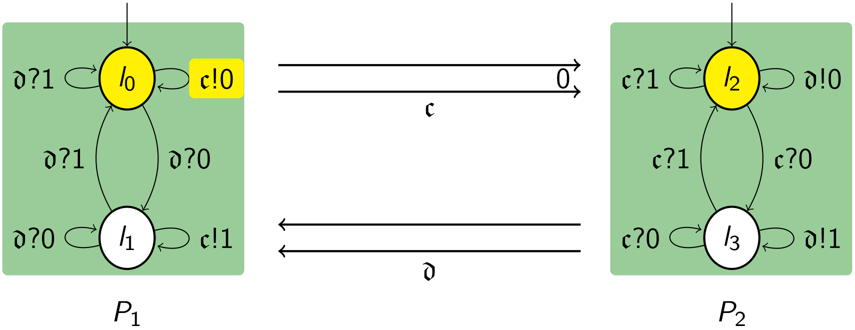
Channel Systems (FIFO): Motivations



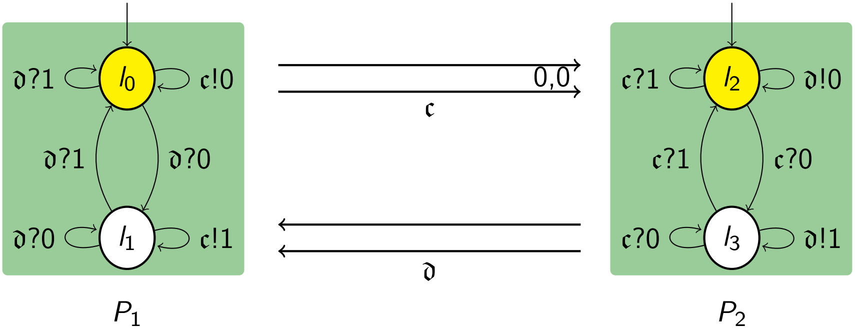
Channel Systems (FIFO): Motivations



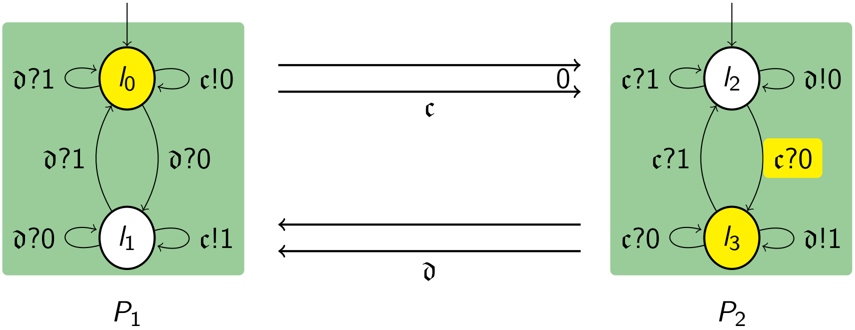
Channel Systems (FIFO): Motivations



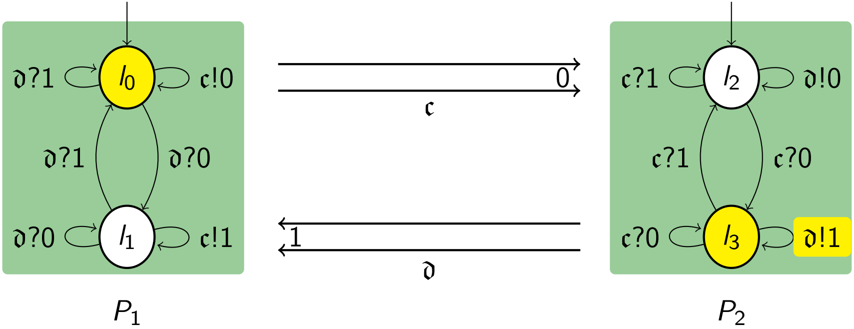
Channel Systems (FIFO): Motivations



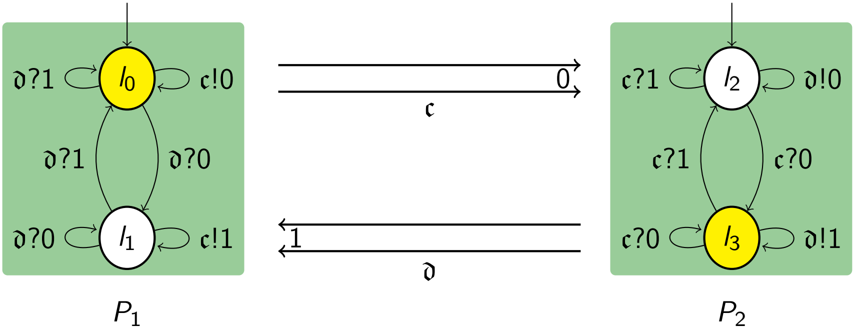
Channel Systems (FIFO): Motivations



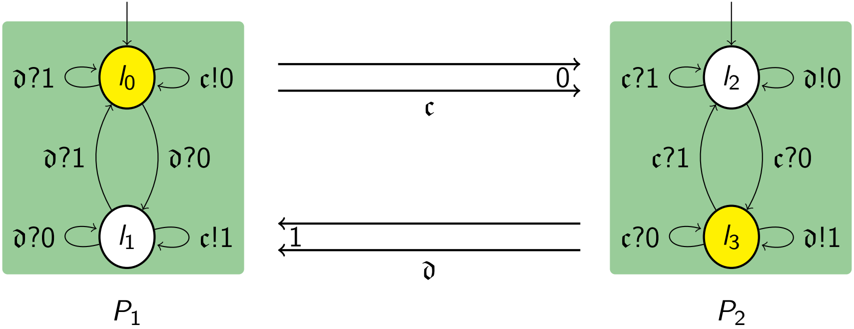
Channel Systems (FIFO): Motivations



Channel Systems (FIFO): Motivations



Channel Systems (FIFO): Motivations

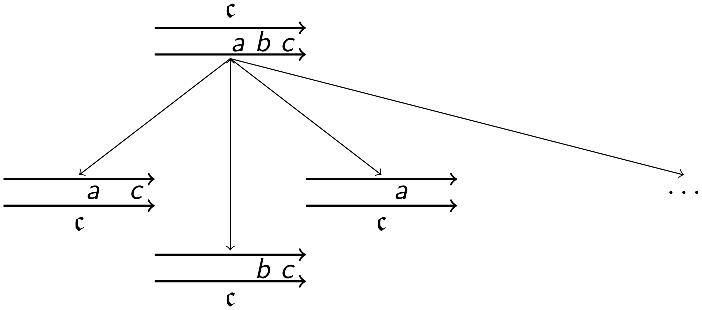


Communication:

- send a message m on $\mathfrak{c}$: $\mathfrak{c}!m$
- receive a message $\mathfrak{c}?m$ pop a message $m \in M$, **only if** m **was at the end of the queue**.

Lossiness assumption

Assumption: at every round, every message *may* disappear.



Effect of a transition (l, f, l') on state $s = l \cdot w$:

1. Change to location l';
2. Apply the channel operation operation f on w;
3. Drop from w to $w' \preceq w$: **subword ordering**.

Subword-ordering $\preceq$ is a well-quasi order [FS01]

Definition ([FS01])

$(S, \preceq)$ is a well-quasi-order (WQO) if:

$$\forall (s_i)_{i \in \mathbb{N}} \in S^{\mathbb{N}}, \exists i < j : s_i \preceq s_j$$

$(S, \to, \preceq)$ is a well-structured transition system:

$$\forall s, s', t, \quad \begin{array}{ccc} t & & \\ \curlyvee\!\mid & & \\ s & \longrightarrow & s' \end{array}$$

Subword-ordering $\preceq$ is a well-quasi order [FS01]

Definition ([FS01])

$(S, \preceq)$ is a well-quasi-order (WQO) if:

$$\forall (s_i)_{i \in \mathbb{N}} \in S^{\mathbb{N}}, \exists i < j : s_i \preceq s_j$$

$(S, \rightarrow, \preceq)$ is a well-structured transition system:

$$\forall s, s', t, \quad \begin{array}{ccc} t & & \\ \curlyvee\!\!| & & \\ s & \longrightarrow & s' \end{array}$$

Subword-ordering $\preceq$ is a well-quasi order [FS01]

Definition ([FS01])

$(S, \preceq)$ is a well-quasi-order (WQO) if:

$$\forall (s_i)_{i\in\mathbb{N}} \in S^{\mathbb{N}}, \exists i < j : s_i \preceq s_j$$

$(S, \to, \preceq)$ is a well-structured transition system:

$$\forall s, s', t, \quad \begin{array}{ccc} t & \longrightarrow & t'^{\,\exists} \\ \curlyvee\!| & & \curlyvee\!| \\ s & \longrightarrow & s' \end{array}$$

Backward-reachability algorithm for ND scheduler

Theorem

Given $l_0, l_f \in L$ in a lossy channel system, one can decide whether $l_0 \rightarrow^* l_f$.

