

Randomized Strategies in Concurrent Games

Daniel Stan

Supervisors: Patricia Bouyer, Nicolas Markey



EqualIS



école —
normale —
supérieure —
paris—saclay —



université
PARIS-SACLAY

Formal methods



- Formal models of complex systems;

Formal methods



- Formal models of complex systems;
- To check **safety** properties;

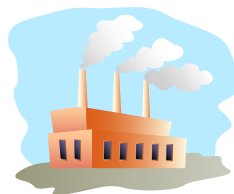
Formal methods



✓ X



X



✓✓

- Formal models of complex systems;
- To check **safety** properties;
- To **optimize** quantitative properties.

Formal methods



✓ X



X

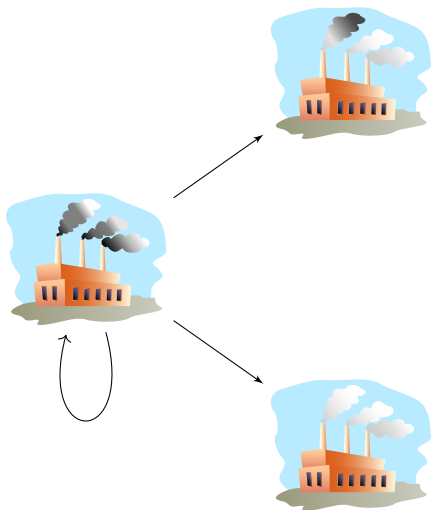


✓✓

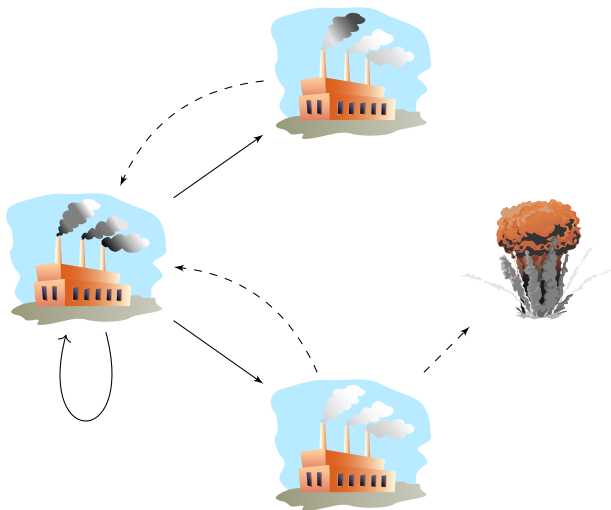
- Formal models of complex systems;
- To check **safety** properties;
- To **optimize** quantitative properties.

→ **In this thesis:** fundamental work, insights on these models.

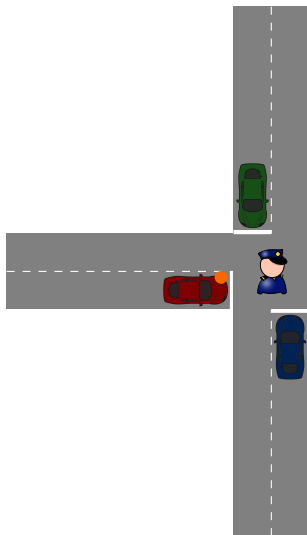
Turn-based games



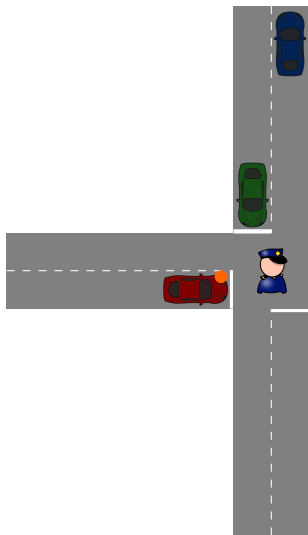
Turn-based games



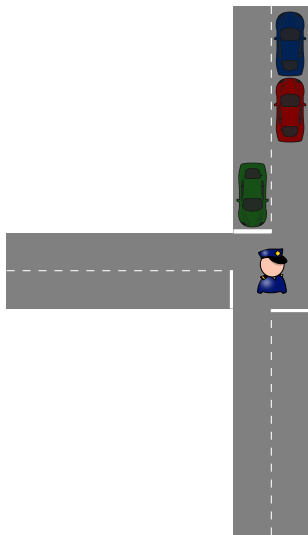
Multi-agent systems



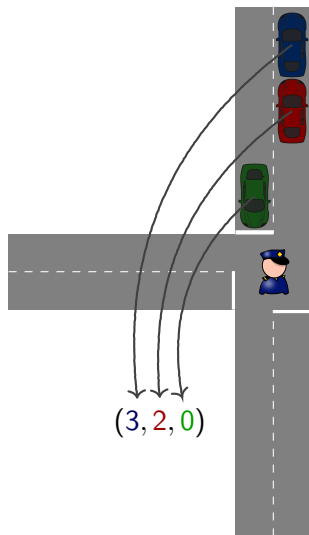
Multi-agent systems



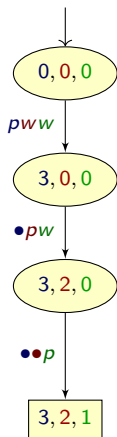
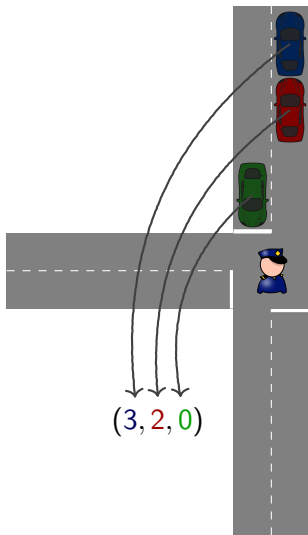
Multi-agent systems



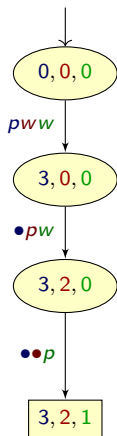
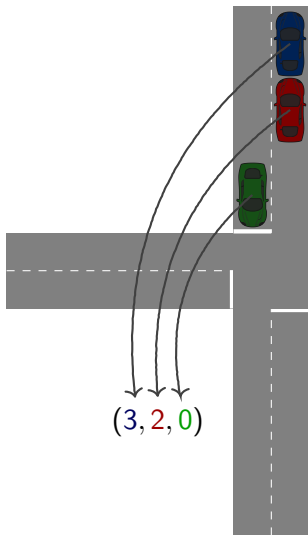
Multi-agent systems



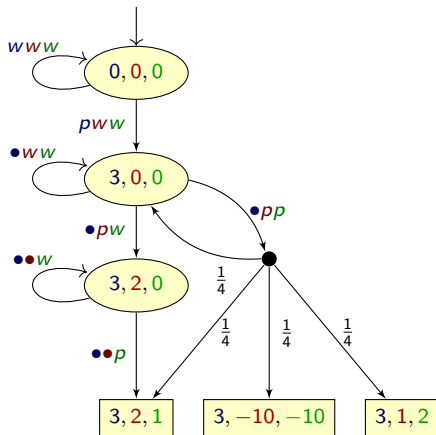
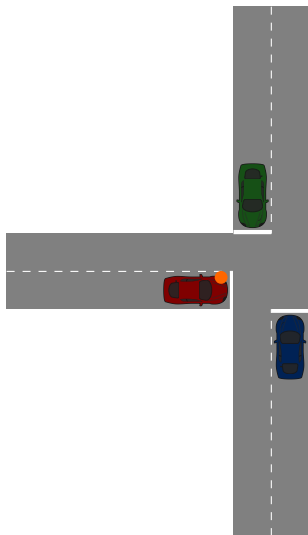
Multi-agent systems



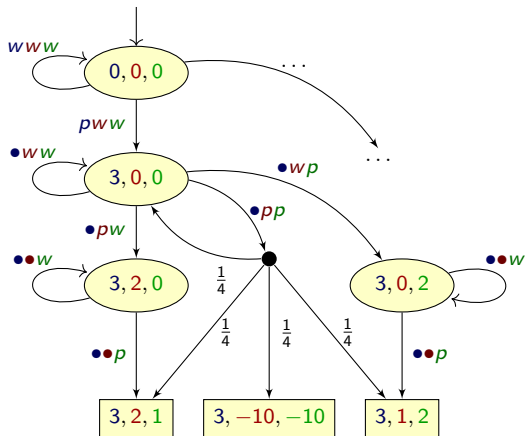
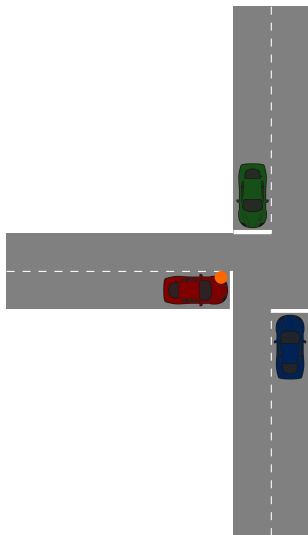
Multi-agent systems



Multi-agent systems



Multi-agent systems



Concurrent games

- A finite graph, played **concurrently**
- by **agents**:
 $\text{Agt} = \{1, 2, \dots\};$
- Each agent i has a **strategy**:
 $\sigma_i : \text{States}^+ \rightarrow \text{Dist}(\text{Act})$
- and an **objective**:
 $\Phi_i : \text{States}^\omega \rightarrow \mathbb{R}$

Concurrent games

- A finite graph, played **concurrently**
- by **agents**:

$$\text{Agt} = \{1, 2, \dots\};$$
- Each agent i has a **strategy**:

$$\sigma_i : \text{States}^+ \rightarrow \text{Dist}(\text{Act})$$
- and an **objective**:

$$\Phi_i : \text{States}^\omega \rightarrow \mathbb{R}$$
- Φ is a **terminal-reward** objective if it depends only on the final visited state:

$$\longrightarrow \boxed{3, -10, -10} \quad , (0, \dots, 0) \text{ otherwise}$$

Concurrent games

- A finite graph, played **concurrently**
- by **agents**:

$$\text{Agt} = \{1, 2, \dots\};$$
- Each agent i has a **strategy**:

$$\sigma_i : \text{States}^+ \rightarrow \text{Dist}(\text{Act})$$
- and an **objective**:

$$\Phi_i : \text{States}^\omega \rightarrow \mathbb{R}$$
- Φ is a **terminal-reward** objective if it depends only on the final visited state:

$$\rightarrow \boxed{3, -10, -10} \quad , (0, \dots, 0) \text{ otherwise}$$
- **Terminal-reachability** if 0 and 1 values.
- Opposed to **safety** objectives ($((0, \dots, 0)$ whenever a final state is reached).

How to break symmetries

How to break symmetries




How to break symmetries




Introduce a **global scheduler**.




How to break symmetries


 Introduce a **global scheduler**.

 **Randomize** the decisions.



How to break symmetries

 Introduce a **global scheduler**.

 **Randomize** the decisions.

Here:

- Agents are devices, computer, processes;
- Communication **collisions**, cooperation for common objectives.



Outline

Two forms of uncertainties:

 Non-determinism: “Does there **exist** a correct scheduler?”

 Stochasticity: “What is the **probability** mass of correct runs?”

Outline

Two forms of uncertainties:

 Non-determinism: “Does there **exist** a correct scheduler?”

 Stochasticity: “What is the **probability** mass of correct runs?”

~> What is the contribution of stochasticity to the system?

Outline

Two forms of uncertainties:

 Non-determinism: “Does there **exist** a correct scheduler?”

 Stochasticity: “What is the **probability** mass of correct runs?”

~> What is the contribution of stochasticity to the system?

Simple objectives: **terminal reachability** and **safety**.

Outline

Two forms of uncertainties:

 Non-determinism: “Does there **exist** a correct scheduler?”

 Stochasticity: “What is the **probability** mass of correct runs?”

~> What is the contribution of stochasticity to the system?

Simple objectives: **terminal reachability** and **safety**.

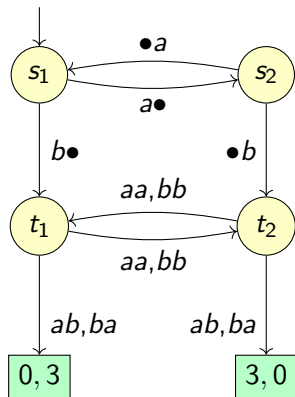
~> Main contributions to:

- Concurrent games on graphs: **fixed** number of agents;
- Parametrized verification of protocols: **arbitrary** number of processes.

