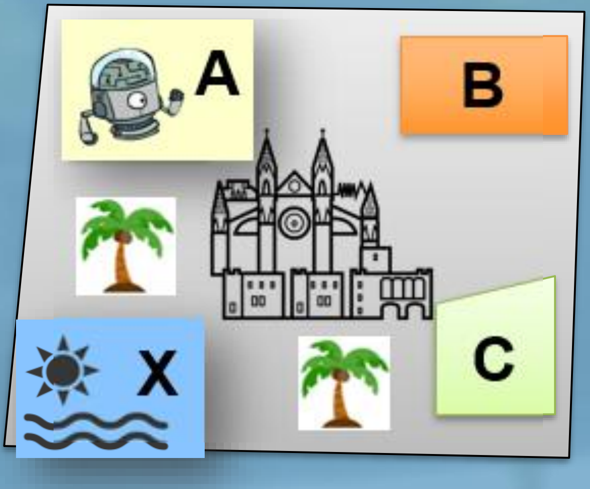
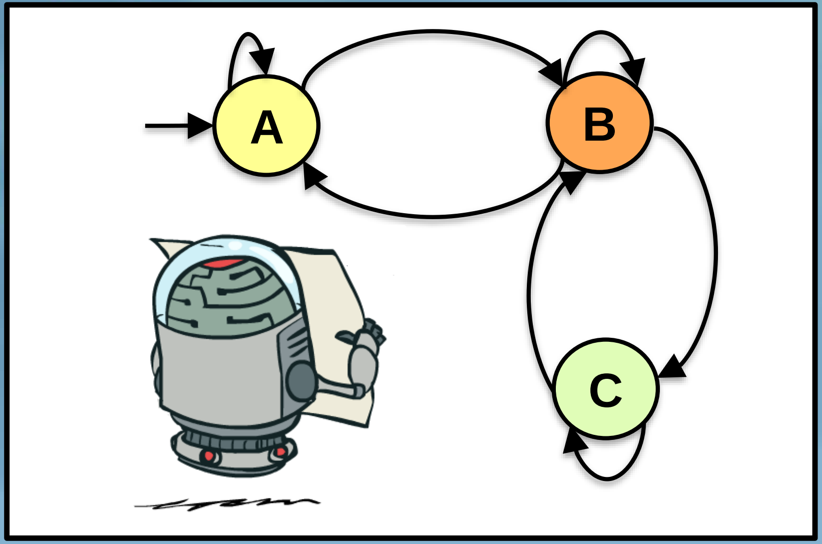
Graz University of Technology
Institute for Applied Information
Processing and Communications

Reactive Synthesis

Bettina Könighofer

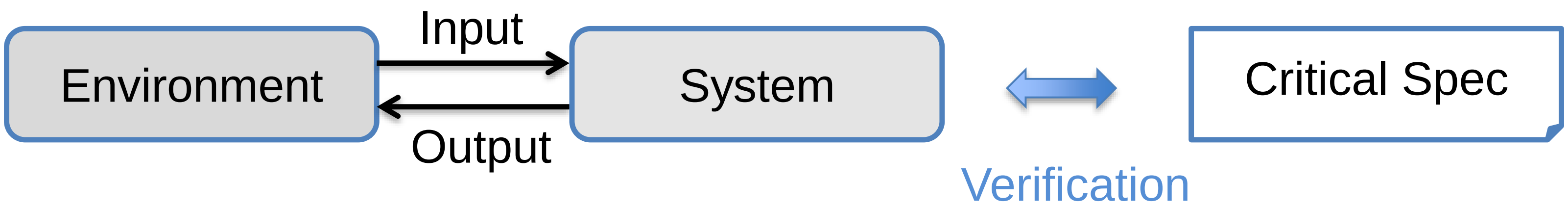




Model Checking SS24

May 13th 2024

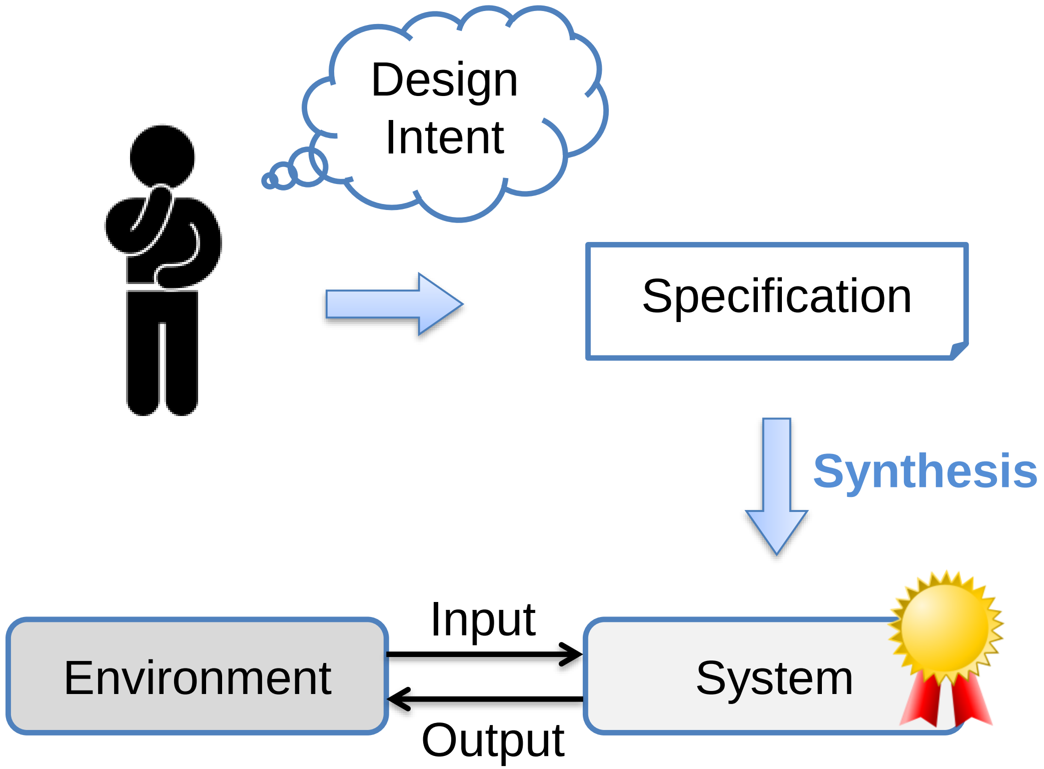
Verification



One needs to do a lot of work:

- Need to write the system + specification

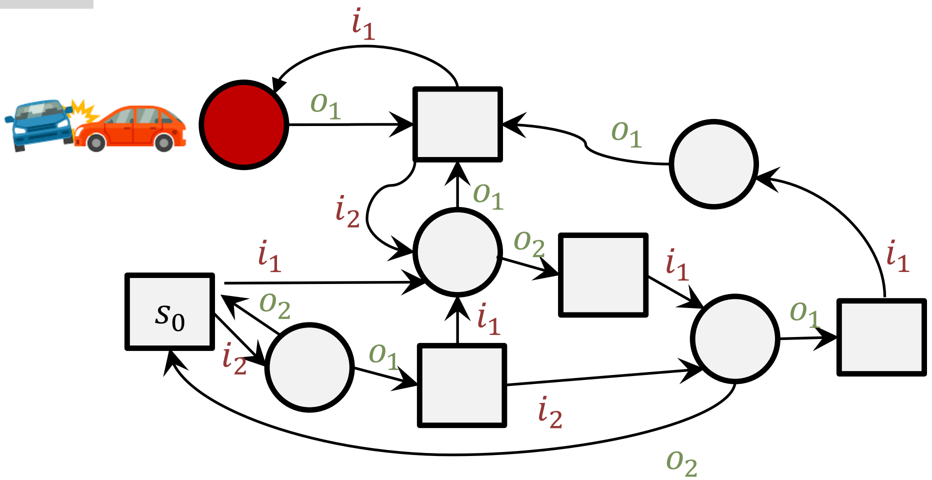
Synthesize it!



How can we compute the Sytem?

- By solving a **Game**
- Played between the Environment-Player and the System-Player