Backward-reachability algorithm for ND scheduler

Theorem

Given $l_0, l_f \in L$ in a lossy channel system, one can decide whether $l_0 \rightarrow^* l_f$.

Sketch.

- Given $X \subseteq S$, the set $\mathrm{Pre}(X) = \{s \mid s \rightarrow X\} \subseteq S$ is **upward-closed** hence **regular**;
- Moreover, if X is a **regular** set, then $\mathrm{Pre}(X)$ can be computed.
- The computation $\cup_{i \geq n}\mathrm{Pre}^i(\{l_f\})$ converges in finite time.

Backward-reachability algorithm for ND scheduler

Theorem

Given $l_0, l_f \in L$ in a lossy channel system, one can decide whether $l_0 \to^* l_f$.

Sketch.

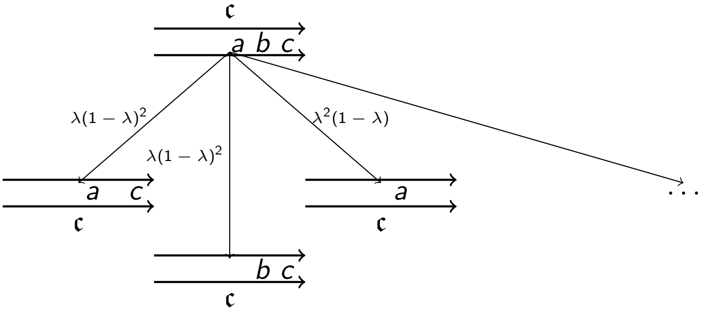
- Given $X \subseteq S$, the set $\mathrm{Pre}(X) = \{s \mid s \to X\} \subseteq S$ is **upward-closed** hence **regular**;
- Moreover, if X is a **regular** set, then $\mathrm{Pre}(X)$ can be computed.
- The computation $\cup_{i \geq n} \mathrm{Pre}^i(\{l_f\})$ converges in finite time.

The result is written $[\![E(\Diamond R)]\!]$ and one just needs to check membership of l_0.

Lossiness in the probabilistic case

Stochastic case: the semantics is a Markov chain (S, Pr).

Local lossiness assumption: at every step, there is a positive probability $\lambda \in (0, 1)$, that a letter in the channel is dropped. Every message drop event is **independent** from the others.



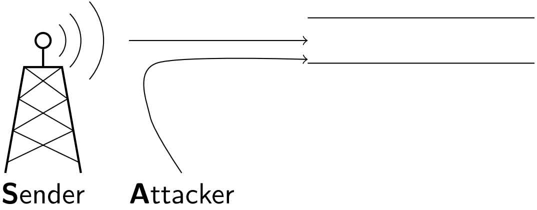
Concurrent Games + Lossy Channel Systems

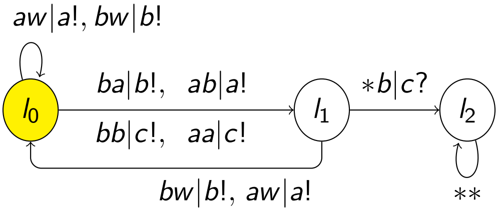
Concurrent Games + Lossy Channel Systems