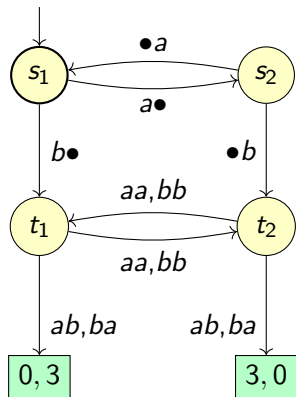
Summary

- 1 Introduction
- 2 Concurrent Games**
- 3 Parametrized Protocols
- 4 Conclusions

Concurrent games



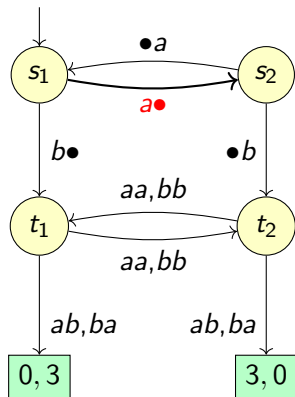
Concurrent games



Strategies:

Observation $h \mid \sigma_1(h) \mid \sigma_2(h)$

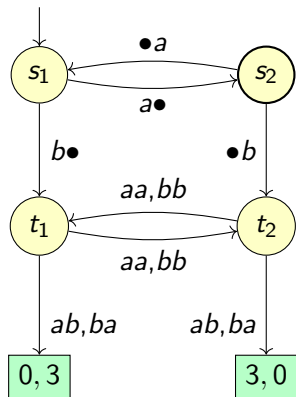
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet

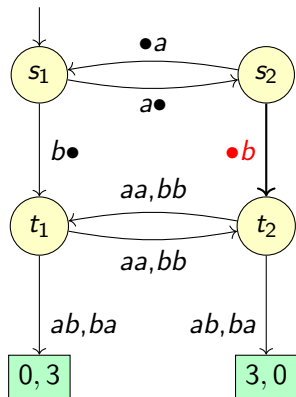
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet

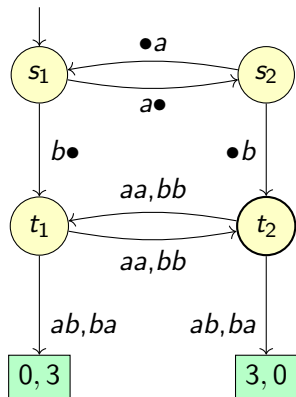
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet
$s_1 s_2$	\bullet	b

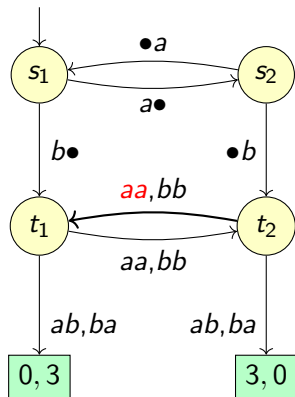
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet
$s_1 s_2$	\bullet	b

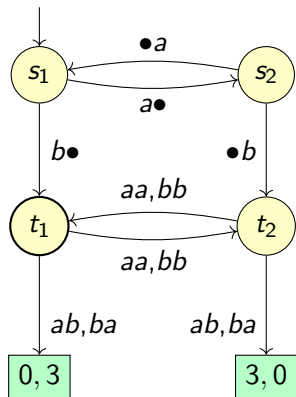
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet
$s_1 s_2$	\bullet	b
$s_1 s_2 t_2$	a	a

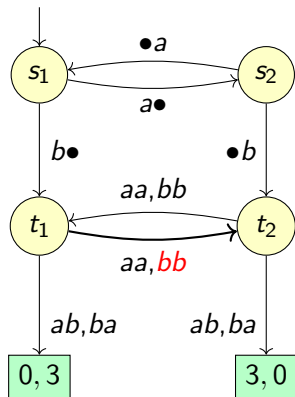
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet
$s_1 s_2$	\bullet	b
$s_1 s_2 t_2$	a	a

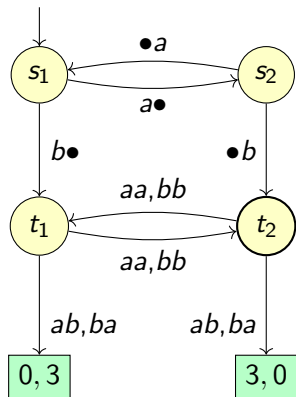
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet
$s_1 s_2$	\bullet	b
$s_1 s_2 t_2$	a	a
$s_1 s_2 t_2 t_1$	b	b

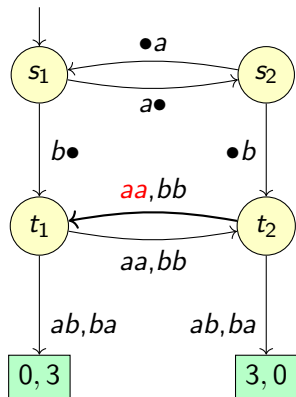
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet
$s_1 s_2$	\bullet	b
$s_1 s_2 t_2$	a	a
$s_1 s_2 t_2 t_1$	b	b

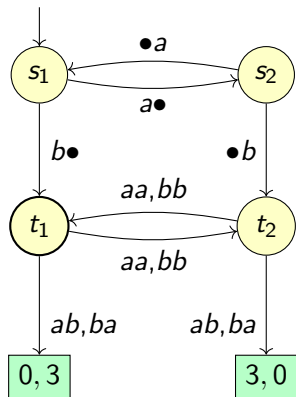
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet
$s_1 s_2$	\bullet	b
$s_1 s_2 t_2$	a	a
$s_1 s_2 t_2 t_1$	b	b
$s_1 s_2 t_2 t_1 t_2$	a	a

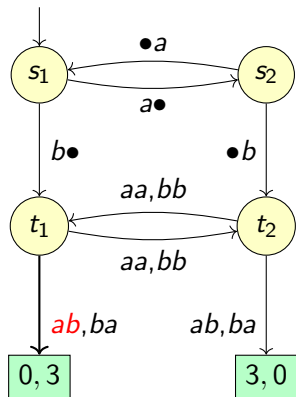
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet
$s_1 s_2$	\bullet	b
$s_1 s_2 t_2$	a	a
$s_1 s_2 t_2 t_1$	b	b
$s_1 s_2 t_2 t_1 t_2$	a	a

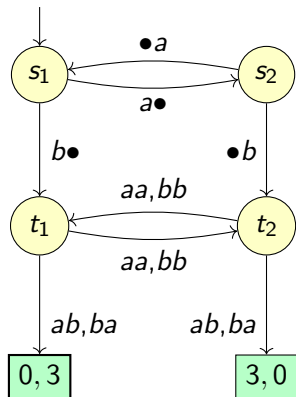
Concurrent games



Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet
$s_1 s_2$	\bullet	b
$s_1 s_2 t_2$	a	a
$s_1 s_2 t_2 t_1$	b	b
$s_1 s_2 t_2 t_1 t_2$	a	a
$s_1 s_2 t_2 t_1 t_2 t_1$	a	b

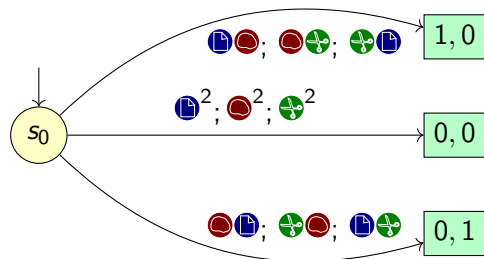
Concurrent games



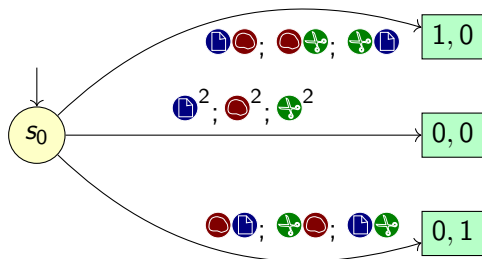
Strategies:

Observation h	$\sigma_1(h)$	$\sigma_2(h)$
s_1	a	\bullet
$s_1 s_2$	\bullet	b
$s_1 s_2 t_2$	a	a
$s_1 s_2 t_2 t_1$	b	b
$s_1 s_2 t_2 t_1 t_2$	a	a
$s_1 s_2 t_2 t_1 t_2 t_1$	a	b

Playing randomly: quantitative analysis

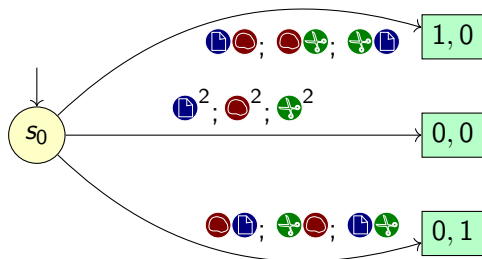


Playing randomly: quantitative analysis

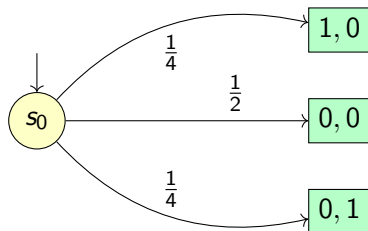


h	$\sigma_1(h)$	$\sigma_2(h)$
s_0	$\frac{1}{2} \text{ (blue document)} + \frac{1}{2} \text{ (red circle)}$	$\frac{1}{2} \text{ (blue document)} + \frac{1}{2} \text{ (red circle)}$

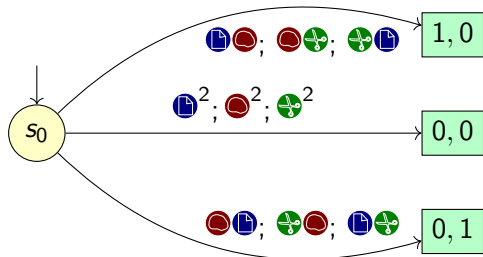
Playing randomly: quantitative analysis



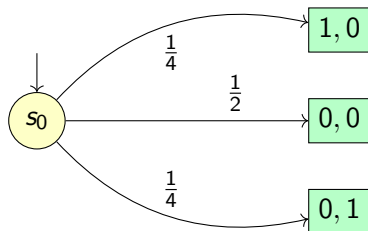
h	$\sigma_1(h)$	$\sigma_2(h)$
s_0	$\frac{1}{2}$ + $\frac{1}{2}$	$\frac{1}{2}$ + $\frac{1}{2}$



Playing randomly: quantitative analysis

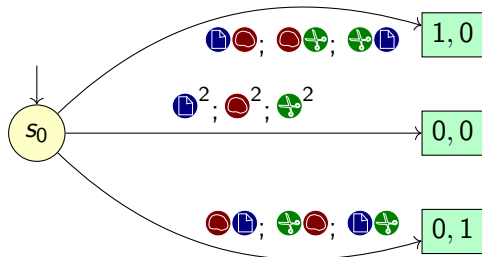


h	$\sigma_1(h)$	$\sigma_2(h)$
s_0	$\frac{1}{2}$ + $\frac{1}{2}$	$\frac{1}{2}$ + $\frac{1}{2}$

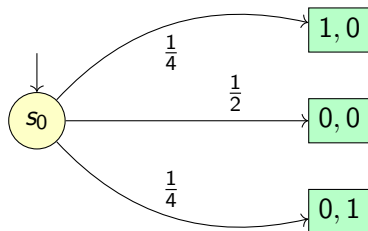


- Expected reward:** $\mathbb{E}^\sigma(\Phi) = \frac{1}{4}(1, 0) + \frac{1}{2}(0, 0) + \frac{1}{4}(0, 1) = (\frac{1}{4}, \frac{1}{4})$.

Playing randomly: quantitative analysis

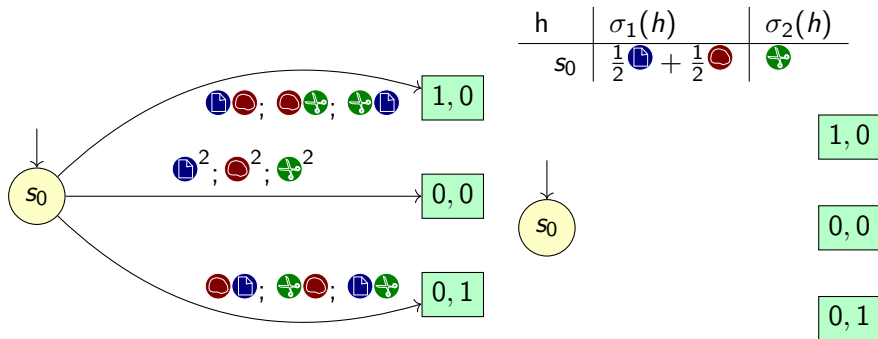


h	$\sigma_1(h)$	$\sigma_2(h)$
s_0	$\frac{1}{2}$ + $\frac{1}{2}$	$\frac{1}{2}$ + $\frac{1}{2}$



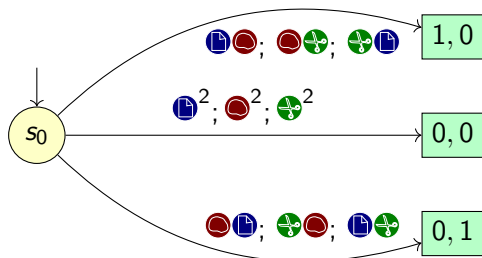
- **Expected reward:** $\mathbb{E}^\sigma(\Phi) = \frac{1}{4}(1, 0) + \frac{1}{2}(0, 0) + \frac{1}{4}(0, 1) = (\frac{1}{4}, \frac{1}{4})$.
- Player i wants to **maximize** $\mathbb{E}^\sigma(\Phi_i)$

Playing randomly: quantitative analysis

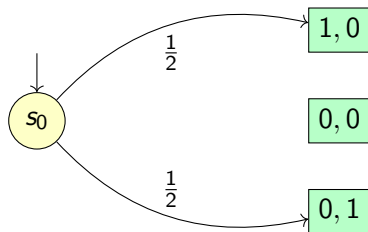


- Expected reward:
- Player i wants to **maximize** $\mathbb{E}^\sigma(\Phi_i)$

Playing randomly: quantitative analysis

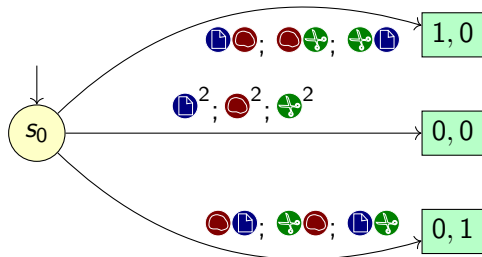





h	$\sigma_1(h)$	$\sigma_2(h)$
s_0	$\frac{1}{2}$ + $\frac{1}{2}$	

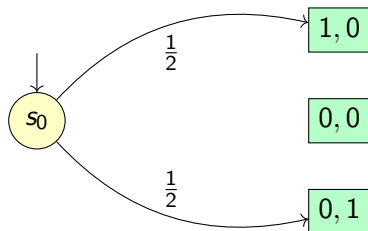


- **Expected reward:**
- Player i wants to **maximize** $\mathbb{E}^\sigma(\Phi_i)$

Playing randomly: quantitative analysis



h	$\sigma_1(h)$	$\sigma_2(h)$
s_0	$\frac{1}{2}$  + $\frac{1}{2}$ 	



- **Expected reward:** $\mathbb{E}^\sigma(\Phi) = (\frac{1}{2}, \frac{1}{2})$.
- Player i wants to **maximize** $\mathbb{E}^\sigma(\Phi_i)$: **a profitable deviation**.