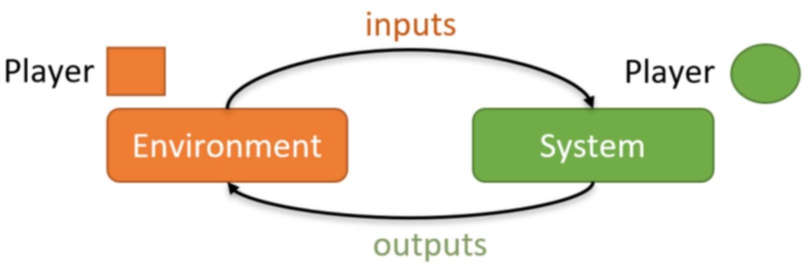
Synthesis is a Game

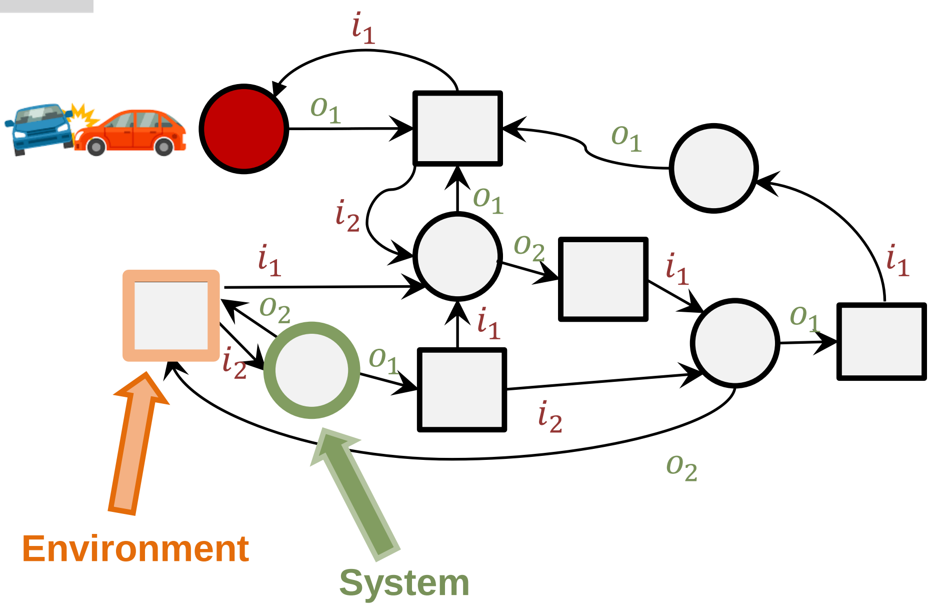


Formal safety specification

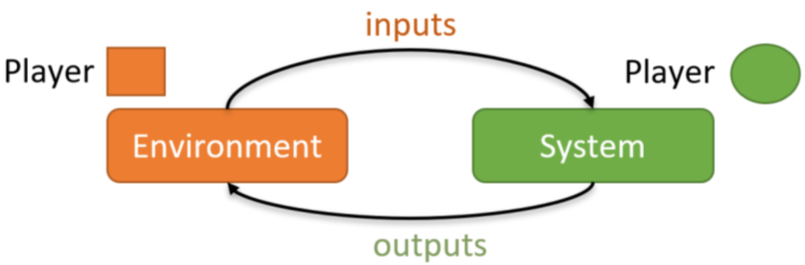
Model of environment

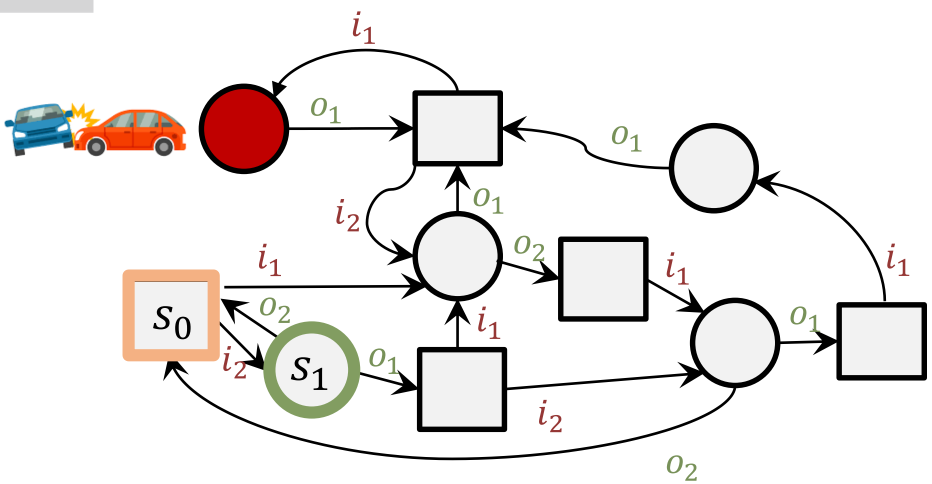
Synthesis is a Game



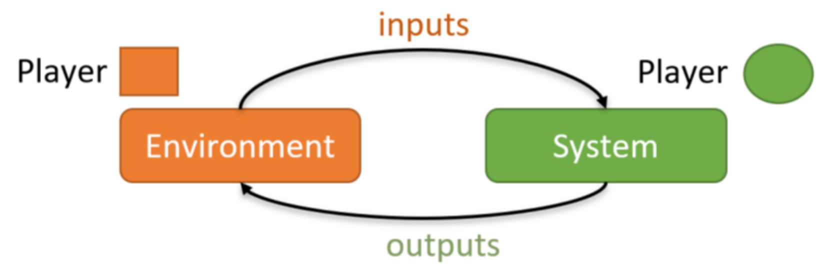


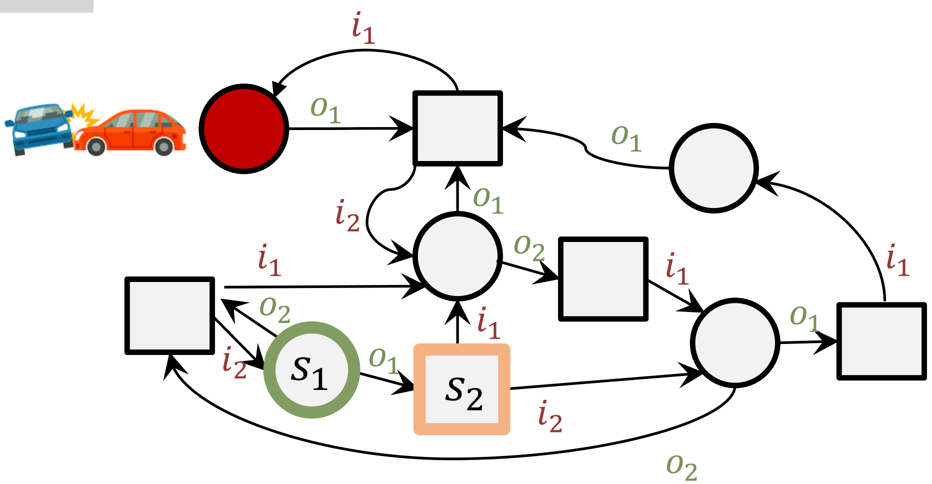
Synthesis is a Game



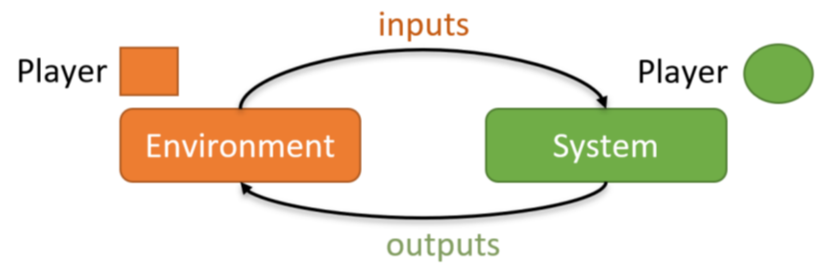


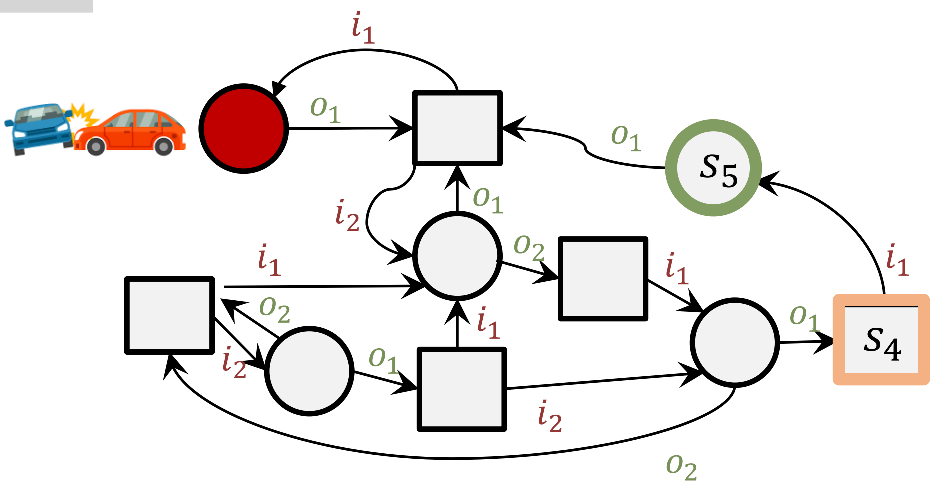
Synthesis is a Game





Synthesis is a Game

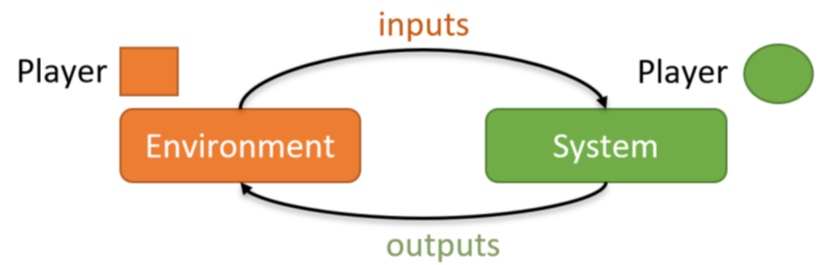


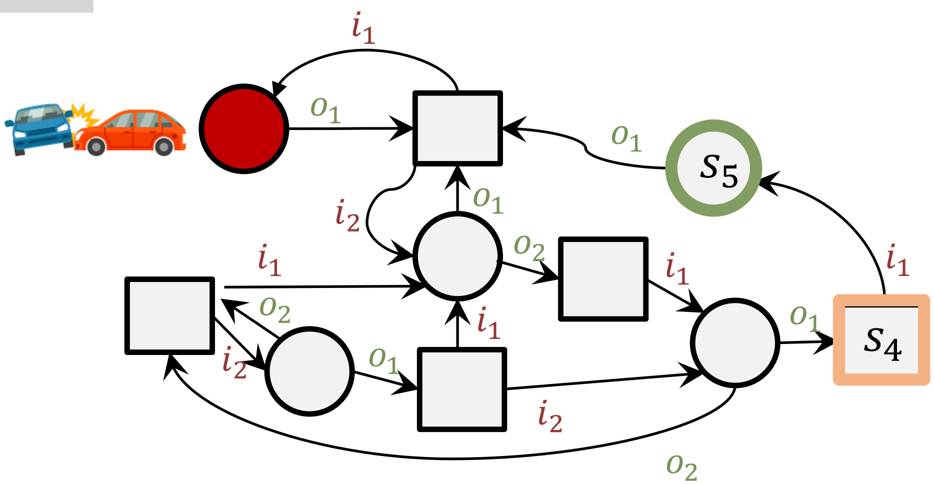


System Player wins, if ● is **never** visited

Winning Region: States from which the system can enforce that ● is **never** visited