=

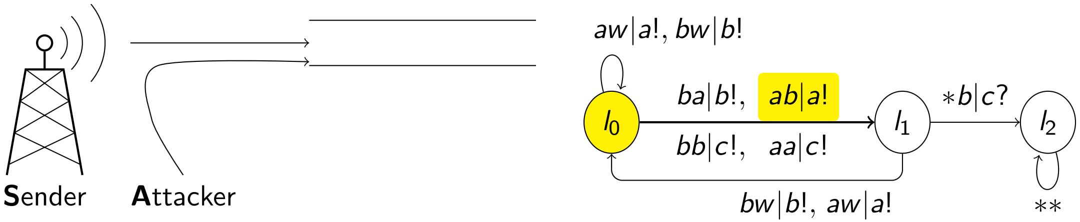
Infinite State Games

Example

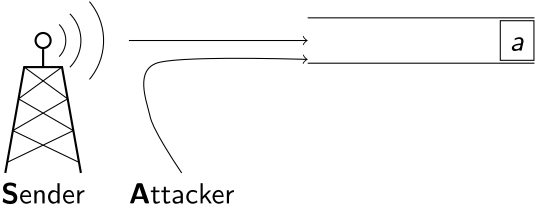


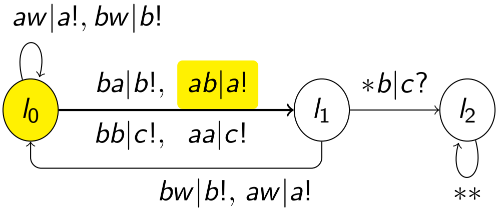


Example

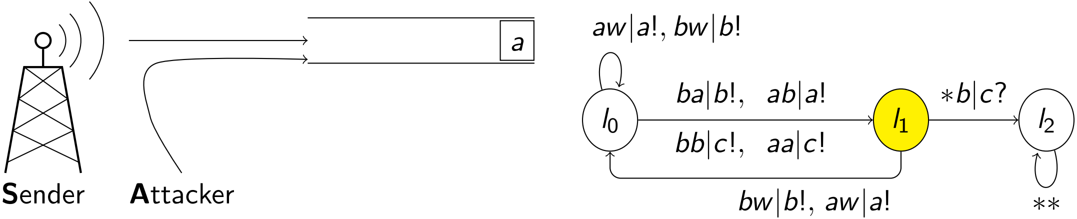


Example

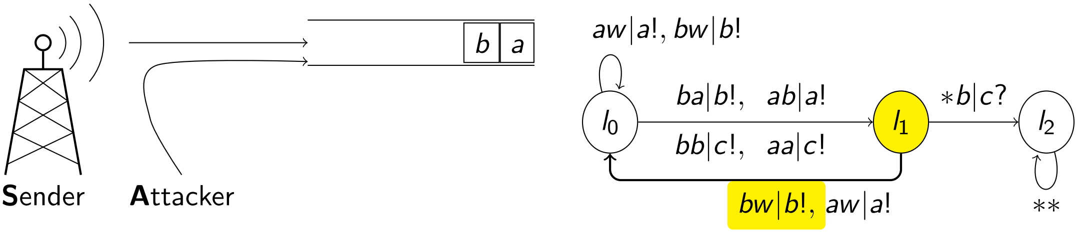




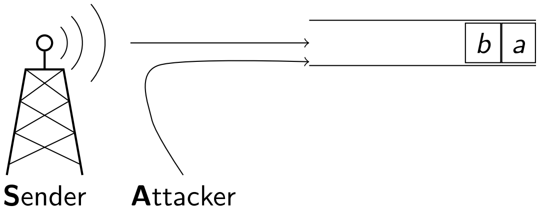
Example

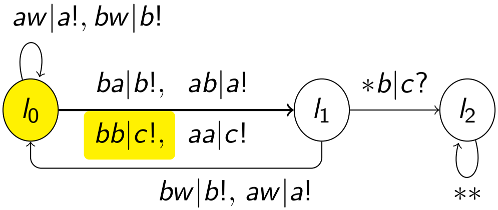


Example

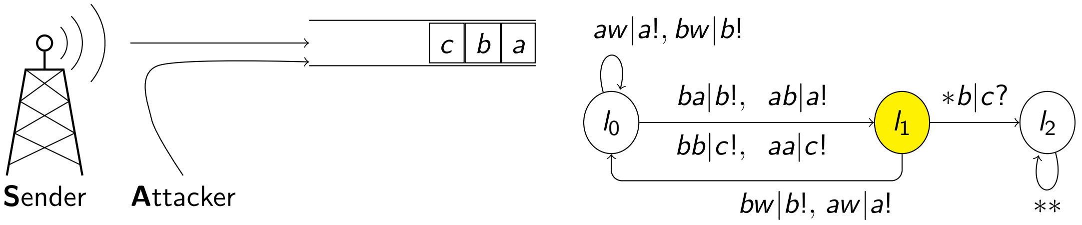


Example

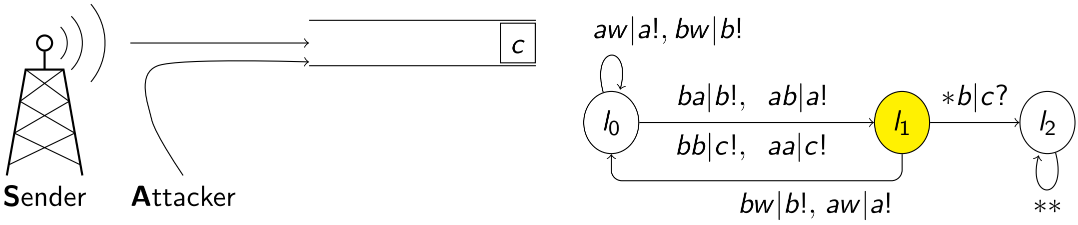




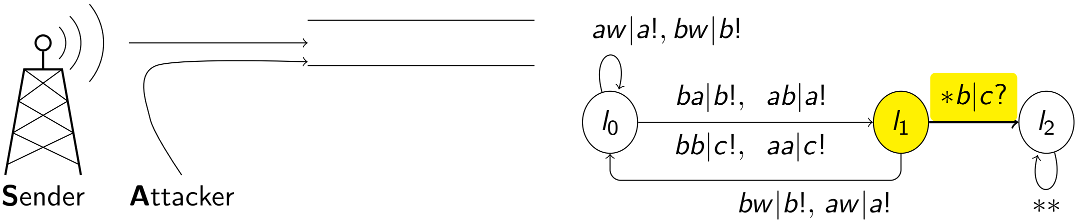
Example



Example



Example



Zero-sum case

The fixed point algorithms ([FS01, dAHK07]) still apply!

Zero-sum case

The fixed point algorithms ([FS01, dAHK07]) still apply!

- $\mathrm{Pre}_i(X) = \{s \mid i$ can enforce reaching X with positive probability$\}$.
- Pre_i is computable for regular sets.
- **Termination**: Thanks to WQO's property [FS01].
- **Correctness**: Finite Attractor property [BBS06].

Zero-sum case

The fixed point algorithms ([FS01, dAHK07]) still apply!

- $\mathrm{Pre}_i(X) = \{s \mid i$ can enforce reaching X with positive probability$\}$.
- Pre_i is computable for regular sets.
- **Termination**: Thanks to WQO's property [FS01].
- **Correctness**: Finite Attractor property [BBS06].

Theorem

Let $R \subseteq L \cdot M^*$ a **regular** *set of configurations. One can compute the set of winning configurations for player 1 for:*

- **Existential** /**Positive P.** *Reachability:* $[\![\mathrm{E}(\Diamond R)]\!]$*;*