Nash Equilibrium

Definition

σ is a Nash Equilibrium (NE) if for all agent i and any other strategy for i (deviation) σ'_i ,

$$\mathbb{E}^{\sigma[i/\sigma'_i]}(\phi_i) \leq \mathbb{E}^{\sigma}(\phi_i)$$

Nash Equilibrium

Definition

σ is a Nash Equilibrium (NE) if for all agent i and any other strategy for i (deviation) σ'_i ,

$$\mathbb{E}^{\sigma[i/\sigma'_i]}(\phi_i) \leq \mathbb{E}^{\sigma}(\phi_i)$$

No incentive to deviate to increase its own expected reward.

Nash Equilibrium

Definition

σ is a Nash Equilibrium (NE) if for all agent i and any other strategy for i (deviation) σ'_i ,

$$\mathbb{E}^{\sigma[i/\sigma'_i]}(\phi_i) \leq \mathbb{E}^{\sigma}(\phi_i)$$

No incentive to deviate to increase its own expected reward.

Example

s	$\sigma_1(s)$	$\sigma_2(s)$
s_0	$\frac{1}{3}(\text{blue square} + \text{red circle} + \text{green circle})$	

Nash Equilibrium

Definition

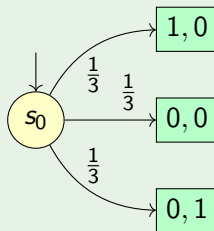
σ is a Nash Equilibrium (NE) if for all agent i and any other strategy for i (deviation) σ'_i ,

$$\mathbb{E}^{\sigma[i/\sigma'_i]}(\phi_i) \leq \mathbb{E}^{\sigma}(\phi_i)$$

No incentive to deviate to increase its own expected reward.

Example

s	$\sigma_1(s)$	$\sigma_2(s)$
s_0	$\frac{1}{3}(\text{blue square} + \text{red circle} + \text{green circle})$	



Nash Equilibrium

Definition

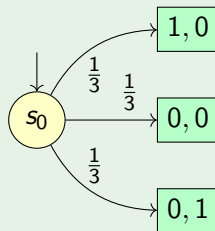
σ is a Nash Equilibrium (NE) if for all agent i and any other strategy for i (deviation) σ'_i ,

$$\mathbb{E}^{\sigma[i/\sigma'_i]}(\phi_i) \leq \mathbb{E}^{\sigma}(\phi_i)$$

No incentive to deviate to increase its own expected reward.

Example

s	$\sigma_1(s)$	$\sigma_2(s)$
s_0	$\frac{1}{3}(\text{blue square} + \text{red circle} + \text{green circle})$	$\frac{1}{3}(\text{blue square} + \text{red circle} + \text{green circle})$



Nash Equilibrium

Definition

σ is a Nash Equilibrium (NE) if for all agent i and any other strategy for i (deviation) σ'_i ,

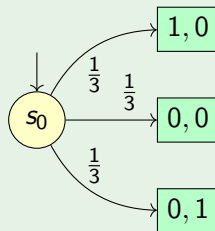
$$\mathbb{E}^{\sigma[i/\sigma'_i]}(\phi_i) \leq \mathbb{E}^{\sigma}(\phi_i)$$

No incentive to deviate to increase its own expected reward.

Example

s	$\sigma_1(s)$	$\sigma_2(s)$
s_0	$\frac{1}{3}(\text{blue square} + \text{red circle} + \text{green triangle})$	$\frac{1}{3}(\text{blue square} + \text{red circle} + \text{green triangle})$

Expected reward: $\mathbb{E}^{\sigma}(\Phi) = (\frac{1}{3}, \frac{1}{3})$.



Existence of NE in concurrent games

Problem

- Input: a game \mathcal{G} with terminal-reachability and safety objectives.
- Output: **YES** if there exists a NE σ for \mathcal{G} , **NO** otherwise.

Preliminary problem before considering the computation of strategies.

Existence of NE in concurrent games

	T. Reachability	Safety	Both
Existence			
Decidability			

Existence of NE in concurrent games

	T. Reachability	Safety	Both
Existence			✓ one-stage-game [Nas50]
Decidability			✓ one-stage-game

Existence of NE in concurrent games

	T. Reachability	Safety	Both
Existence			✓ one-stage-game [Nas50] ✗ deterministic games ¹
Decidability			✓ one-stage-game ✓ deterministic games [BBMU11]

¹and strategies

Existence of NE in concurrent games

	T. Reachability	Safety	Both
Existence			✓ one-stage-game [Nas50] ✗ deterministic games ¹ ✗ zero-sum games [KS81]
Decidability			✓ one-stage-game ✓ deterministic games [BBMU11] ✗ at least 11 players [UW11]

¹and strategies

Existence of NE in concurrent games

	T. Reachability	Safety	Both
Existence		✓ [SS01]	✓ one-stage-game [Nas50] ✗ deterministic games ¹ ✗ zero-sum games [KS81]
Decidability		✓	✓ one-stage-game ✓ deterministic games [BBMU11] ✗ at least 11 players [UW11]

¹and strategies

Existence of NE in concurrent games

	T. Reachability	Safety	Both
Existence	Unknown ✓ ϵ -NE [CJM04]	✓ [SS01]	✓ one-stage-game [Nas50] ✗ deterministic games ¹ ✗ zero-sum games [KS81]
Decidability	Unknown ✓ ϵ -NE	✓	✓ one-stage-game ✓ deterministic games [BBMU11] ✗ at least 11 players [UW11]

¹and strategies

Undecidability

Theorem (Bouyer, Markey, S. in *FSTTCS'2014*)

The existence problem is **undecidable** for three players.

Undecidability

Theorem (Bouyer, Markey, S. in *FSTTCS'2014*)

The existence problem is **undecidable** for three players.



Features:

- Simulation of a 2-counter machine [Min67].
- Constraint NE: first player has to win **almost-surely**.



Undecidability

Theorem (Bouyer, Markey, S. in *FSTTCS'2014*)

The existence problem is **undecidable** for three players.



Features:

- Simulation of a 2-counter machine [Min67].
- Constraint NE: first player has to win **almost-surely**.



- Heavy use of concurrent actions.
- Heavy use of action-invisibility.
- Limited number of players.
- We can restrict to qualitative objectives.

Summary of the study

	T. Reachability	Safety	Both
Existence	Unknown ✓ ϵ -NE [CJM04]	✓ [SS01]	✓ one-stage-game [Nas50] ✗ deterministic games ✗ zero-sum games [KS81]
Decidability	Unknown ✓ ϵ -NE	✓	✓ one-stage-game ✓ deterministic games [BBMU11] ✗ at least 11 players [UW11]

Summary of the study

	T. Reachability	Safety	Both
Existence	Unknown ✓ ϵ -NE [CJM04]	✓ [SS01]	✓ one-stage-game [Nas50] ✗ deterministic games ✗ zero-sum games [KS81]
Decidability	Unknown ✓ ϵ -NE	✓	✓ one-stage-game ✓ deterministic games [BBMU11] ✗ at least 11 players [UW11] ✗ at least 3 players (FSTTCS'2014)

Summary of the study

	T. Reachability	Safety	Both
Existence	Unknown ✓ ϵ -NE [CJM04] ✓ ϵ -precise dev. (GandALF'2016)	✓ [SS01]	✓ one-stage-game [Nas50] ✗ deterministic games ✗ zero-sum games [KS81]
Decidability	Unknown ✓ ϵ -NE ✓ ϵ -precise dev.	✓	✓ one-stage-game ✓ deterministic games [BBMU11] ✗ at least 11 players [UW11] ✗ at least 3 players (FSTTCS'2014)

Summary

- 1 Introduction
- 2 Concurrent Games
- 3 Parametrized Protocols**
- 4 Conclusions

Parametrized verification



- Concrete situations: most agents are **identical**;
- The number of agents is an unknown **parameter**;

Parametrized verification



Several choices:

- Concrete situations: most agents are **identical**;
- The number of agents is an unknown **parameter**;
- Idea: relax parameter assumptions to gain regularity;
- Obtain an answer for any parameter, asymptotically;

Parametrized verification



Several choices:

- Agent model: **Finite automaton**, pushdown automaton;

- Concrete situations: most agents are **identical**;
- The number of agents is an unknown **parameter**;
- Idea: relax parameter assumptions to gain regularity;
- Obtain an answer for any parameter, asymptotically;

Parametrized verification

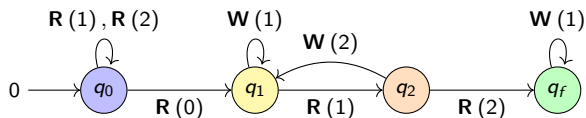


- Concrete situations: most agents are **identical**;
- The number of agents is an unknown **parameter**;
- Idea: relax parameter assumptions to gain regularity;
- Obtain an answer for any parameter, asymptotically;

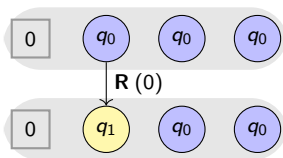
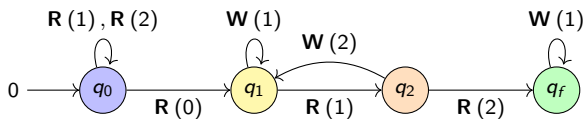
Several choices:

- Agent model: **Finite automaton**, pushdown automaton;
- Communication model [Esp14]: broadcast communication, shared register, *Rendez-Vous* (population protocols), **shared register with non-atomicity**.

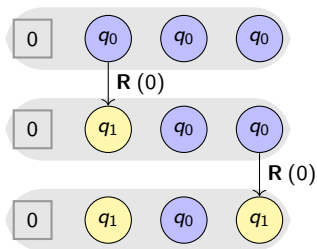
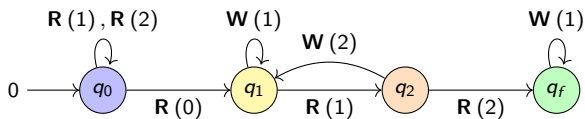
Example: read and write with 3 agents



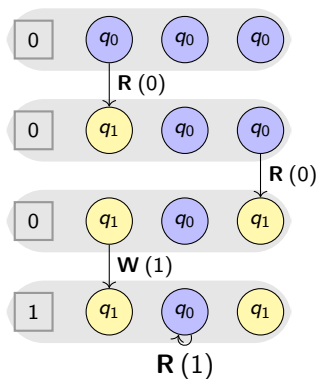
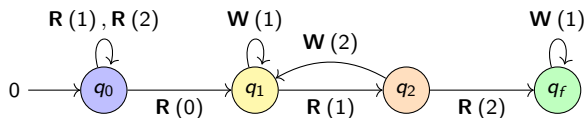
Example: read and write with 3 agents



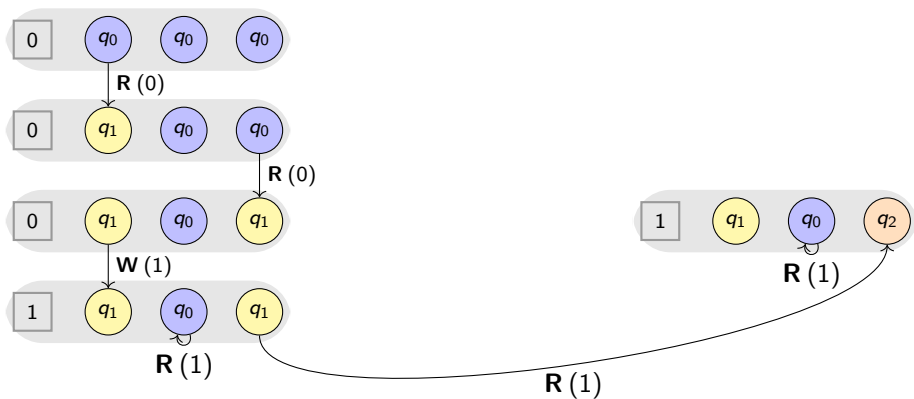
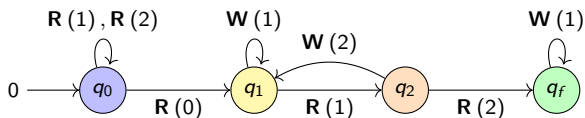
Example: read and write with 3 agents



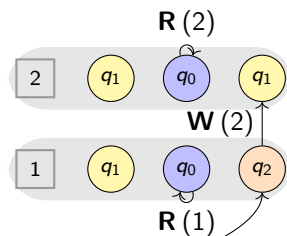
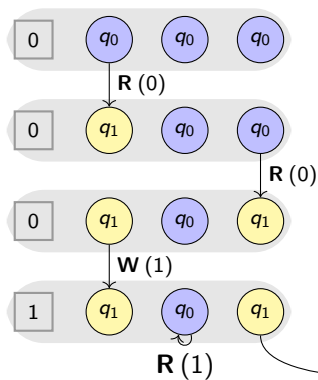
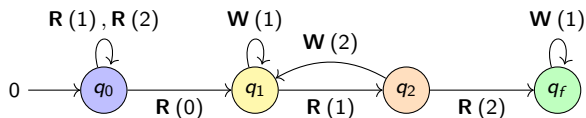
Example: read and write with 3 agents



Example: read and write with 3 agents

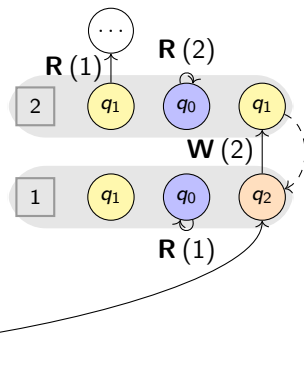
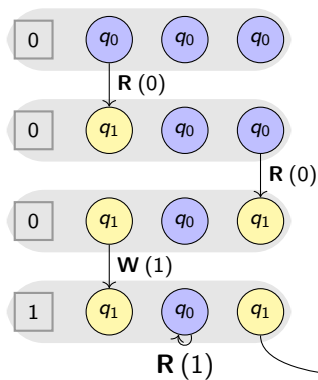
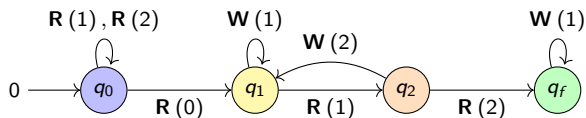