Synthesis is a Game



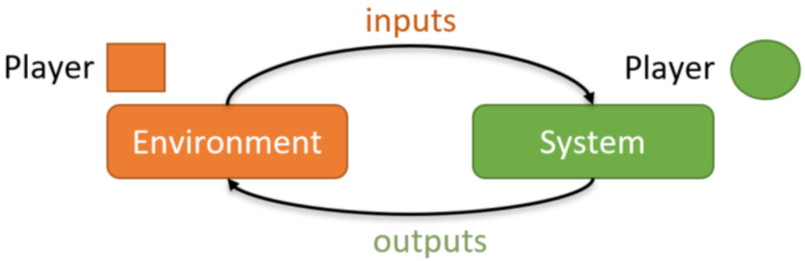


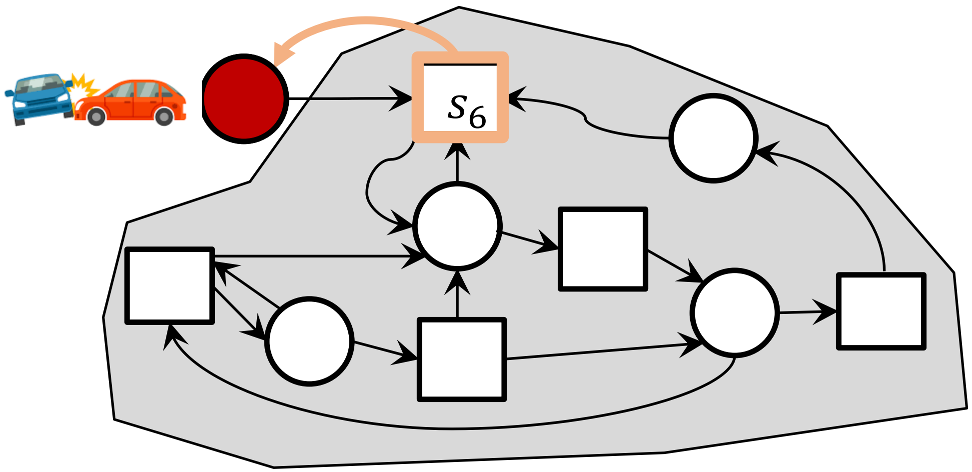
What is the winning region for this example?

System Player wins, if is **never** visited

Winning Region: States from which the system can enforce that is **never** visited

Synthesis is a Game



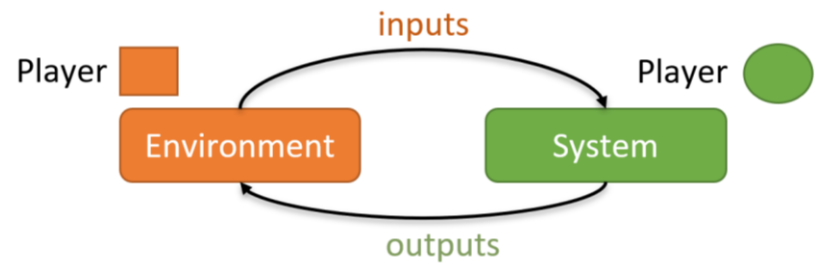


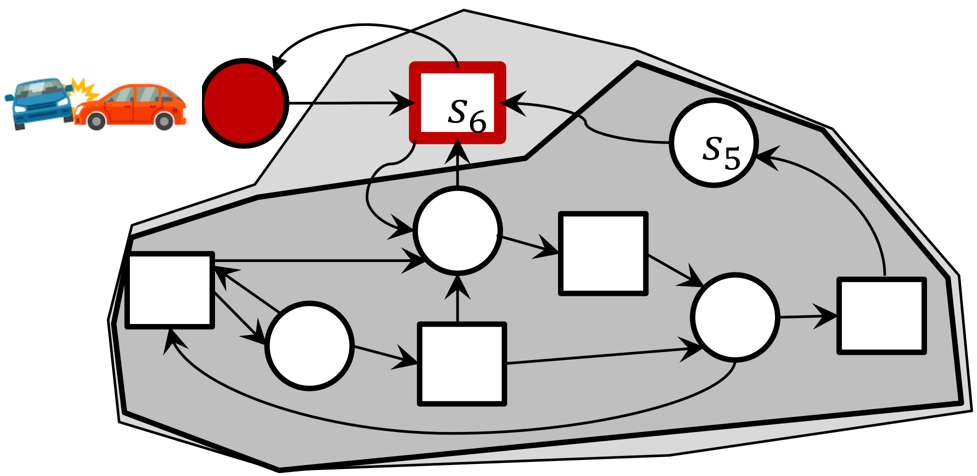
What is the winning region for this example?

System Player wins, if ● is **never** visited

Winning Region: States from which the system can enforce that ● is **never** visited

Synthesis is a Game



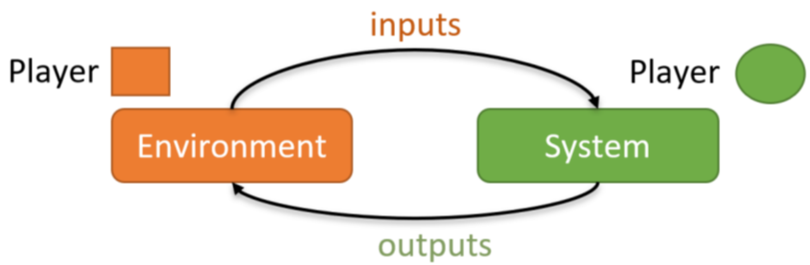


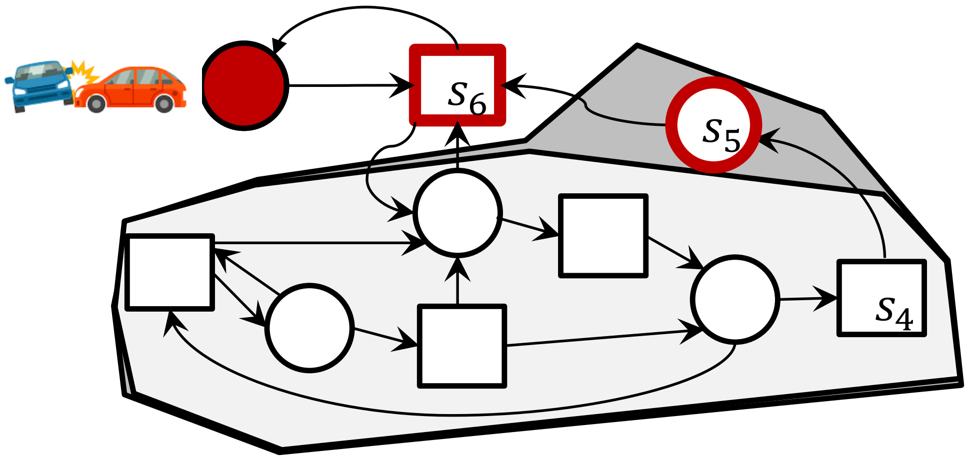
What is the winning region for this example?

System Player wins, if (red circle) is **never** visited

Winning Region: States from which the system can enforce that (red circle) is **never** visited

Synthesis is a Game



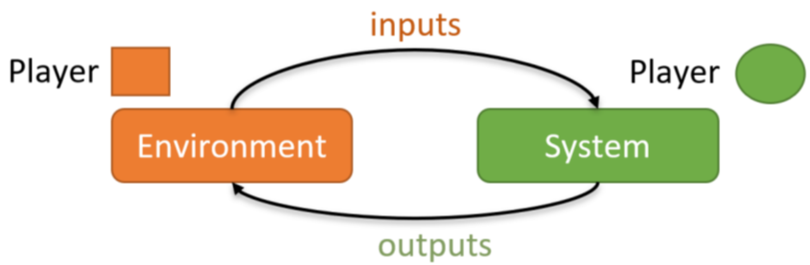


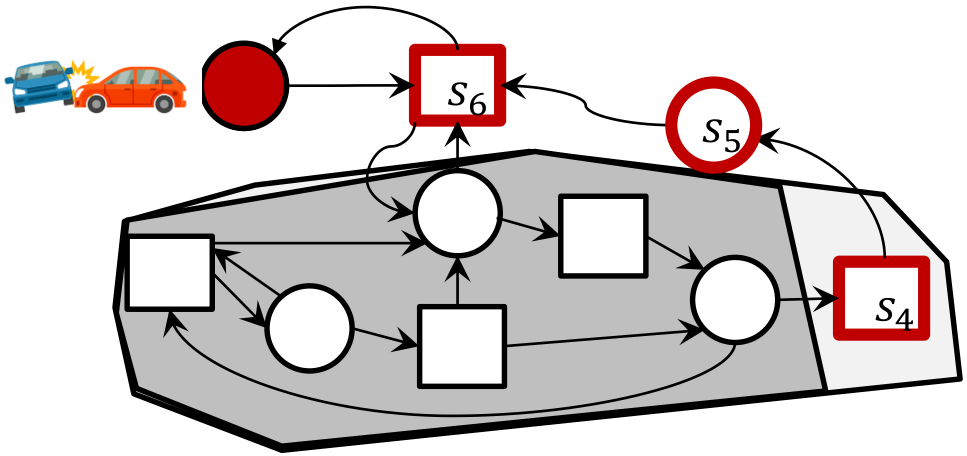
What is the winning region for this example?