- **Almost Sure** *Reachability:* $[\![\mathrm{AS}(\Diamond R)]\!]$*;*

Zero-sum case

The fixed point algorithms ([FS01, dAHK07]) still apply!

- $\mathrm{Pre}_i(X) = \{s \mid i$ can enforce reaching X with positive probability$\}$.
- Pre_i is computable for regular sets.
- **Termination**: Thanks to WQO's property [FS01].
- **Correctness**: Finite Attractor property [BBS06].

Theorem

Let $R \subseteq L \cdot M^*$ a **regular** *set of configurations. One can compute the set of winning configurations for player 1 for:*

- **Existential** /**Positive P.** *Reachability:* $[\![\mathrm{E}(\Diamond R)]\!]$;
- **Almost Sure** *Reachability:* $[\![\mathrm{AS}(\Diamond R)]\!]$;
- **Sure**/**Almost sure** *Safety:* $[\![\mathrm{Sure}(\Box R)]\!]$;
- **Positive P.** *Safety:* $[\![\mathrm{NZ}(\Diamond R)]\!]$;

Strategy Classes, Updated

A strategy for player i is: $\sigma_i : (L \cdot M^*)^+ \to \mathrm{Dist}(\mathrm{Act})$

- **D**etermistic: only one action with probability 1;

$$\forall h \in (L \cdot M^*)^+, \exists \alpha : \sigma_i(h)[\alpha] = 1$$

Strategy Classes, Updated

A strategy for player i is: $\sigma_i : (L \cdot M^*)^+ \to \mathrm{Dist}(\mathrm{Act})$

- **D**etermistic: only one action with probability 1;

$$\forall h \in (L \cdot M^*)^+, \exists \alpha : \sigma_i(h)[\alpha] = 1$$

- **P**ositional: depends only on the current state;

$$\forall h \in (L \cdot M^*)^*, \forall s \in L \cdot M^*, \sigma_i(h \cdot s) = \sigma_i(s)$$