Example: read and write with 3 agents

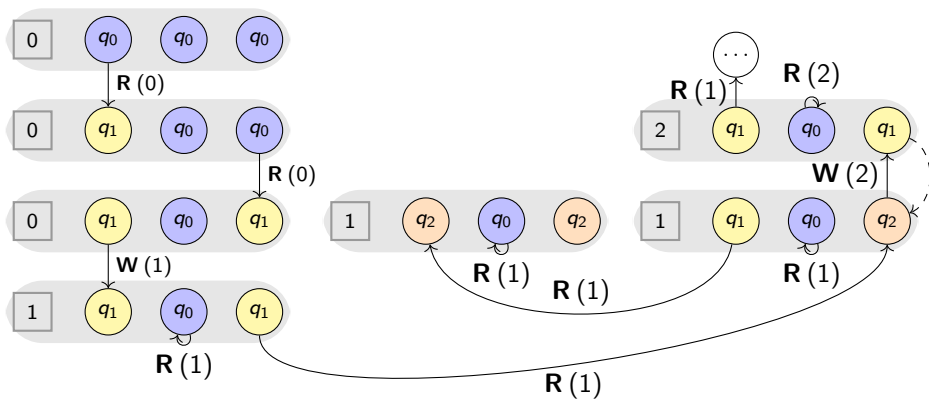
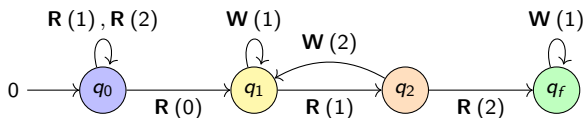


$R(1)$

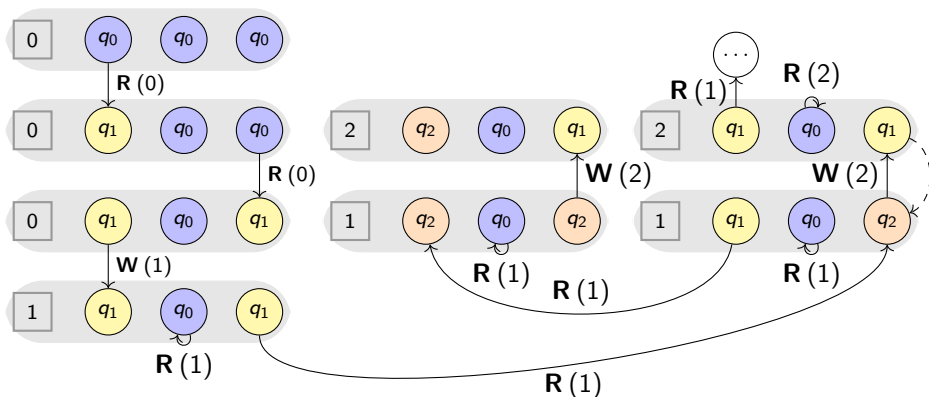
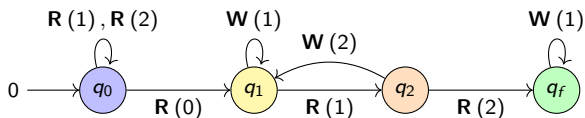
Example: read and write with 3 agents



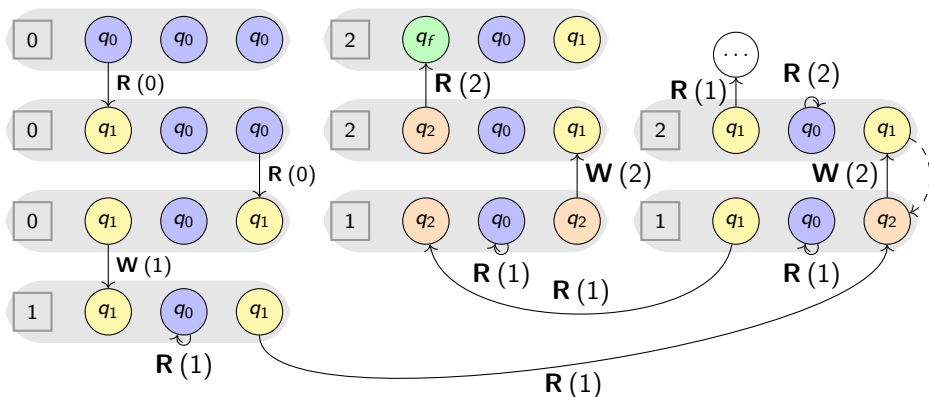
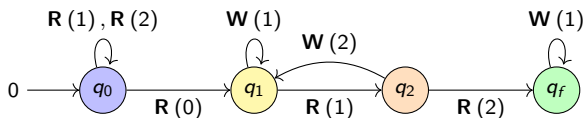
Example: read and write with 3 agents



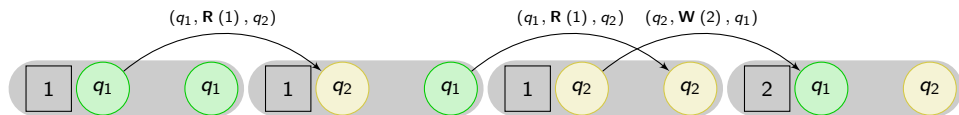
Example: read and write with 3 agents



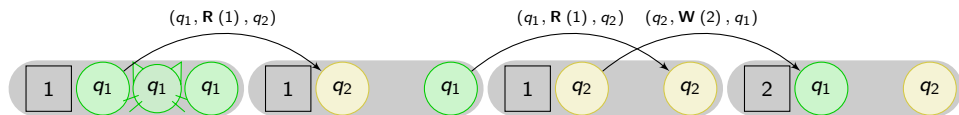
Example: read and write with 3 agents



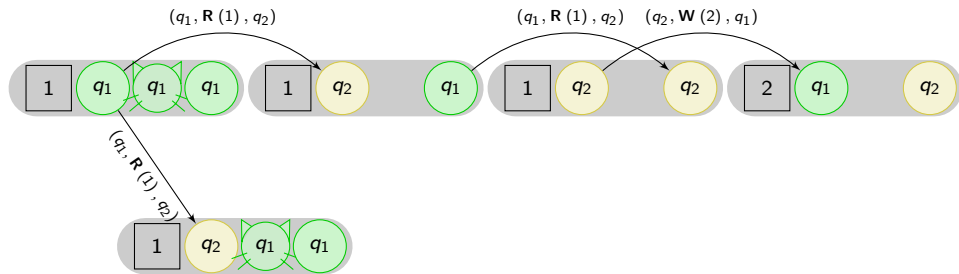
Deterministic case: copycat lemma [EGM13]



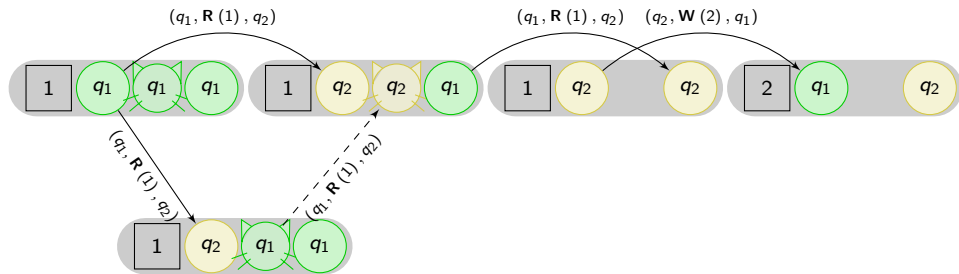
Deterministic case: copycat lemma [EGM13]



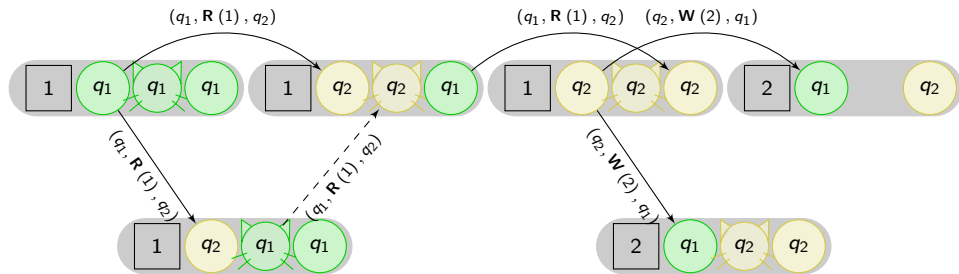
Deterministic case: copycat lemma [EGM13]



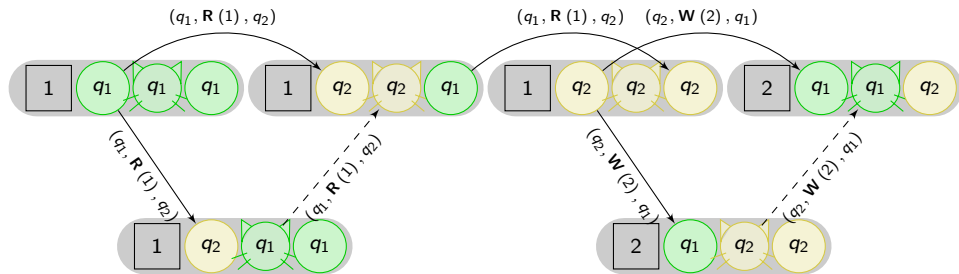
Deterministic case: copycat lemma [EGM13]



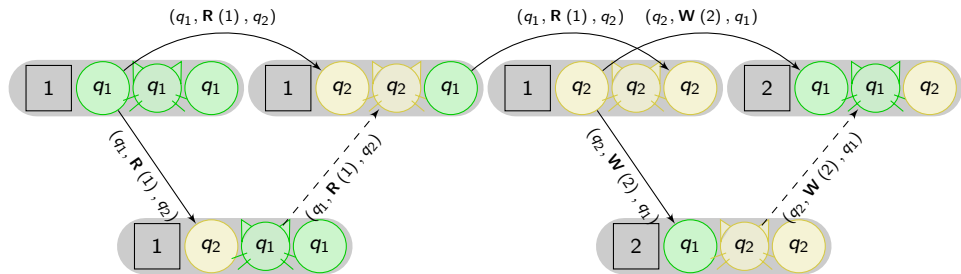
Deterministic case: copycat lemma [EGM13]



Deterministic case: copycat lemma [EGM13]



Deterministic case: copycat lemma [EGM13]



- **Coverability objective:** reach $\Diamond q_f$, a configuration with at least one process in q_f .
- If reachable with parameter n , it is **still reachable** with parameter $n + 1$.
- This property holds because of the **non-atomicity** hypothesis.

Role of the scheduler

Theorem ([EGM13, DEGM15])

*The **coverability** problem can be solved in polynomial time and becomes NP-complete in the presence of a distinguished **leader** process.*

Role of the scheduler

Theorem ([EGM13, DEGM15])

The **coverability** problem can be solved in polynomial time and becomes NP-complete in the presence of a distinguished **leader** process.

Scheduler : $\text{Configuration}^+ \rightarrow \text{Configuration}$

Role of the scheduler

Theorem ([EGM13, DEGM15])

The **coverability** problem can be solved in polynomial time and becomes NP-complete in the presence of a distinguished **leader** process.

Scheduler : $\text{Configuration}^+ \rightarrow \text{Configuration}$



The scheduler is deterministic, and has a **global** view over the system.

Role of the scheduler

Theorem ([EGM13, DEGM15])

The **coverability** problem can be solved in polynomial time and becomes NP-complete in the presence of a distinguished **leader** process.

Scheduler : $\text{Configuration}^+ \rightarrow \text{Configuration}$



The scheduler is deterministic, and has a **global** view over the system.

In this thesis:



What happens with a **stochastic** scheduler?

↪ Denote $\mathbb{P}_n(\diamond q_f)$ the probability to cover a state q_f .

Almost-sure reachability

Lemma (Qualitative assumption)

*The properties $\mathbb{P}_n(\Diamond q_f) > 0$ and $\mathbb{P}_n(\Diamond q_f) = 1$ **do not** depend on the actual probability distributions.*

Almost-sure reachability

Lemma (Qualitative assumption)

*The properties $\mathbb{P}_n(\Diamond q_f) > 0$ and $\mathbb{P}_n(\Diamond q_f) = 1$ **do not** depend on the actual probability distributions.*

We have already solved the case $\mathbb{P}_n(\Diamond q_f) > 0$: it corresponds to finding a path to $\Diamond q_f$.