System Player wins, if ● is **never** visited

Winning Region: States from which the system can enforce that ● is **never** visited

Synthesis is a Game



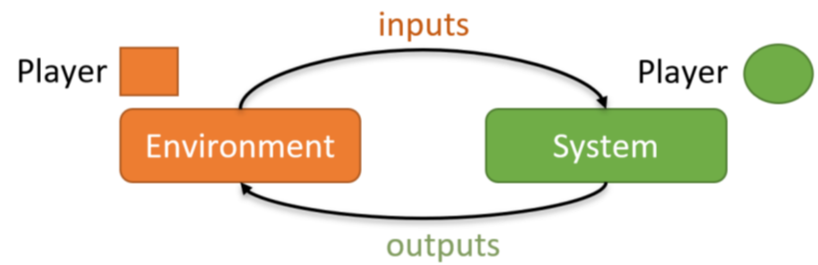


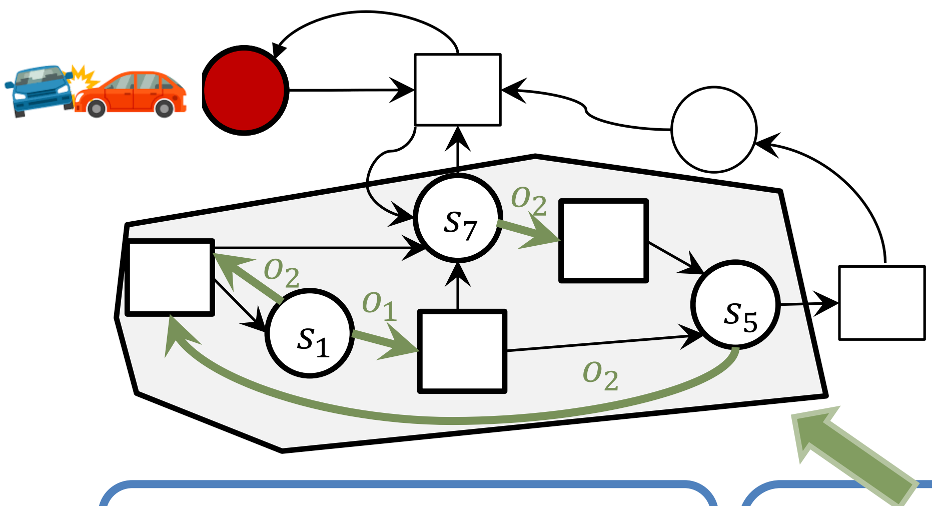
What is the winning region for this example?

System Player wins, if ● is **never** visited

Winning Region: States from which the system can enforce that ● is **never** visited

Synthesis is a Game



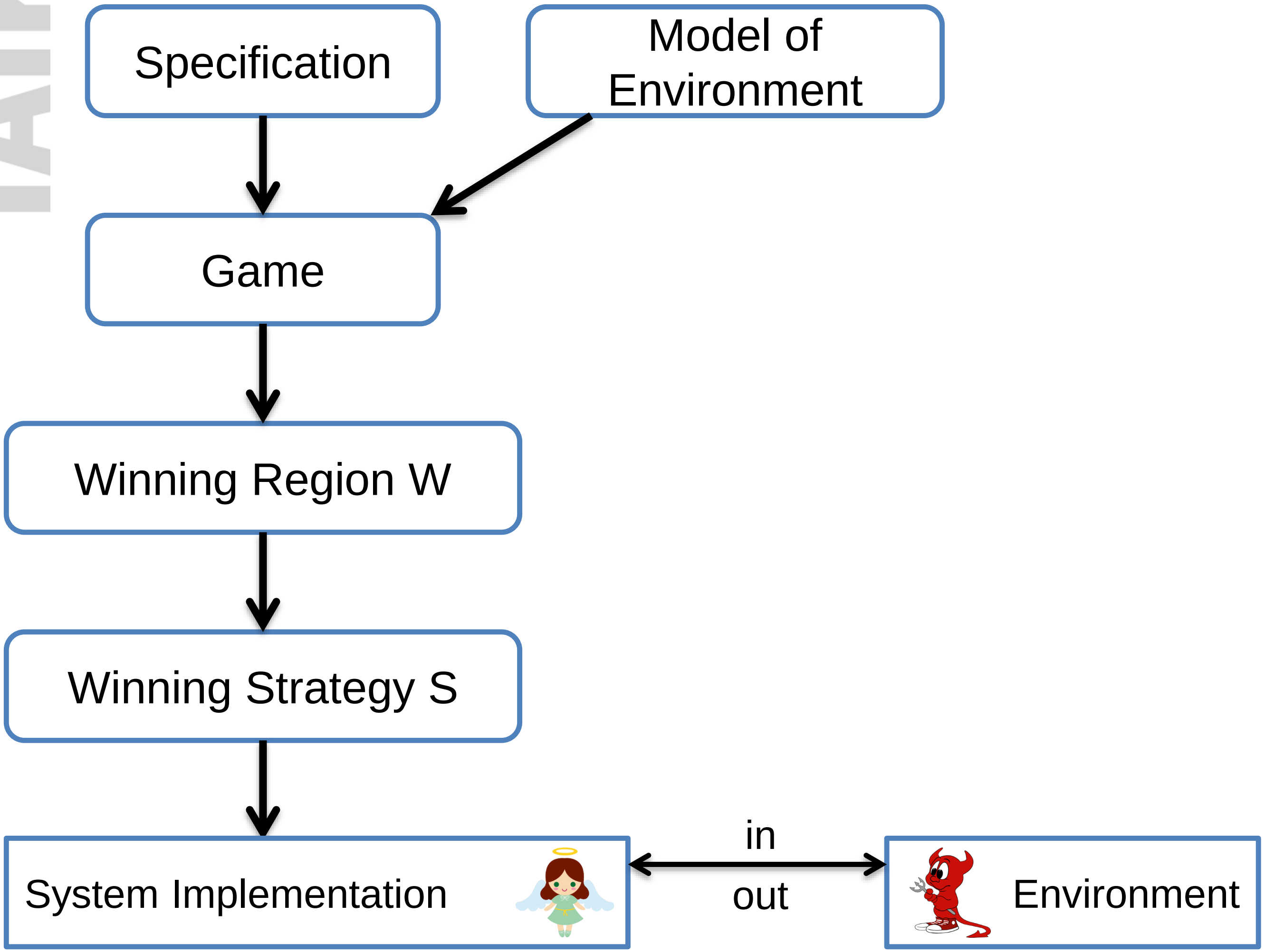


What is the winning region for this example?

System Player wins, if ● is **never** visited

Winning Region: States from which the system can enforce that ● is **never** visited

Synthesis Flow



Outline: Synthesis is a Game

- What is a game?
- Games on Graphs
- Solving Games

What characterizes a game?

Games are **fun**

Several **players**

At least 2

Games

Goal: **win** the game

Competitive:
If one wins, the other loses

You need a good **strategy** to win

Players can make **moves**

Moves must follow **rules**

Game has a **state**

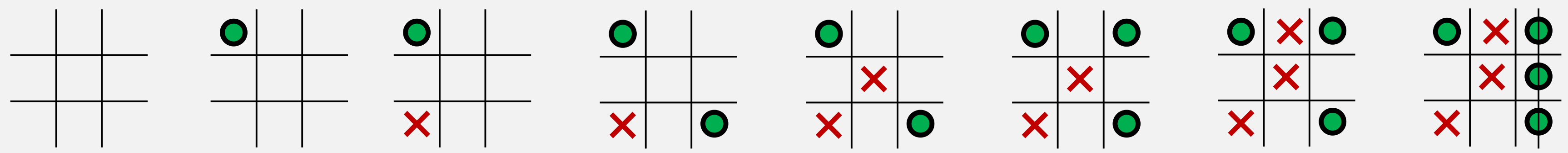
State is changed by moves

Example: Tic-Tac-Toe

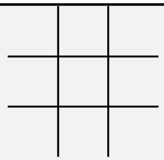
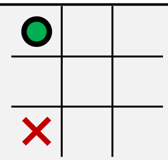
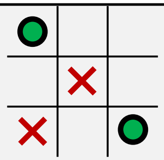
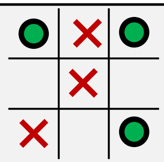
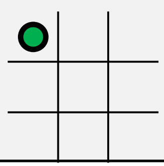
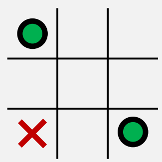
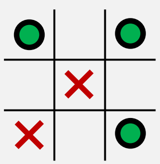
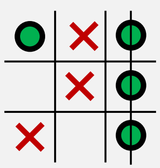
- 2 Players: ● and ×
- Players can make moves
  - ●-player: place ●
  - ×-player: place ×
- Rules for moves:
  - Players move in alternation
  - ... can only pick free slots
  - ...
- Winning condition: connect 3
- Players play **against** each other

Terminology

- **Play:** Execution of the Game

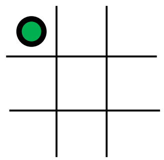


- **Strategy:** Defines what should be done when

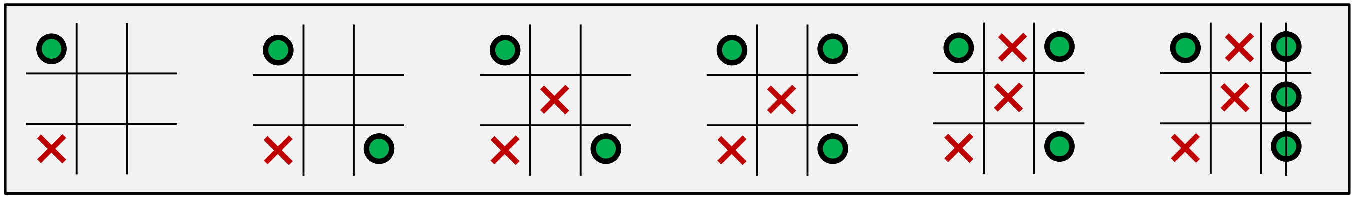
if:					and so on …
then:					and so on …

Terminology

- Solving a game = finding a strategy that wins any game
  - No matter how the opponent plays
- Such a strategy is called a **Winning Strategy**
  - Not always possible
  - Computation needs to think ahead
  - Example: we are × and need to respond to