Strategy Classes, Updated

A strategy for player i is: $\sigma_i : (L \cdot M^*)^+ \to \mathrm{Dist}(\mathrm{Act})$

- **D**etermistic: only one action with probability 1;

$$\forall h \in (L \cdot M^*)^+, \exists \alpha : \sigma_i(h)[\alpha] = 1$$

- **P**ositional: depends only on the current state;

$$\forall h \in (L \cdot M^*)^*, \forall s \in L \cdot M^*, \sigma_i(h \cdot s) = \sigma_i(s)$$

- **F**inite **M**emory: the distribution of actions can be computed by a finite automaton.

$$\forall \delta \in \mathrm{Dist}(\mathrm{Act}), \{h \in (L \cdot M^*)^+ \mid \sigma_i(h) = \delta\} \text{ is a regular set}$$

and the set of possible distributions is **finite**.

Strategy Classes, Updated

A strategy for player i is: $\sigma_i : (L \cdot M^*)^+ \to \mathrm{Dist}(\mathrm{Act})$

- **D**etermistic: only one action with probability 1;

$$\forall h \in (L \cdot M^*)^+, \exists \alpha : \sigma_i(h)[\alpha] = 1$$

- **P**ositional: depends only on the current state;

$$\forall h \in (L \cdot M^*)^*, \forall s \in L \cdot M^*, \sigma_i(h \cdot s) = \sigma_i(s)$$

- **F**inite **M**emory: the distribution of actions can be computed by a finite automaton.

$$\forall \delta \in \mathrm{Dist}(\mathrm{Act}), \{h \in (L \cdot M^*)^+ \mid \sigma_i(h) = \delta\} \text{ is a regular set}$$

  and the set of possible distributions is **finite**.

P strategies **may** not be finitely represented. **PFM** are finitely represented, **Counting** too.

Conjunction of objectives

Theorem

Let Φ be a conjunction of NZ and AS objectives for safety and reachability path specifications. Then the winning region $[\![\Phi]\!]_i$ is computable.

Conjunction of objectives

Theorem

Let Φ be a conjunction of NZ and AS objectives for safety and reachability path specifications. Then the winning region $[\![\Phi]\!]_i$ is computable.

NB: no hope for anything more:

-  Mayr [May03] proved that $[\![\mathrm{E}(\Box R)]\!]$ cannot be computed.
-  Bertrand and al [BBS07] proves that $\mathrm{NZ}(\Box\Diamond R)$ (Büchi) and $\mathrm{AS}(\Box\Diamond R_1 \wedge \Diamond\Box R_2)$ cases are undecidable.

Conclusion and future work

- Infinite state game with stochastic and concurrent behaviours;
- Qualitative Objectives;
- Subword ordering for **termination** and **correctness**

Conclusion and future work

- Infinite state game with stochastic and concurrent behaviours;
- Qualitative Objectives;
- Subword ordering for **termination** and **correctness**

Future work:

- Restrictions on the strategy classes (FM only?);
- Support more combination of objectives to deal with Nash Equilibria;
- Partial observation (Signals).

Thank you for your attention

And have a nice trip back home :)

Bibliography I

[AABN18] Parosh Aziz Abdulla, Mohamed Faouzi Atig, Ahmed Bouajjani, and Tuan Phong Ngo. A load-buffer semantics for total store ordering. Logical Methods in Computer Science, 14(1), 2018.

[BBS06] Christel Baier, Nathalie Bertrand, and Philippe Schnoebelen. A note on the attractor-property of infinite-state markov chains. Inf. Process. Lett., 97(2):58–63, 2006.

[BBS07] Christel Baier, Nathalie Bertrand, and Philippe Schnoebelen. Verifying nondeterministic probabilistic channel systems against ω-regular linear-time properties. ACM Trans. Comput. Log., 9(1):5, 2007.

[dAHK07] Luca de Alfaro, Thomas A. Henzinger, and Orna Kupferman. Concurrent reachability games. Theor. Comput. Sci., 386(3):188–217, 2007.

[FS01] Alain Finkel and Philippe Schnoebelen. Well-structured transition systems everywhere! Theor. Comput. Sci., 256(1-2):63–92, 2001.

[May03] Richard Mayr. Undecidable problems in unreliable computations. Theor. Comput. Sci., 297(1-3):337–354, 2003.