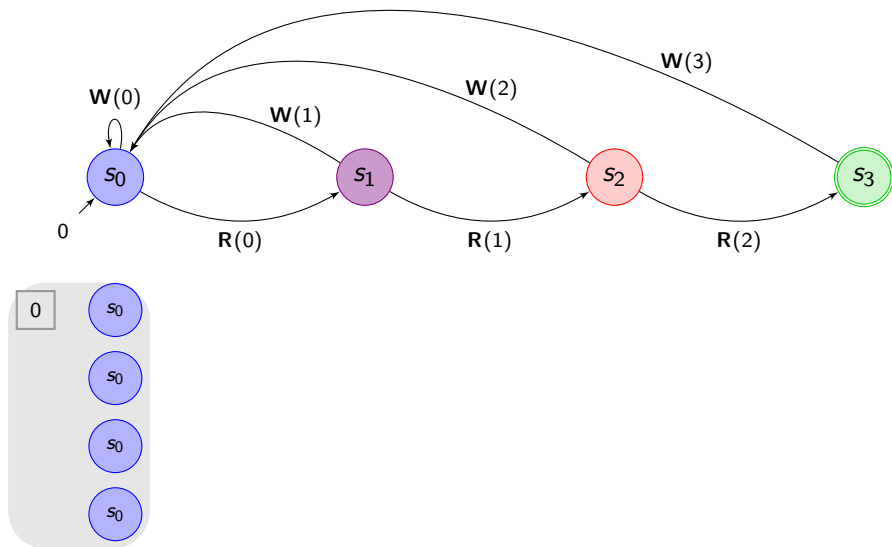
Almost-sure reachability

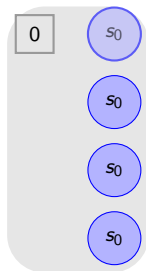
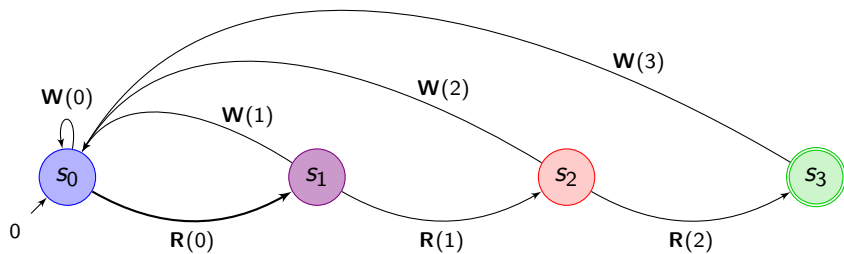
Lemma (Qualitative assumption)

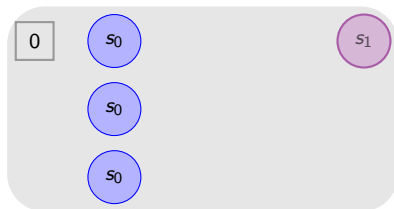
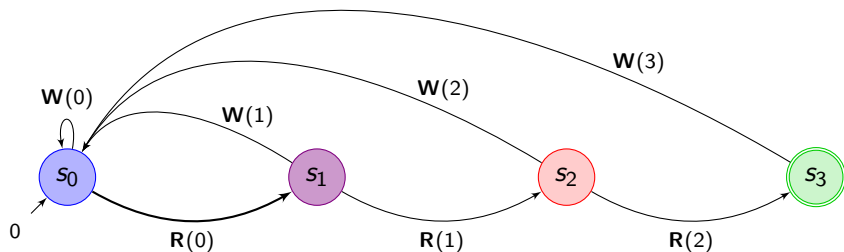
*The properties $\mathbb{P}_n(\Diamond q_f) > 0$ and $\mathbb{P}_n(\Diamond q_f) = 1$ **do not** depend on the actual probability distributions.*

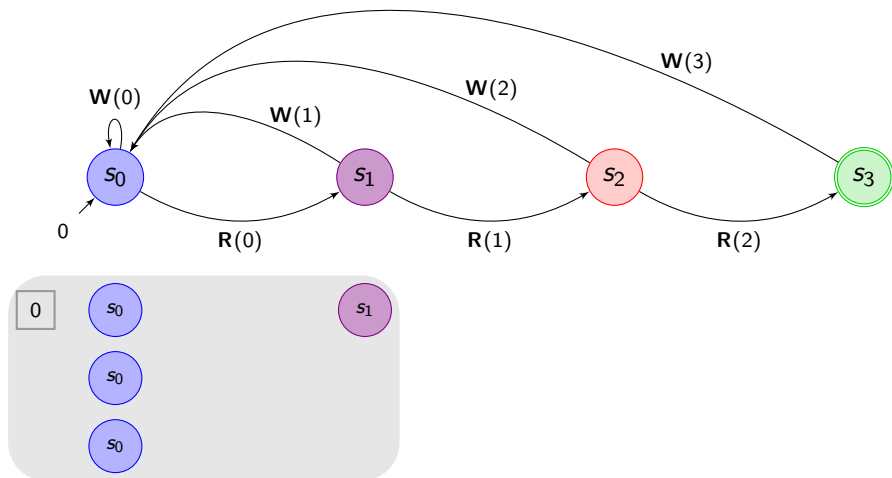
We have already solved the case $\mathbb{P}_n(\Diamond q_f) > 0$: it corresponds to finding a path to $\Diamond q_f$.

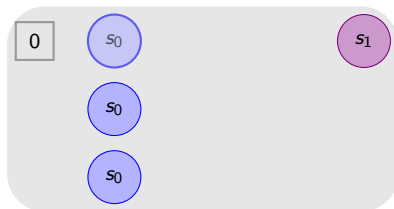
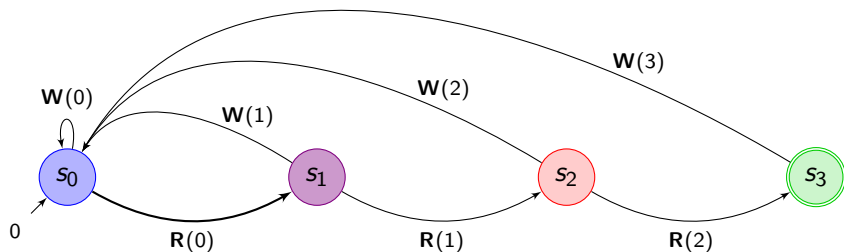
↪ We focus now on the **almost-sure** case: $\mathbb{P}_n(\Diamond q_f) = 1$.

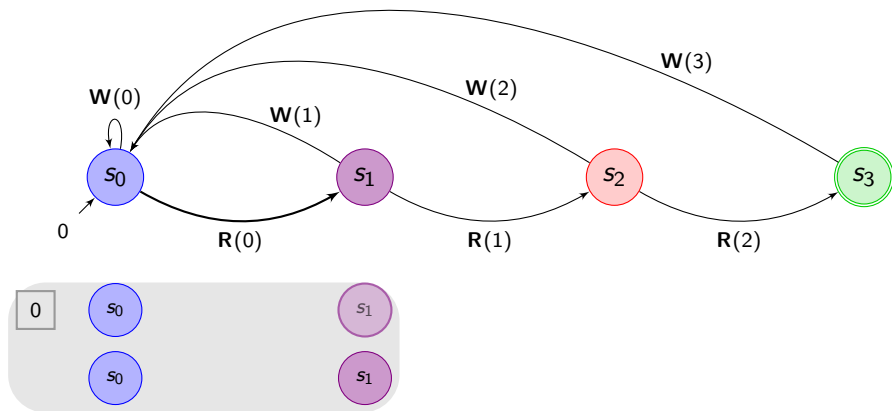
Filter protocol \mathcal{F}_4 

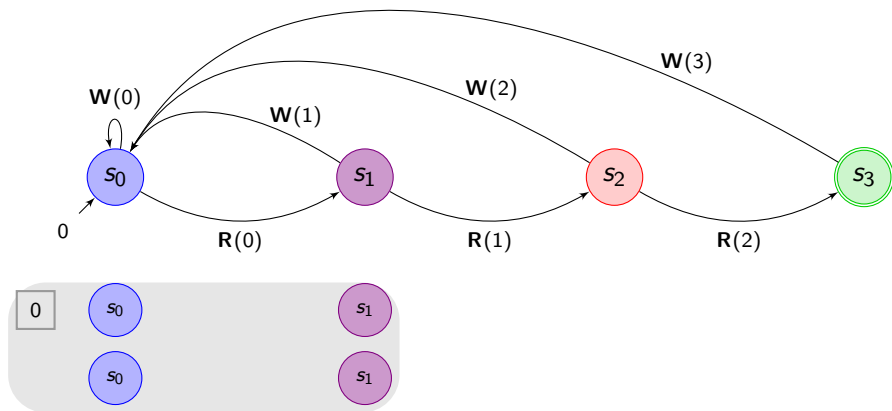
Filter protocol \mathcal{F}_4 

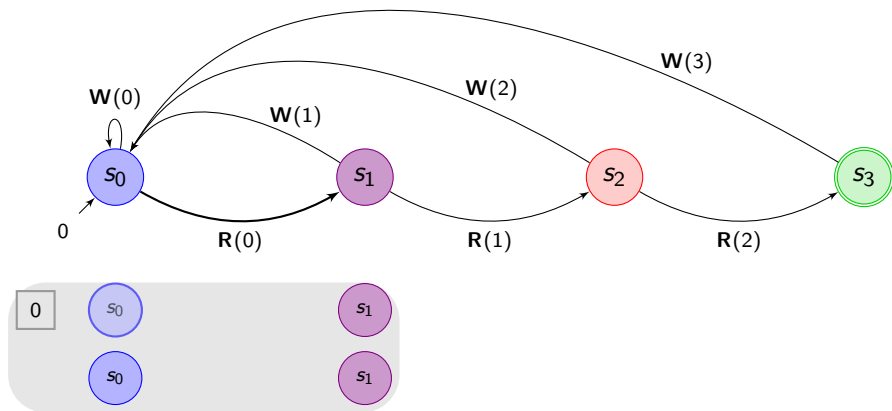
Filter protocol \mathcal{F}_4 

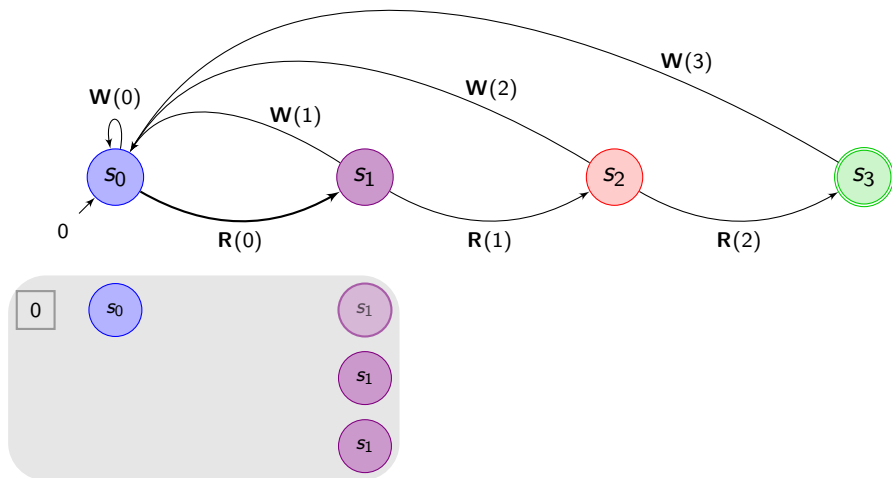
Filter protocol \mathcal{F}_4 

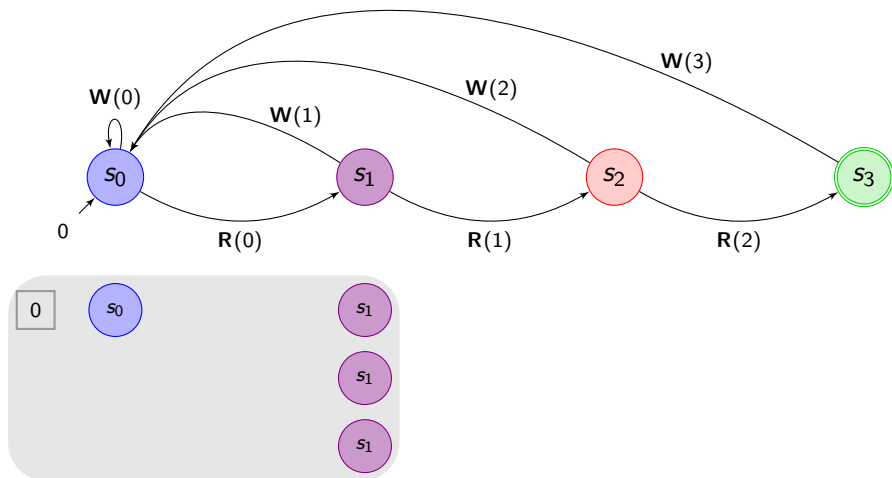
Filter protocol \mathcal{F}_4 

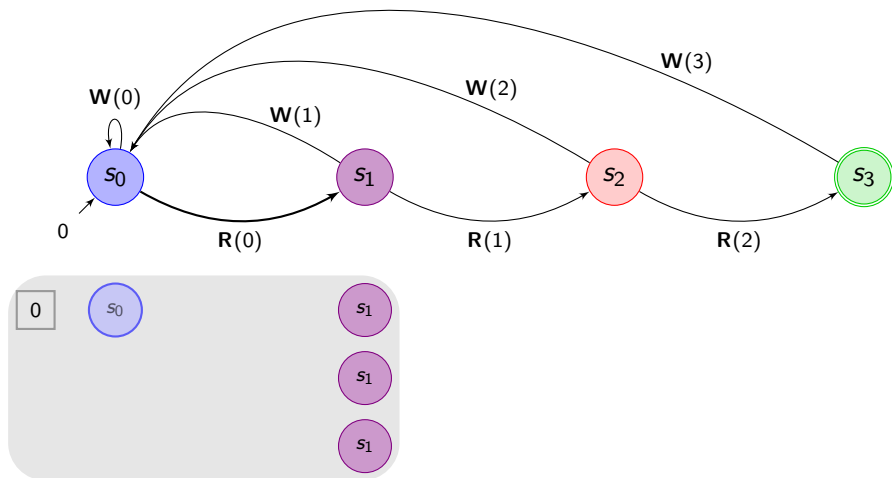
Filter protocol \mathcal{F}_4 

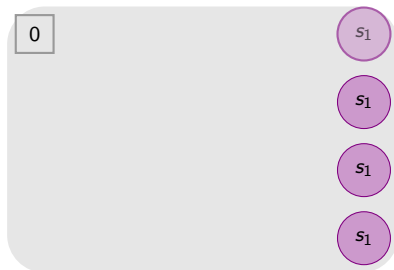
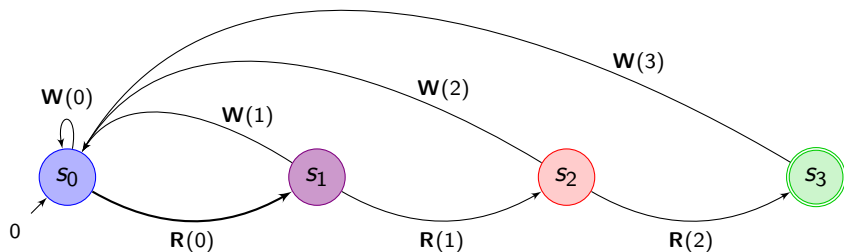
Filter protocol \mathcal{F}_4 