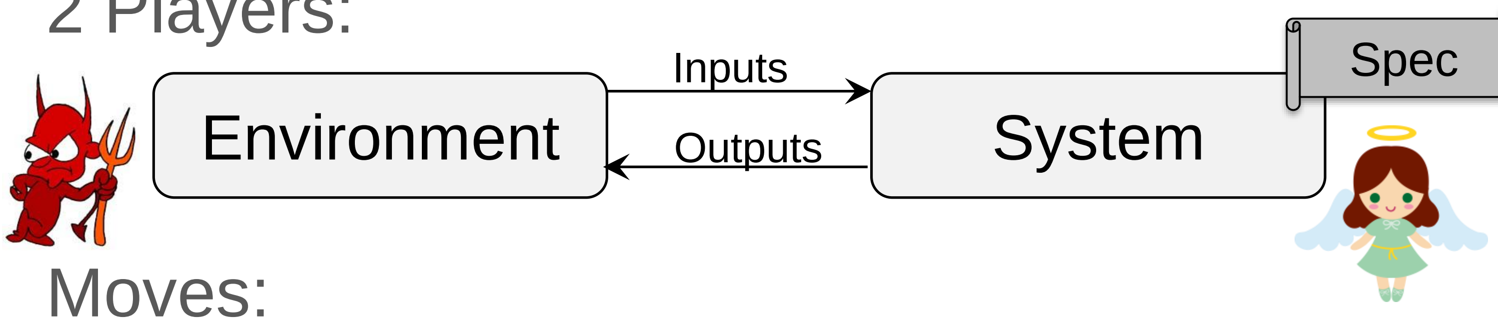


  - Take the lower left corner → dead in ≤ 5 moves:



Synthesis is a Game

- 2 Players:



- Moves:
  - Provide input values
    Provide output values
- Winning Condition:
  - Violate the specification
    Satisfy the specification
- Strategy of System:

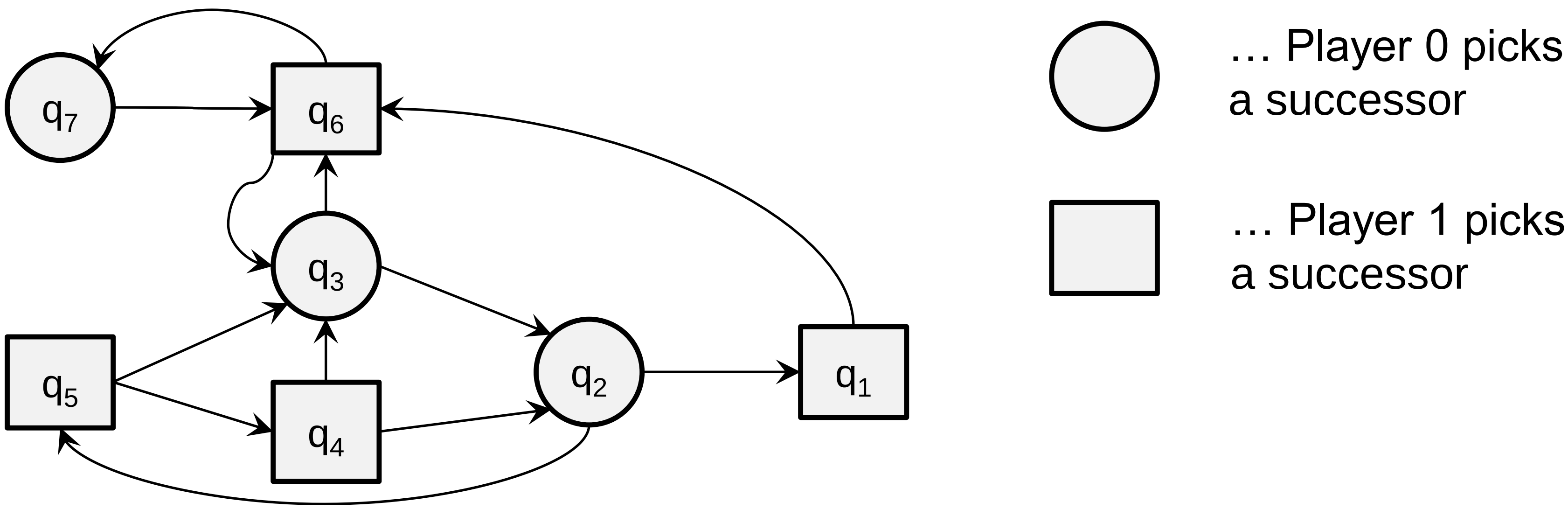
if:	state $s=q_7$ input $i=i_4$	$s=q_7$ $i=i_5$	$s=q_9$ $i=i_1$	and so on
then:	output $o=o_3$ next state $s=q_9$	$o=o_4$ $s=q_7$	$o=o_1$ $s=q_1$	and so on

Outline: Synthesis is a Game

- What is a game?
- Games on Graphs
- Game solving requires some math…
- Solving Games

Game = Graph + Winning condition

- Game = Graph + Winning condition
- Game graph: $G = (Q_0 \cup Q_1, E)$



Play ρ:

- Infinite sequence of states: $\rho = q_0 q_1 q_2 \ldots \in Q^\omega$

Games on Graphs

- Game = Graph + Winning condition
- Game graph G = (Q, E):
  - Every state has an outgoing edge:
    - $\forall q \in Q : \exists q' \in Q : (q, q') \in E$
  - Q is partitioned into Q_0 and Q_1:
    - $Q = Q_0 \cup Q_1$ with $Q_0 \cap Q_1 = \emptyset$
    - Player 0 picks a successor state in Q_0
    - Player 1 picks a successor state in Q_1

Game = Game Graph + Winning Condition

- **Winning condition** $\varphi: \mathrm{Q}^{\omega} \rightarrow \mathbb{B}$
  - ρ is won by the Player 0 iff $\varphi(\rho) = \top$
  - ρ is won by the Player 1 iff $\varphi(\rho) = \bot$

- **Types of winning conditions**
  - Let $\mathrm{F} \subseteq Q$ be a set of states

1. Reach F at least once
2. Stay in F forever
3. Reach F infinitely often

- ☐ Safety Games
- ☐ Büchi Games
- ☐ Reachability Game

Game = Game Graph + Winning Condition

- **Winning condition** $\varphi: Q^{\omega} \rightarrow \mathbb{B}$
  - ρ is won by the Player 0 iff $\varphi(\rho) = \top$
  - ρ is won by the Player 1 iff $\varphi(\rho) = \bot$

- **Types of winning conditions**
  - Let $F \subseteq Q$ be a set of states

1. Reach F at least once
2. Stay in F forever
3. Reach F infinitely often

2	Safety Games
3	Büchi Games
1	Reachability Game

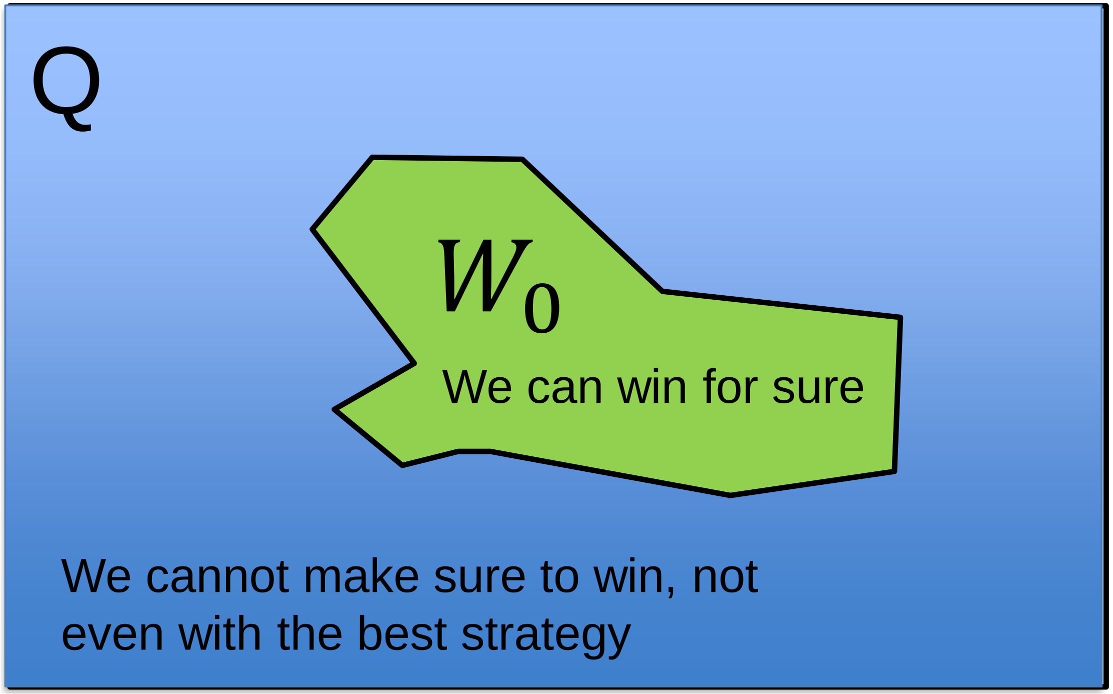
Winning Strategy

- Positional Strategy:
  - $f^0 : Q_0 \to Q$
  - A play $\rho = q_0 q_1 \ldots$ **follows** positional strategy $f^0 : Q_0 \to Q$ iff $q_{i+1} = f^0(q_i)$ for all $q_i \in Q_0$
- Winning Strategy:
  - Makes sure that P0 **always** wins

Winning Region

- W_0 is the set of states from which a winning strategy exists

ToDo What is the winning region W_1 of P1?



Winning Region

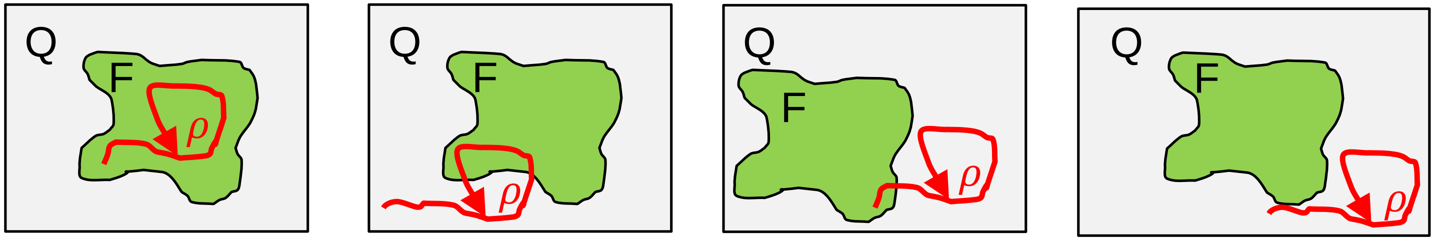
- W_0 is the set of states from which a winning strategy exists
- What is the winning region W_1 of P1?

Winning Conditions:

Reachability Games

- φ is defined using a set **F** of “target states”
- Player 0 wins a play $\rho = q_0 q_1$ ... iff **F** is visited
- $\varphi(\rho) \Leftrightarrow \exists i\colon q_i \in F$

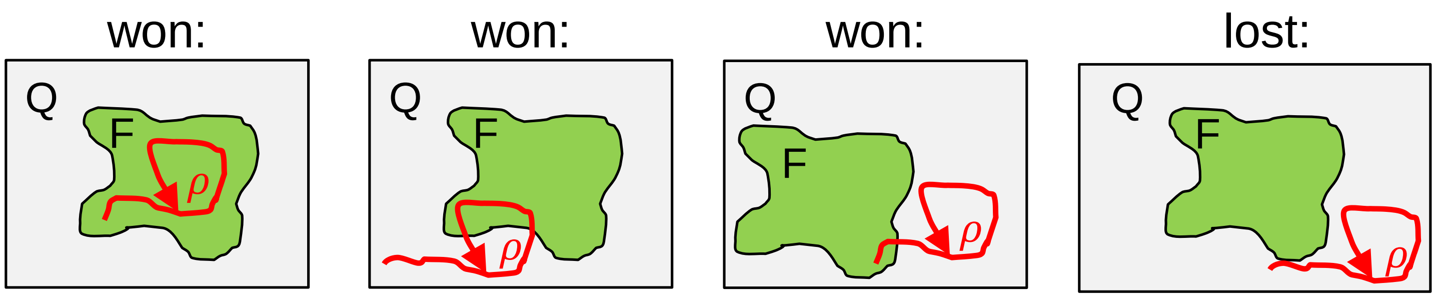
ToDo **Which plays are winning plays for P0?**



Winning Conditions:

Reachability Games

- φ is defined using a set **F** of "target states"
- Player 0 wins a play $\rho = q_0 q_1$ ... iff **F** is visited
- $\varphi(\rho) \Leftrightarrow \exists i: q_i \in F$



Winning Conditions:

Safety Games

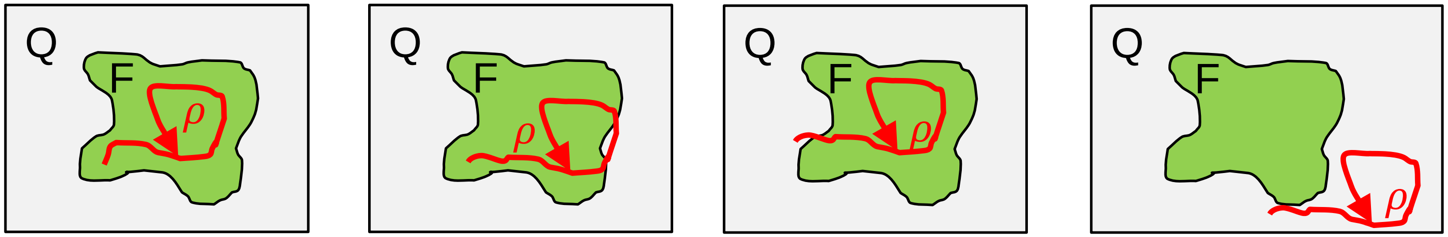
- φ is defined using a set **F** of "safe states"
- Player 0 wins a play $\rho = q_0 q_1 \dots$ iff **it stays in F**
- $\varphi(\rho) \Leftrightarrow \forall i : q_i \in F$

Winning Conditions:

Safety Games

- φ is defined using a set **F** of "safe states"
- Player 0 wins a play $\rho = q_0 q_1 \ldots$ iff **it stays in F**
- $\varphi(\rho) \Leftrightarrow \forall i: q_i \in F$

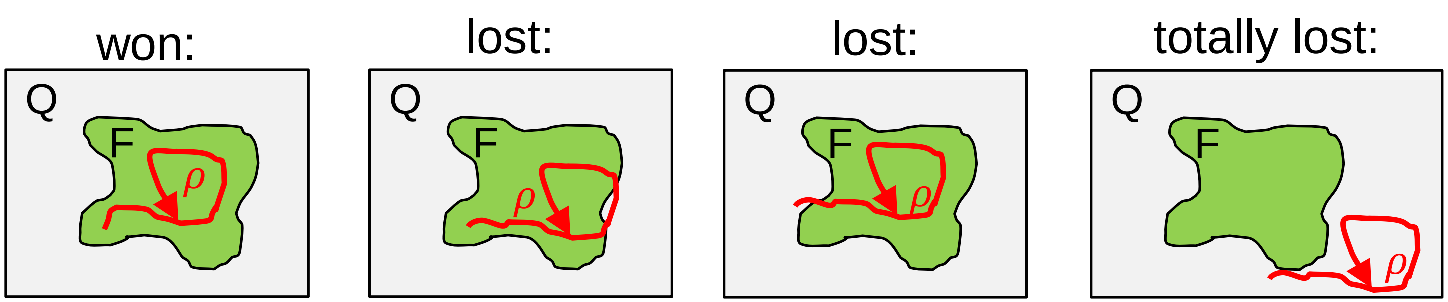
Which plays are winning plays for P0?



Winning Conditions:

Safety Games

- φ is defined using a set **F** of "safe states"
- Player 0 wins a play $\rho = q_0 q_1 \ldots$ iff **it stays in F**
- $\varphi(\rho) \Leftrightarrow \forall i: q_i \in F$



Winning Conditions:

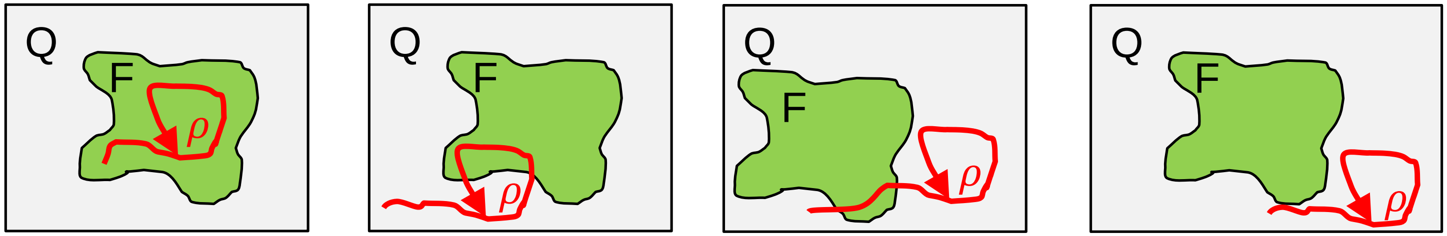
Büchi Games

- φ is defined using a set **F** of “accepting states”
- Player 0 wins a play ρ iff **F** is visited infinitely often

Winning Conditions:

Büchi Games

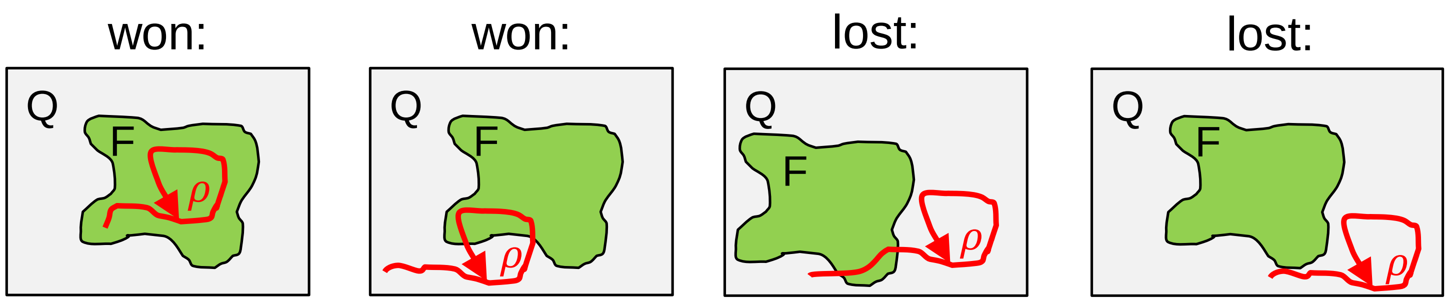
- φ is defined using a set **F** of "accepting states"
- Player 0 wins a play ρ iff **F** is visited infinitely often



Winning Conditions:

Büchi Games

- φ is defined using a set **F** of "accepting states"
- Player 0 wins a play ρ iff **F** is visited infinitely often



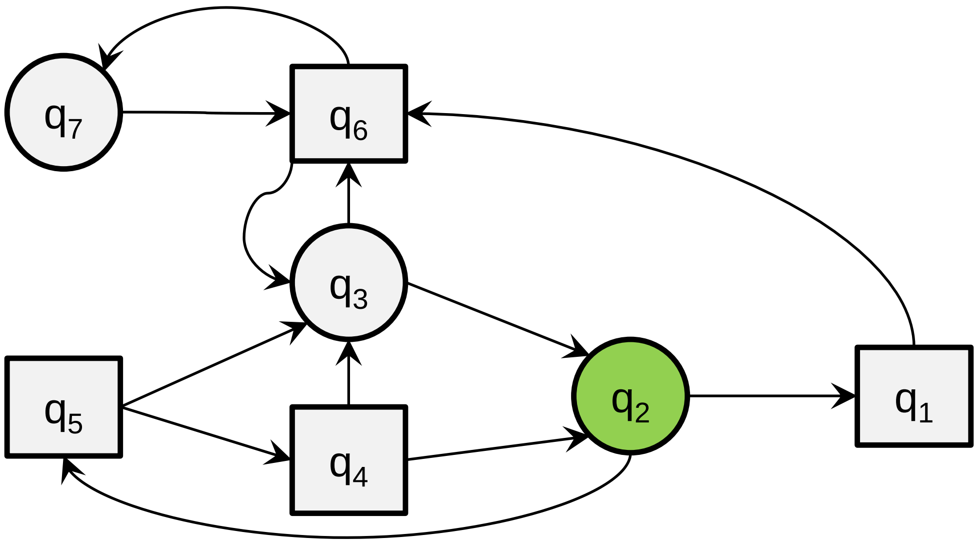
- $\mathrm{Inf}(\rho)$: the states occurring infinitely often in ρ
- $\varphi(\rho) \Leftrightarrow Inf(\rho) \cap F \neq \emptyset$