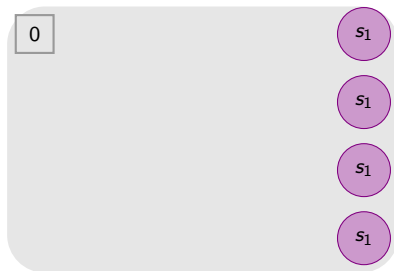
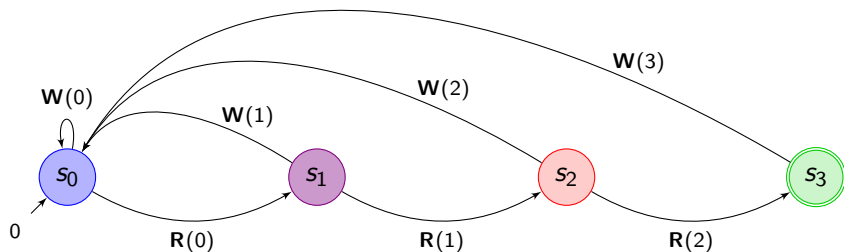
Filter protocol \mathcal{F}_4 

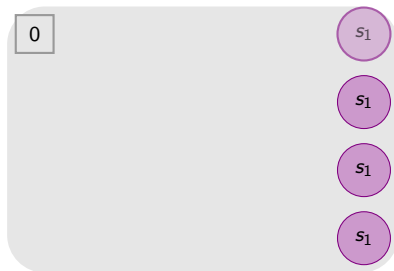
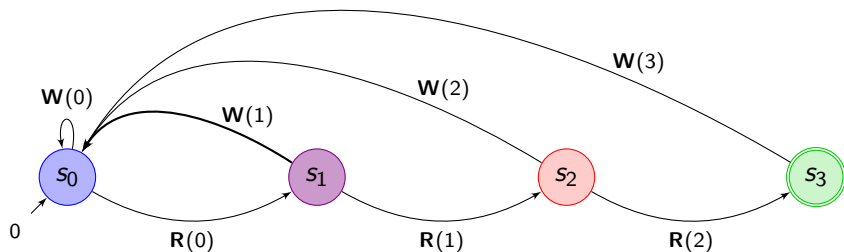
Filter protocol \mathcal{F}_4 

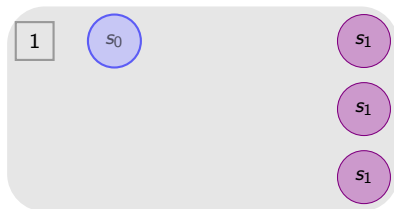
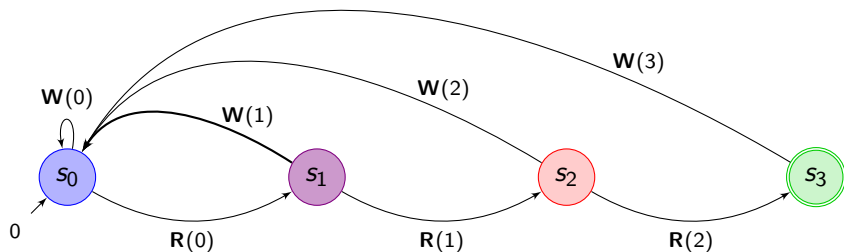
Filter protocol \mathcal{F}_4 

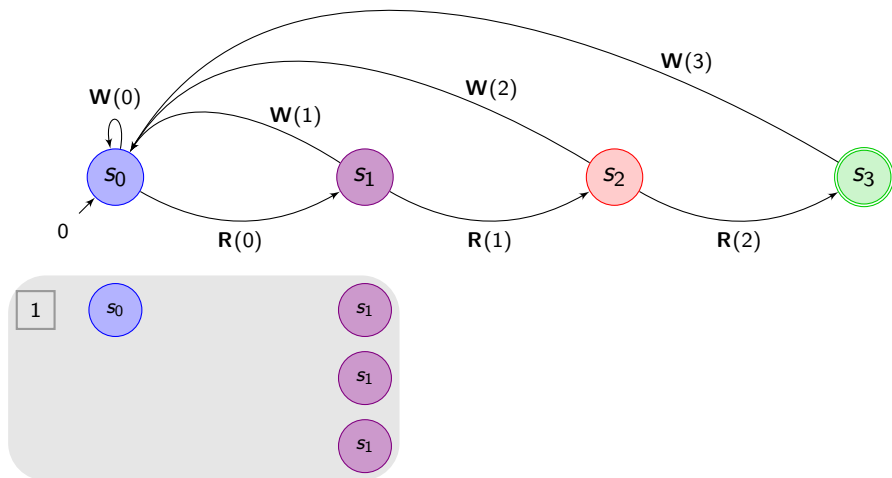
Filter protocol \mathcal{F}_4 

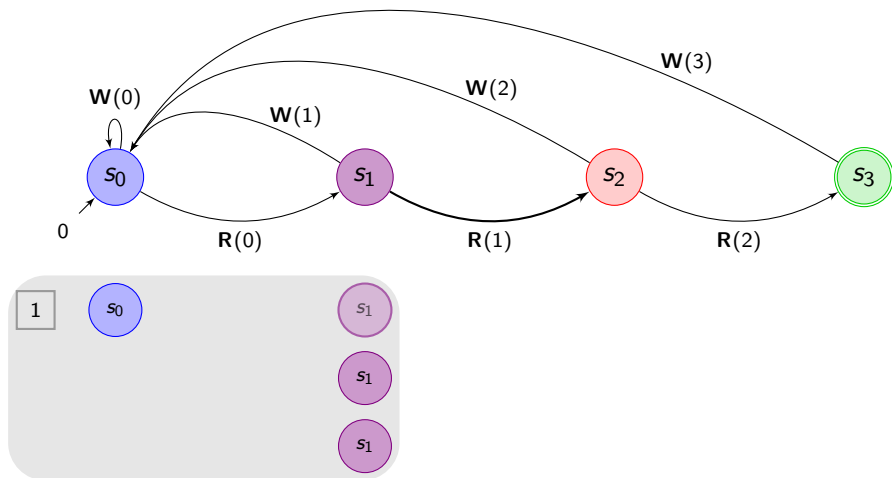
Filter protocol \mathcal{F}_4 

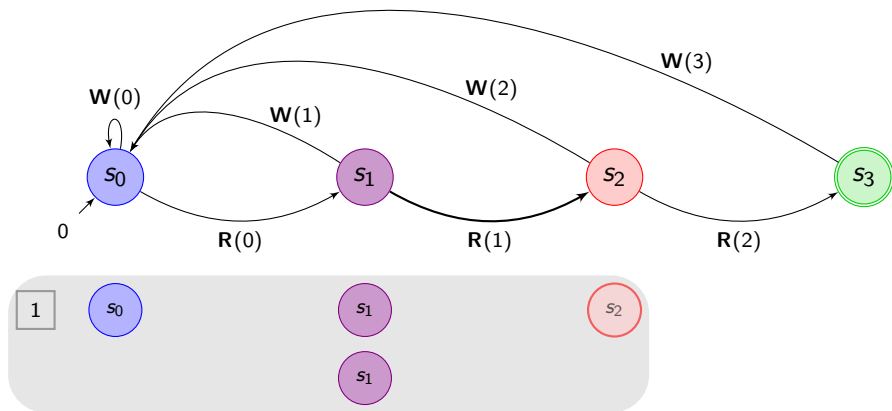
Filter protocol \mathcal{F}_4 

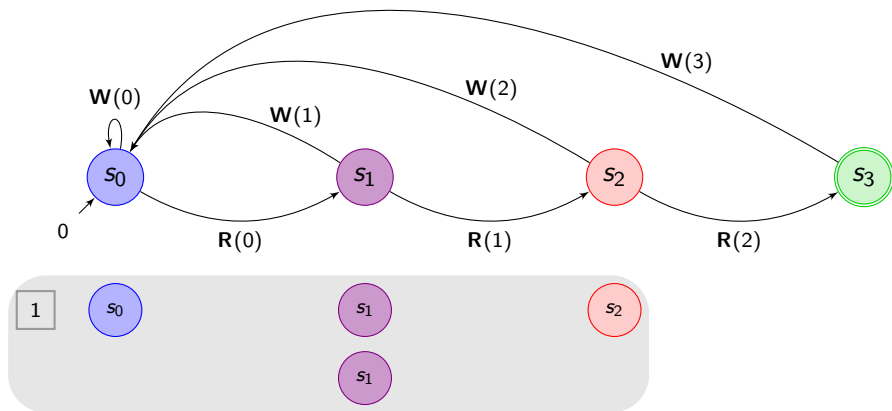
Filter protocol \mathcal{F}_4 

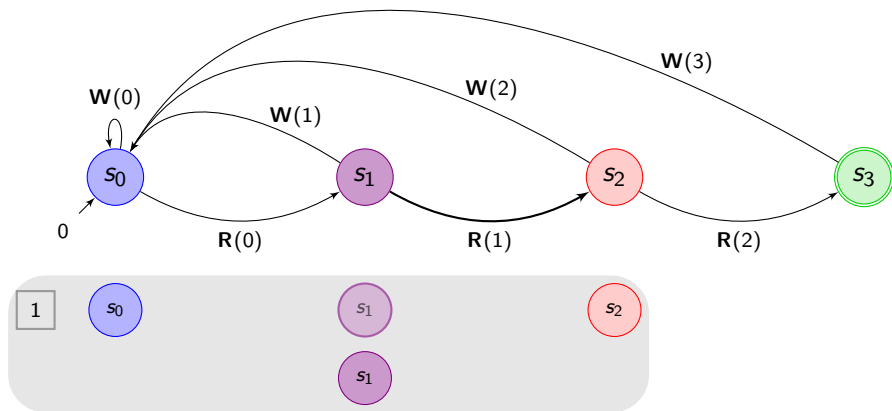
Filter protocol \mathcal{F}_4 

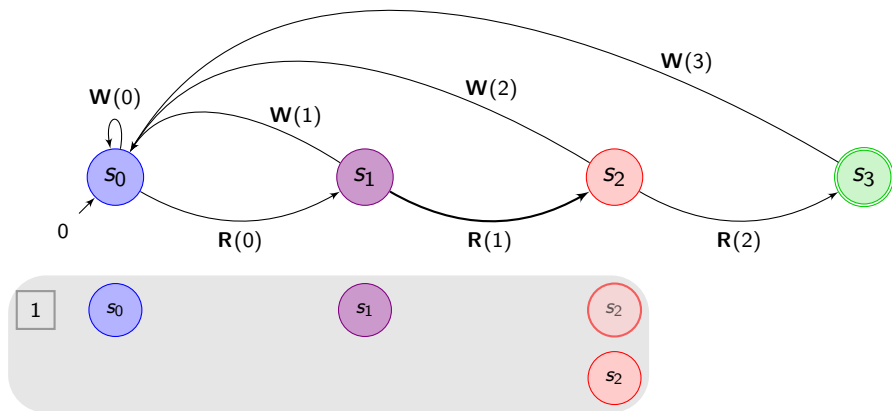
Filter protocol \mathcal{F}_4 

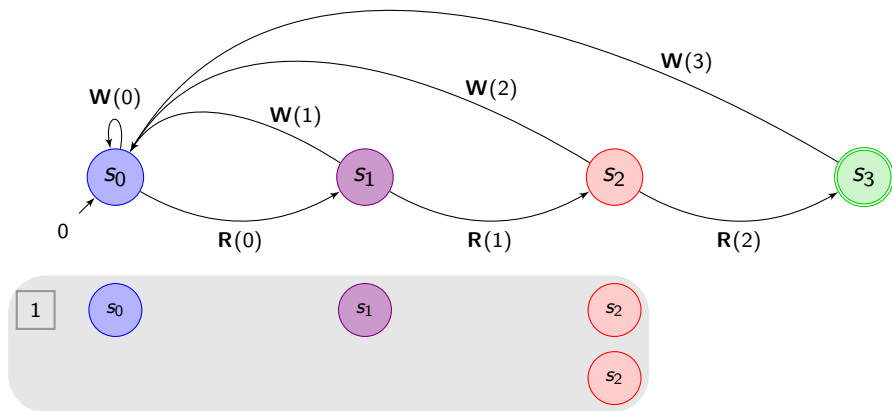
Filter protocol \mathcal{F}_4 

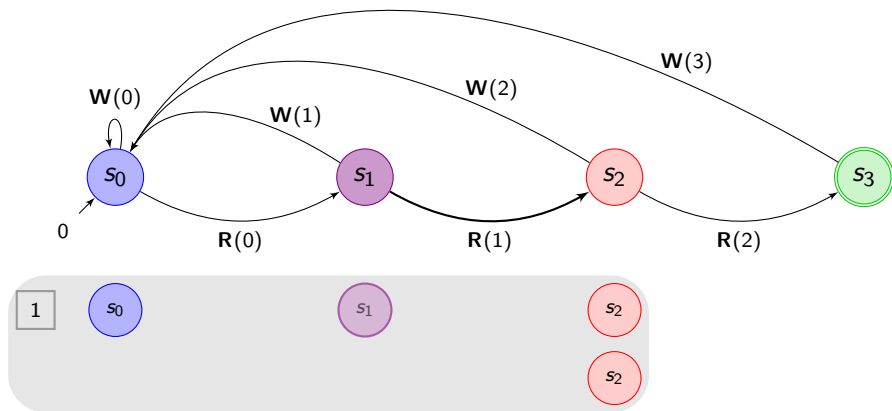
Filter protocol \mathcal{F}_4 

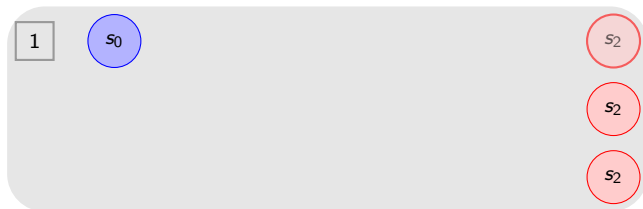
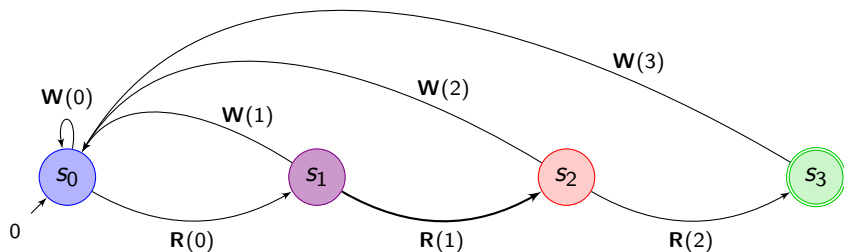
Filter protocol \mathcal{F}_4 