Outline: Synthesis is a Game

- What is a game?
- Games on Graphs
- Solving Games
  - Reachability
  - Safety

Reachability Game

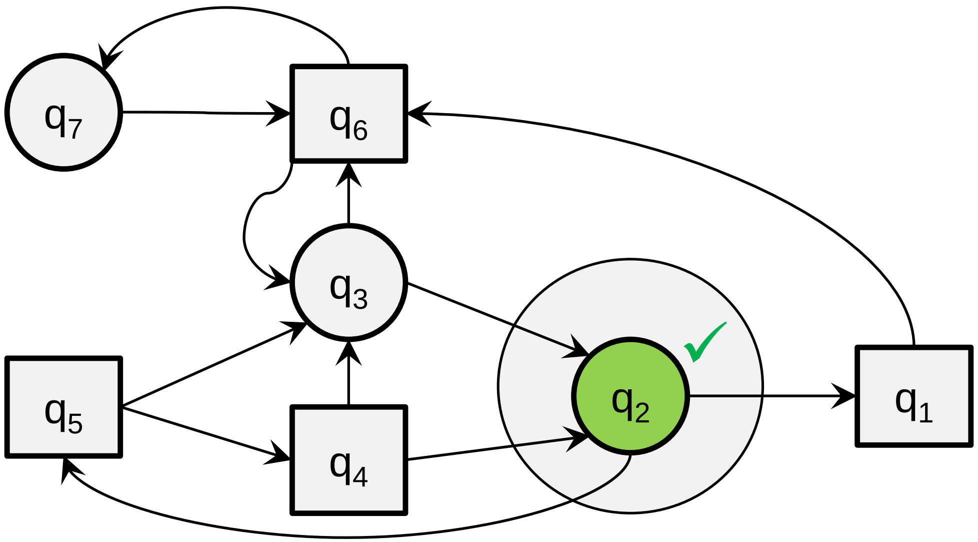
- Compute the winning region of Player 0 for:



- For the reachability game with $F = \{q_2\}$

Reachability Game

- Compute the winning region of Player 0 for:

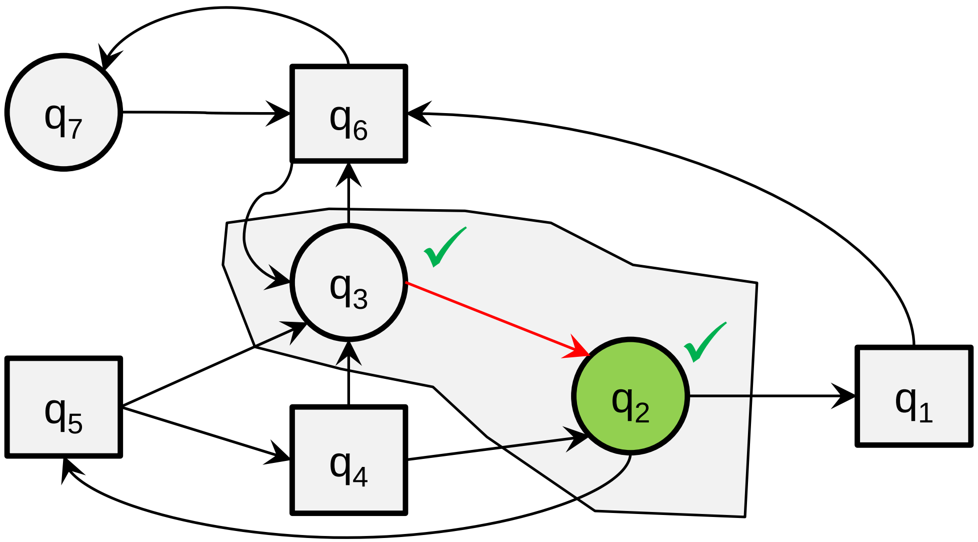


Every strategy is winning from q_2
➔
q_2 is part of the winning region

- For the reachability game with $F = \{q_2\}$

Reachability Game

- Compute the winning region of Player 0 for:



If we start from q_3, we can get into the winning region (go from q_3 to q_2, not q_6)

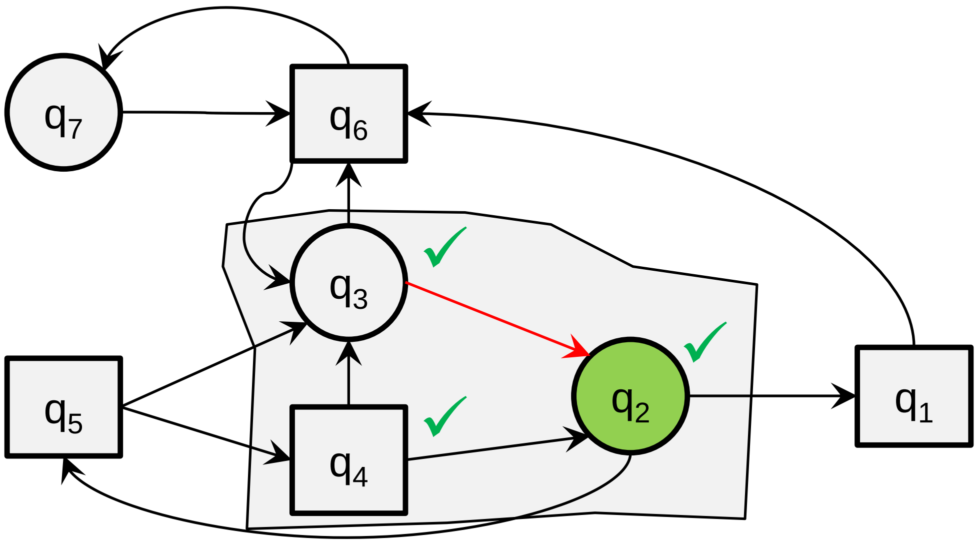
➔

q_3 is part of the winning region

- For the reachability game with $F = \{q_2\}$

Reachability Game

- Compute the winning region of Player 0 for:



If we start from q_4, we must end up in the winning region

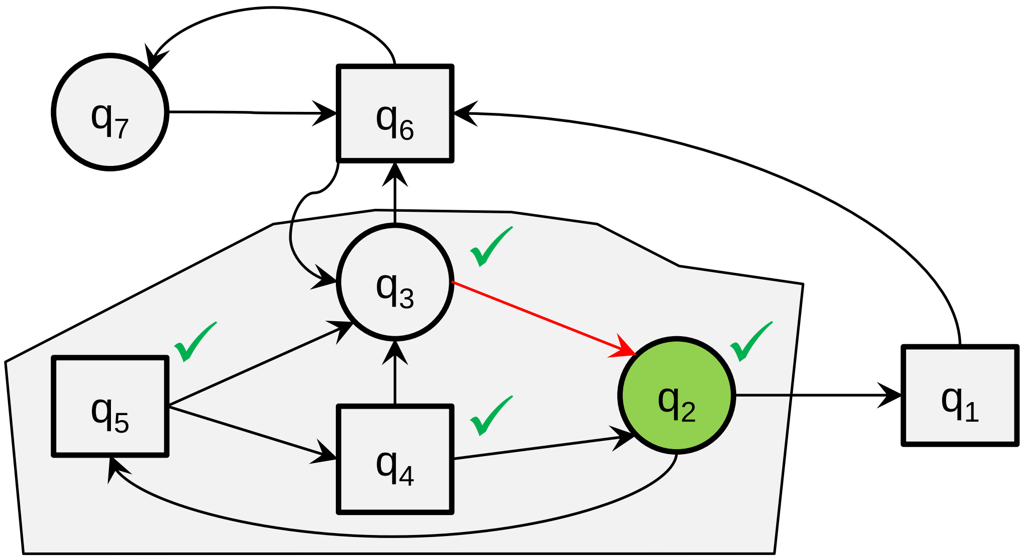
➔

q_4 is part of the winning region

- For the reachability game with $F = \{q_2\}$

Reachability Game

- Compute the winning region of Player 0 for:

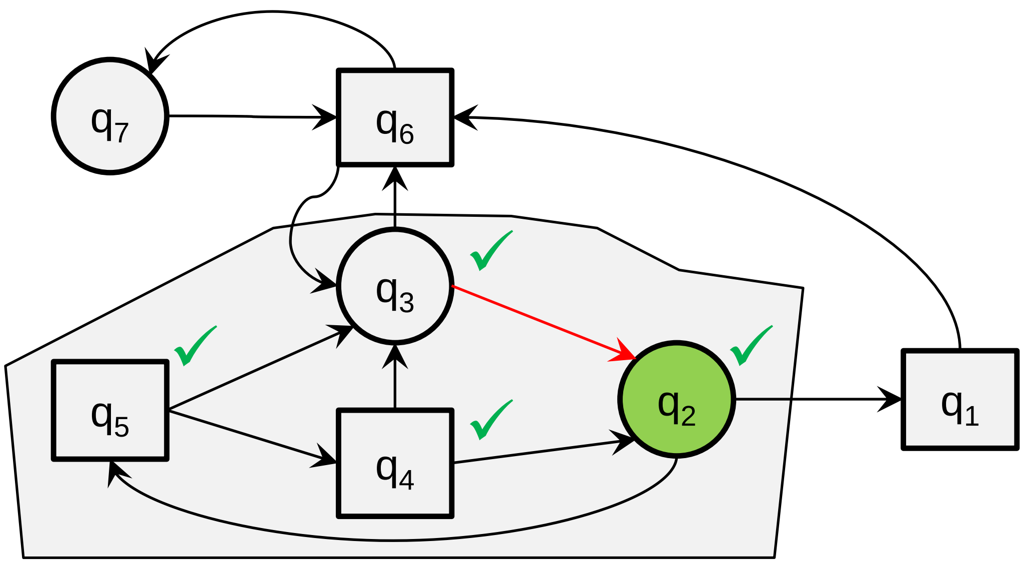


If we start from q_5, we must end up in the winning region
➔
q_5 is part of the winning region