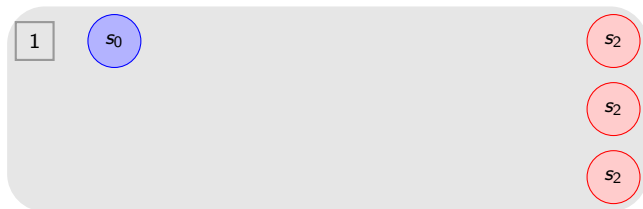
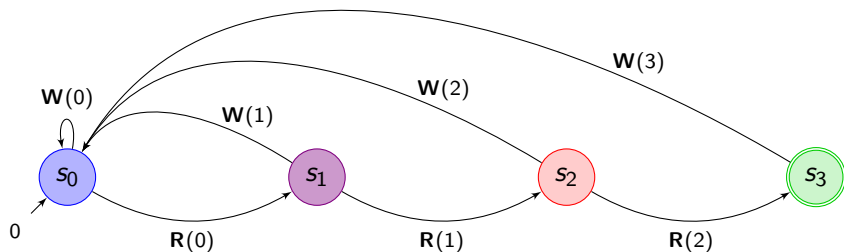
Filter protocol \mathcal{F}_4 

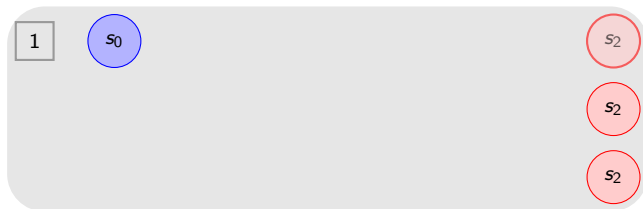
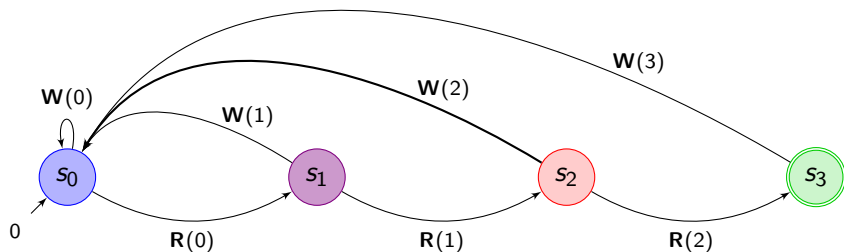
Filter protocol \mathcal{F}_4 

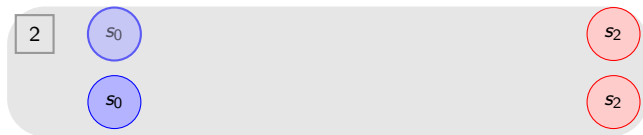
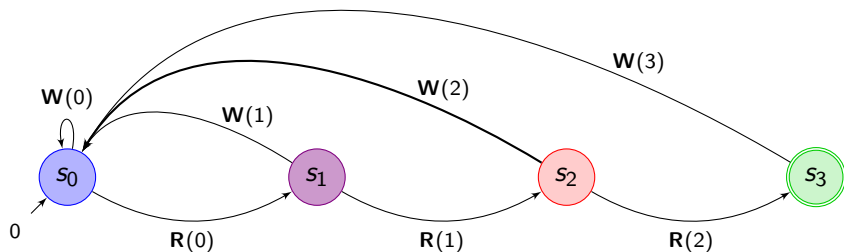
Filter protocol \mathcal{F}_4 

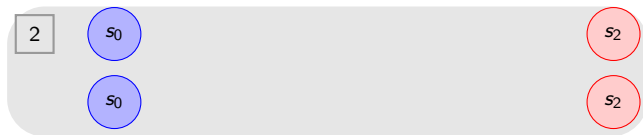
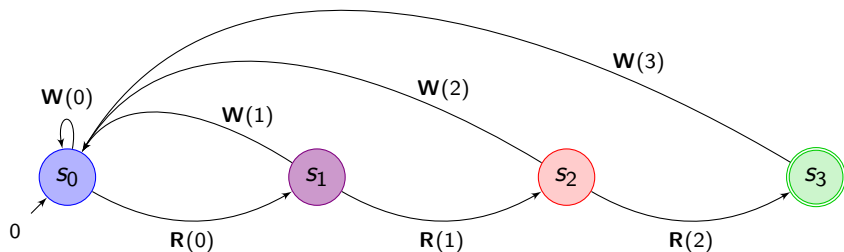
Filter protocol \mathcal{F}_4 

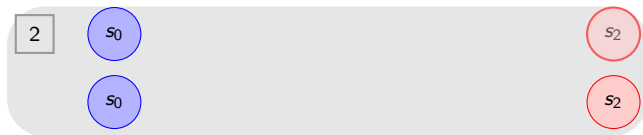
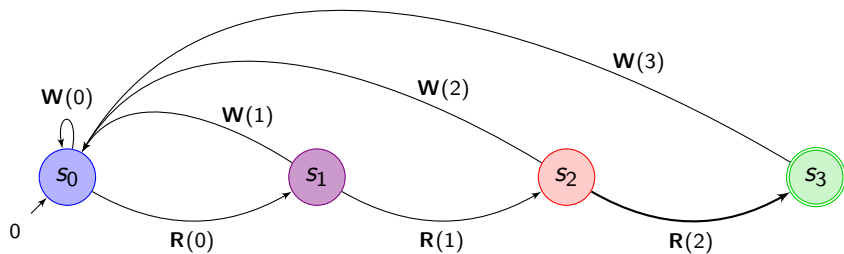
Filter protocol \mathcal{F}_4 

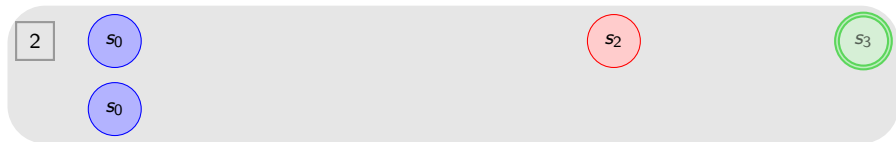
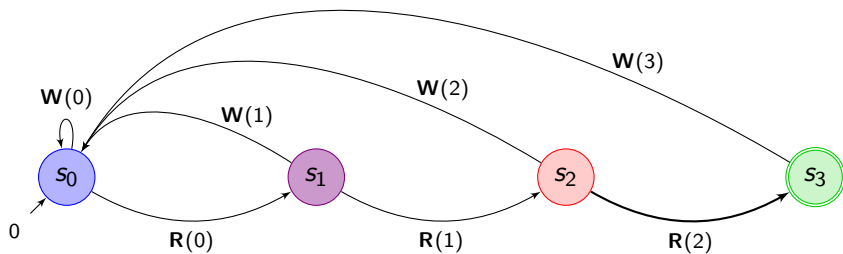
Filter protocol \mathcal{F}_4 

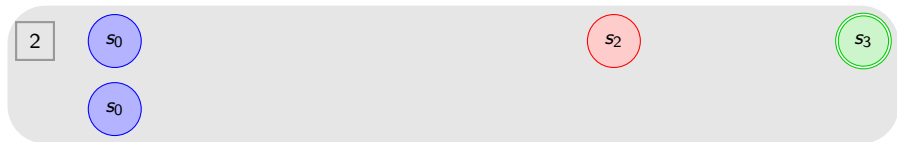
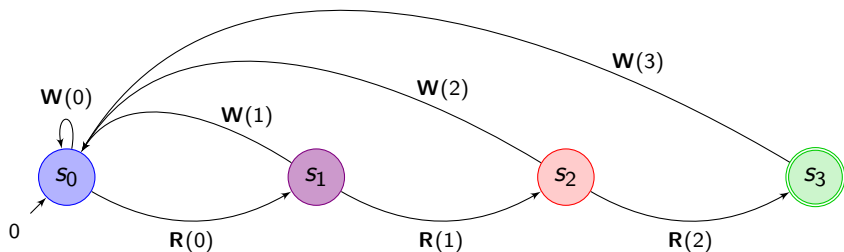
Filter protocol \mathcal{F}_4 

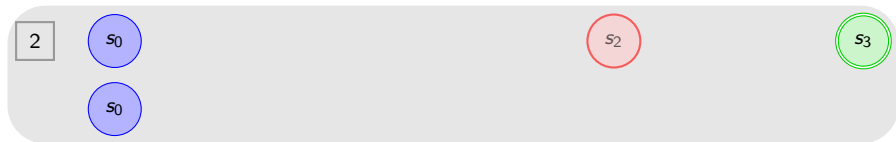
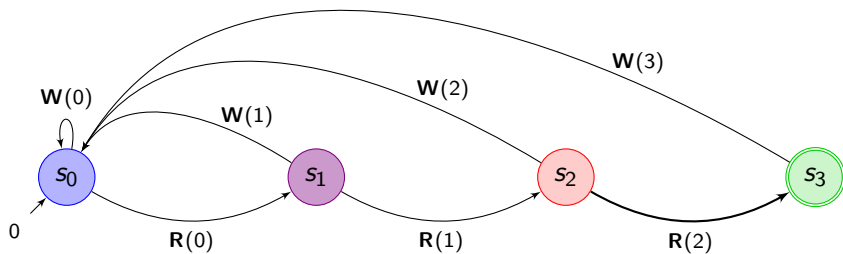
Filter protocol \mathcal{F}_4 

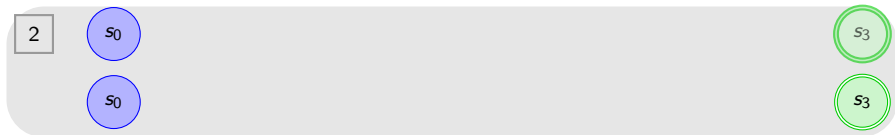
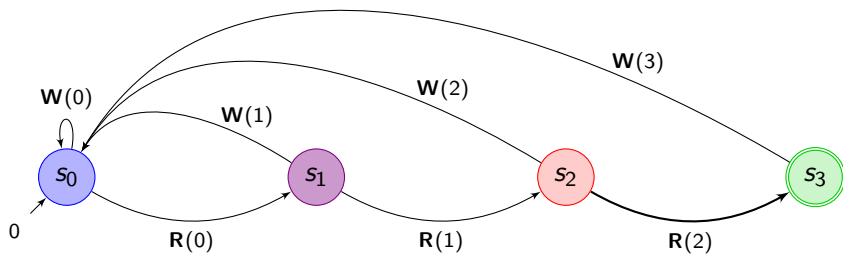
Filter protocol \mathcal{F}_4 

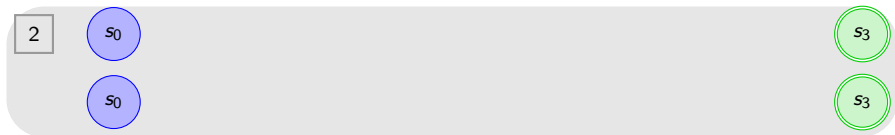
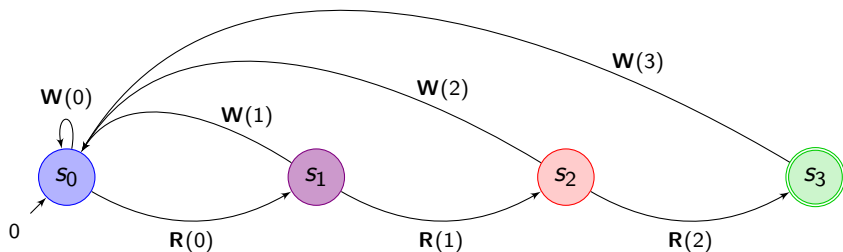
Filter protocol \mathcal{F}_4 

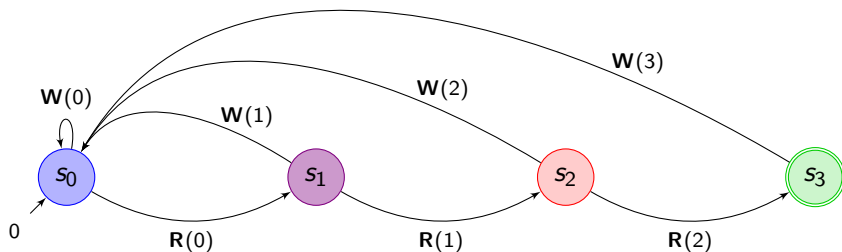
Filter protocol \mathcal{F}_4 

Filter protocol \mathcal{F}_4 

Filter protocol \mathcal{F}_4 

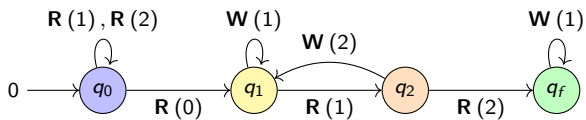
Filter protocol \mathcal{F}_4 

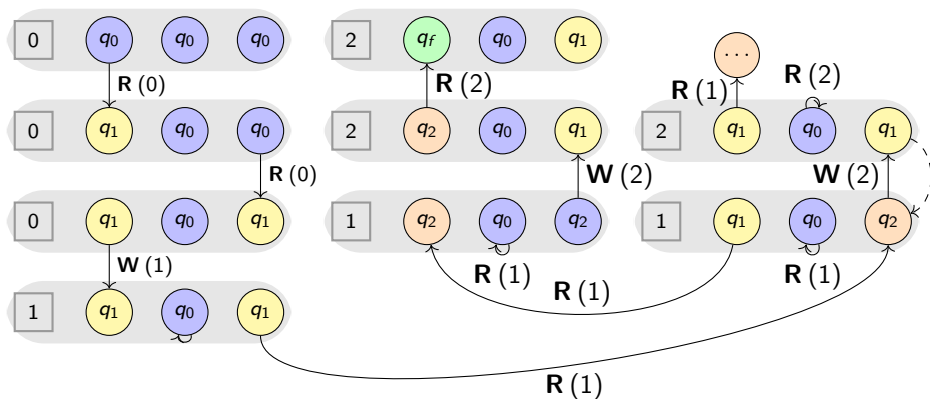
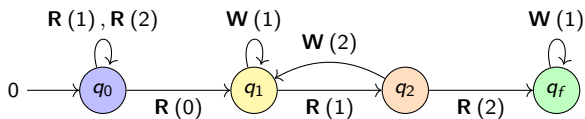
Filter protocol \mathcal{F}_4 

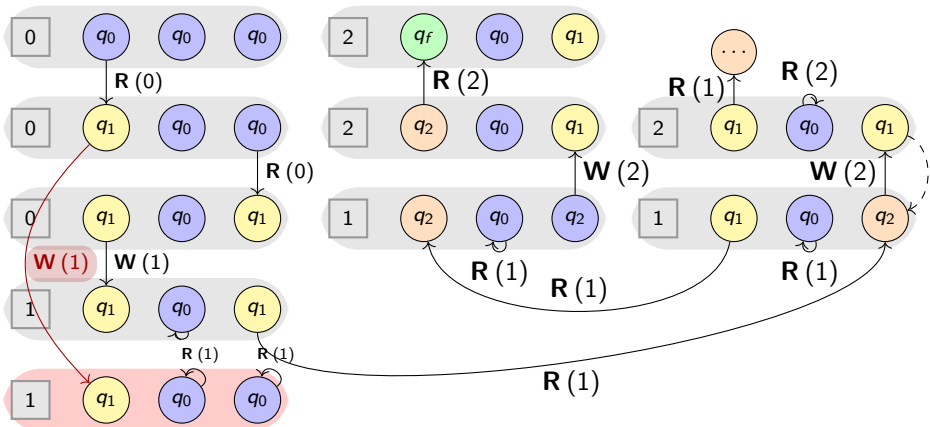
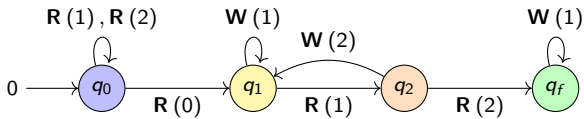
Filter protocol \mathcal{F}_4 

Conclusion of the filter protocol

$\diamond q_f$ is reached **almost-surely** if, and only if, the parameter is **larger** than the number of states -1 .







What we are looking for

A **limit** behaviour, if possible

What we are looking for

A **limit** behaviour, if possible

Definition (Cut-off [EK00])

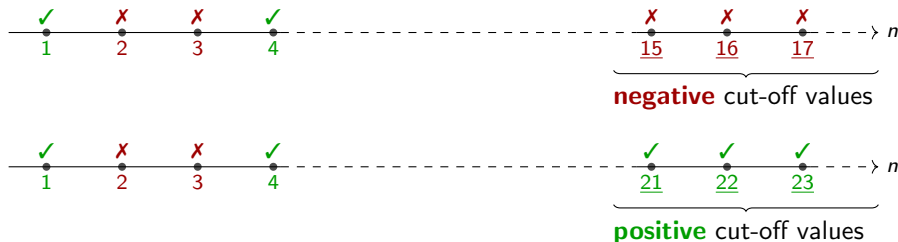
Let N a parameter. If $\forall n \geq N \mathbb{P}_n(\Diamond q_f) = 1$ or $\forall n \geq N \mathbb{P}_n(\Diamond q_f) < 1$, then N is a cut-off.

What we are looking for

A **limit** behaviour, if possible

Definition (Cut-off [EK00])

Let N a parameter. If $\forall n \geq N \mathbb{P}_n(\Diamond q_f) = 1$ or $\forall n \geq N \mathbb{P}_n(\Diamond q_f) < 1$, then N is a cut-off.



Existence of a cut-off

Theorem (Bouyer, Markey, Randour, Sangnier, S. in *ICALP'2016*)

Given a protocol \mathcal{P} there **always exists** a cut-off N .

Existence of a cut-off

Theorem (Bouyer, Markey, Randour, Sangnier, S. in *ICALP'2016*)

Given a protocol \mathcal{P} there **always exists** a cut-off N .



Idea:

- Particular case of a **well-structured-transition-system** [FS01];
- Post and Pre operations preserves **upward-closed** sets;
- Initial and final configuration spaces are **upward-closed**.

Existence of a cut-off

Theorem (Bouyer, Markey, Randour, Sangnier, S. in *ICALP'2016*)

Given a protocol \mathcal{P} there **always exists** a cut-off N .



Idea:

- Particular case of a **well-structured-transition-system** [FS01];
- Post and Pre operations preserves **upward-closed** sets;
- Initial and final configuration spaces are **upward-closed**.

Existence of a cut-off

Theorem (Bouyer, Markey, Randour, Sangnier, S. in *ICALP'2016*)

Given a protocol \mathcal{P} there **always exists** a cut-off N .



Idea:

- Particular case of a **well-structured-transition-system** [FS01];
- Post and Pre operations preserves **upward-closed** sets;
- Initial and final configuration spaces are **upward-closed**.

→ How to efficiently decide the type (**positive** or **negative**) of the cut-off?

Decision problem

Problem

- Input: a protocol \mathcal{P} .
- Output: **YES** if there exists a **positive** cut-off, **NO** otherwise.

Decision problem

Problem

- Input: a protocol \mathcal{P} .
- Output: **YES** if there exists a **positive** cut-off, **NO** otherwise.

In the size of the protocol:

- So far, all examples have **linear** cut-off.

Decision problem

Problem

- Input: a protocol \mathcal{P} .
- Output: **YES** if there exists a **positive** cut-off, **NO** otherwise.

In the size of the protocol:

- So far, all examples have **linear** cut-off.
- There exist protocols with **exponential negative** cut-off.

Theorem (ICALP'2016)

The cut-off decision problem is PSPACE-hard

Decision problem

Problem

- Input: a protocol \mathcal{P} .
- Output: **YES** if there exists a **positive** cut-off, **NO** otherwise.

In the size of the protocol:

- So far, all examples have **linear** cut-off.
- There exist protocols with **exponential negative** cut-off.
- Existence proof: minimal elements are shown to be **doubly-exponential** [Rac78].

Theorem (ICALP'2016)

The cut-off decision problem is PSPACE-hard and can be solved in EXPSPACE.

Extensions

Previous results are extended, with the same complexity, to:

- A distinguished **leader** process;
- Multiple registers with **move** operations;
- **Simple** upward-closed target objectives (instead of coverability);
- Almost-sure **repeated** reachability.

Extensions

Previous results are extended, with the same complexity, to:

- A distinguished **leader** process;
- Multiple registers with **move** operations;
- **Simple** upward-closed target objectives (instead of coverability);
- Almost-sure **repeated** reachability.

↪ Extension to the **local strategy** synthesis (with positive and zero probability).

Summary

- 1 Introduction
- 2 Concurrent Games
- 3 Parametrized Protocols
- 4 Conclusions**

Summary

Summary

Stochastic games: understanding of fundamental models

- Existence of a Nash equilibrium is undecidable even with three players and qualitative terminal-reachability and safety objectives.
- Narrow **the gap** between known existence and undecidability.
- Hope in the **approximation case** (ϵ -NE, imprecise deviations).

Summary

Stochastic games: understanding of fundamental models

- Existence of a Nash equilibrium is undecidable even with three players and qualitative terminal-reachability and safety objectives.
- Narrow **the gap** between known existence and undecidability.
- Hope in the **approximation case** (ε -NE, imprecise deviations).

↪ Open problems:

- Two-player case;
- Terminal-reachability only objectives;
- Exact complexity of equilibrium under ε -precise deviations.

Summary

Parametrized verification

- **Qualitative** analysis of almost-sure reachability.
- Several proposed **extensions**.
- Proof techniques (well-quasi-orders) reusable in other contexts.

Summary

Parametrized verification

- **Qualitative** analysis of almost-sure reachability.
- Several proposed **extensions**.
- Proof techniques (well-quasi-orders) reusable in other contexts.

~> Open problems:

- Fundamental study: **atomic operations**;
- Process identifiers;
- **Limit-sure**;
- **Quantitative analysis**;
- **Local strategy** synthesis for almost-sure reachability.
- NE with local strategies?

Perspectives

~> Common goals:

- Complexity gaps.
- More complex objectives: **model checking**.
- **Efficient** strategy synthesis (SMT-solvers, non-linear optimization).

Thank you.

Bibliography I

- [BBMU11] Patricia Bouyer, Romain Brenguier, Nicolas Markey, and Michael Ummels. Nash equilibria in concurrent games with Büchi objectives. In Proc. 30th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'11), volume 13 of LIPIcs, pages 375–386. Leibniz-Zentrum für Informatik, 2011.
- [CJM04] Krishnendu Chatterjee, Marcin Jurdziński, and Rupak Majumdar. On Nash equilibria in stochastic games. In Proc. 18th International Workshop on Computer Science Logic (CSL'04), volume 3210 of Lecture Notes in Computer Science, pages 26–40. Springer, 2004.
- [DEGM15] Antoine Durand-Gasselin, Javier Esparza, Pierre Ganty, and Rupak Majumdar. Model checking parameterized asynchronous shared-memory systems. In Daniel Kroening and Corina S. Pasareanu, editors, Proceedings of the 27th International Conference on Computer Aided Verification (CAV'15), volume 9206 of Lecture Notes in Computer Science, pages 67–84. Springer-Verlag, July 2015.
- [EGM13] Javier Esparza, Pierre Ganty, and Rupak Majumdar. Parameterized verification of asynchronous shared-memory systems. In Natasha Sharygina and Helmut Veith, editors, Proceedings of the 25th International Conference on Computer Aided Verification (CAV'13), volume 8044 of Lecture Notes in Computer Science, pages 124–140. Springer-Verlag, July 2013.

Bibliography II

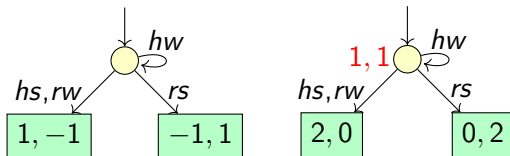
- [EK00] E. Allen Emerson and Vineet Kahlon. Reducing model checking of the many to the few. In David McAllester, editor, Proceedings of the 17th International Conference on Automated Deduction (CADE'00), volume 1831 of Lecture Notes in Artificial Intelligence, pages 236–254. Springer-Verlag, June 2000.
- [Esp14] Javier Esparza. Keeping a crowd safe: On the complexity of parameterized verification (invited talk). In Ernst W. Mayr and Natacha Portier, editors, Proceedings of the 31st Symposium on Theoretical Aspects of Computer Science (STACS'14), volume 25 of Leibniz International Proceedings in Informatics, pages 1–10. Leibniz-Zentrum für Informatik, March 2014.
- [FS01] Alain Finkel and Philippe Schnoebelen. Well structured transition systems everywhere! Theoretical Computer Science, 256(1–2):63–92, 2001.
- [KS81] P. R. Kumar and T. H. Shiao. Existence of value and randomized strategies in zero-sum discrete-time stochastic dynamic games. SIAM Journal on Control and Optimization, 19(5):617–634, 1981.
- [Min67] Marvin Minsky. Computation: Finite and Infinite Machines. Prentice Hall International, 1967.
- [Nas50] John F. Nash. Equilibrium points in n -person games. Proceedings of the National Academy of Sciences of the United States of America, 36(1):48–49, 1950.
- [Rac78] Charles Rackoff. The covering and boundedness problems for vector addition systems. Theoretical Computer Science, 6:223–231, 1978.

Bibliography III

- [SS01] Piercesare Secchi and William D. Sudderth. Stay-in-a-set games. International Journal of Game Theory, 30:479–490, 2001.
- [UW11] Michael Ummels and Dominik Wojtczak. The complexity of Nash equilibria in limit-average games. Technical Report abs/1109.6220, CoRR, 2011.

Does a mixed Nash Equilibrium always exist?

Idea: two player concurrent zero-sum games may not have optimal strategies but only ε -optimal strategies (for any $\varepsilon > 0$).



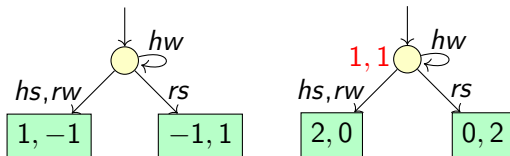
Hide-or-Run game.

Shifted hide-or-Run game

Value problem in a zero-sum game is not a special case of Nash Equilibrium problem with positive terminal rewards.

Does a mixed Nash Equilibrium always exist?

Idea: two player concurrent zero-sum games may not have optimal strategies but only ε -optimal strategies (for any $\varepsilon > 0$).



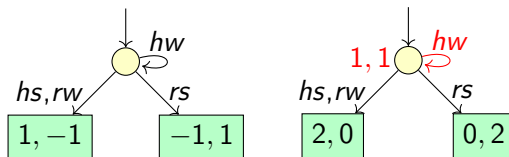
Hide-or-Run game.

Shifted hide-or-Run game

Value problem in a zero-sum game is not a special case of Nash Equilibrium problem with positive terminal rewards.

Does a mixed Nash Equilibrium always exist?

Idea: two player concurrent zero-sum games may not have optimal strategies but only ε -optimal strategies (for any $\varepsilon > 0$).



Hide-or-Run game.

Shifted hide-or-Run game

Value problem in a zero-sum game is not a special case of Nash Equilibrium problem with positive terminal rewards.