- For the reachability game with $F = \{q_2\}$

Reachability Game

- Compute the winning region of Player 0 for:



We cannot enforce going into the winning region from other states

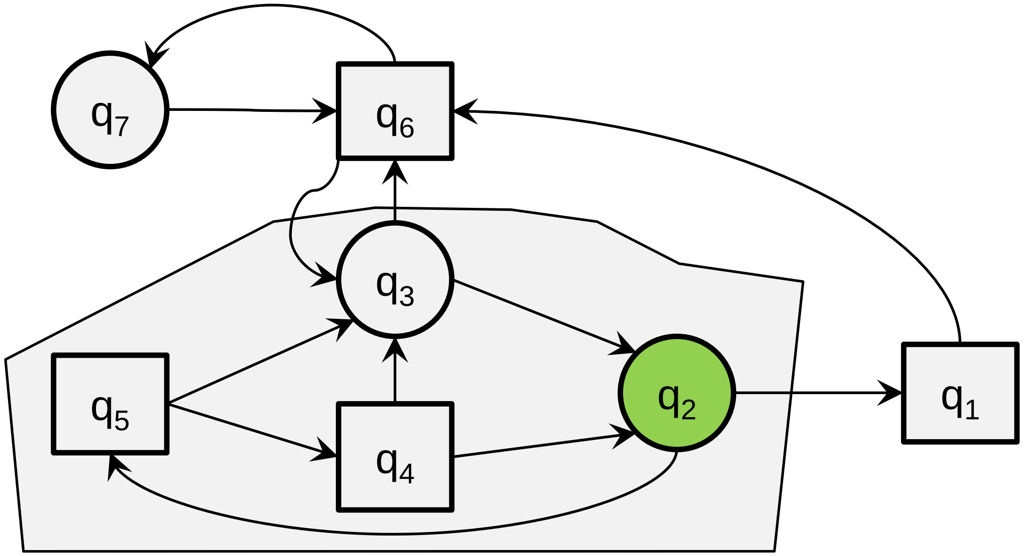
➔

We are done! The winning region is $\{q_2,q_3,q_4,q_5\}$

- For the reachability game with $F = \{q_2\}$

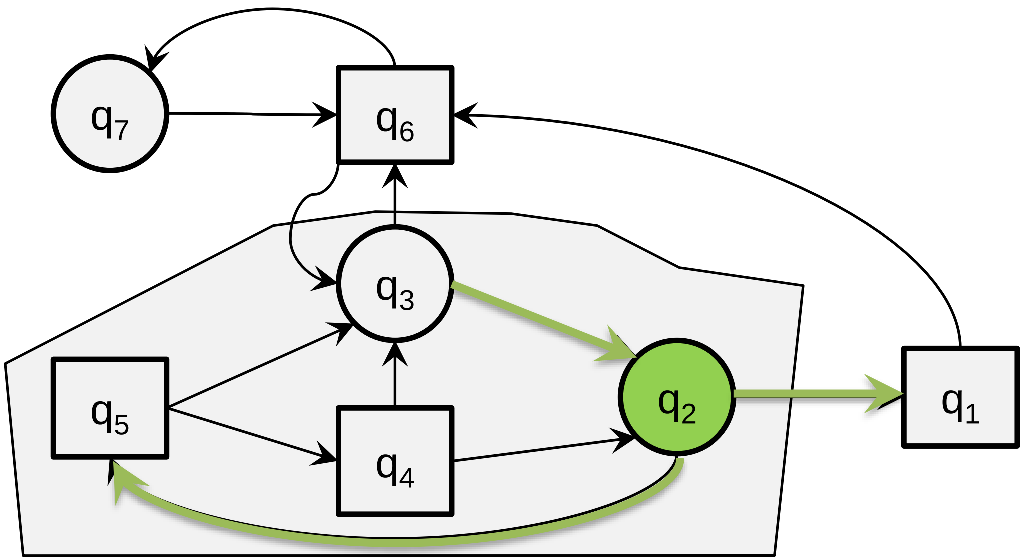
Winning Strategy

- Winning Strategy for Player 0?



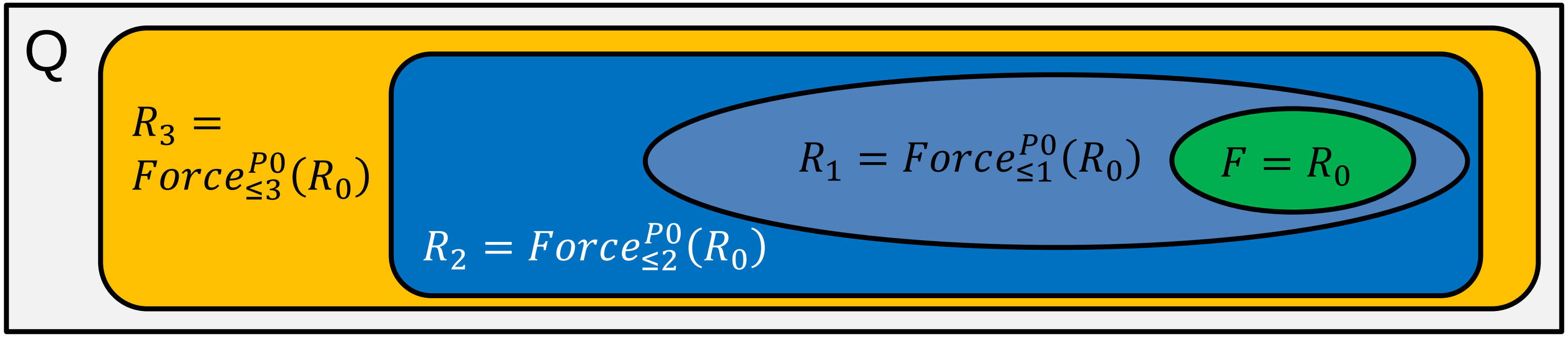
Winning Strategy

- Winning Strategy for Player 0



Reachability Game:

Winning Region

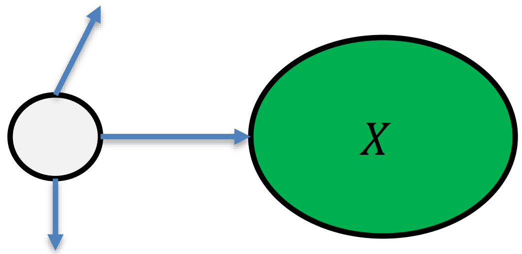


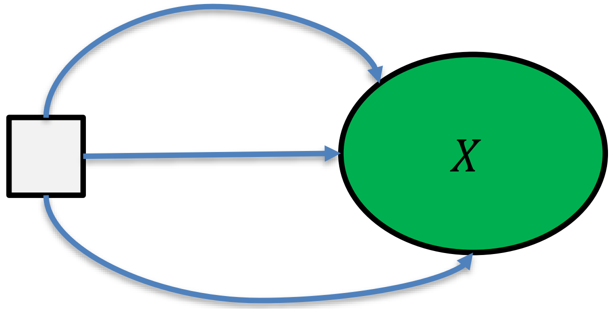
- $R_i(F)$ = $\{q \in Q$ | Player 0 can enforce to visit F in $\leq i$ steps}
- $R_\infty(F) = W = \{q \in Q$ | Player 0 can enforce to visit F}

Reachability Game:

Winning Region

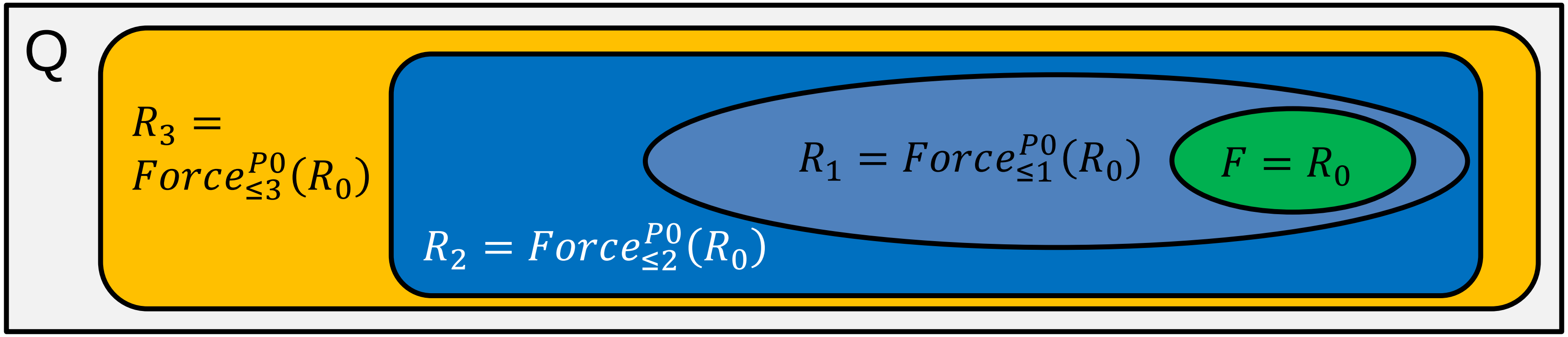
- $\mathrm{Force}_1^{P0}(X)$ = $\{q \in Q$ | Player 0 can force to reach X in exactly 1 step$\}$
- $\mathrm{Force}_1^{P0}(X) = \{q \in Q_0 | \exists (q, q') \in E: q' \in X\} \cup$
  $\{q \in Q_1 | \forall (q, q') \in E: q' \in X\}$





Reachability Game:

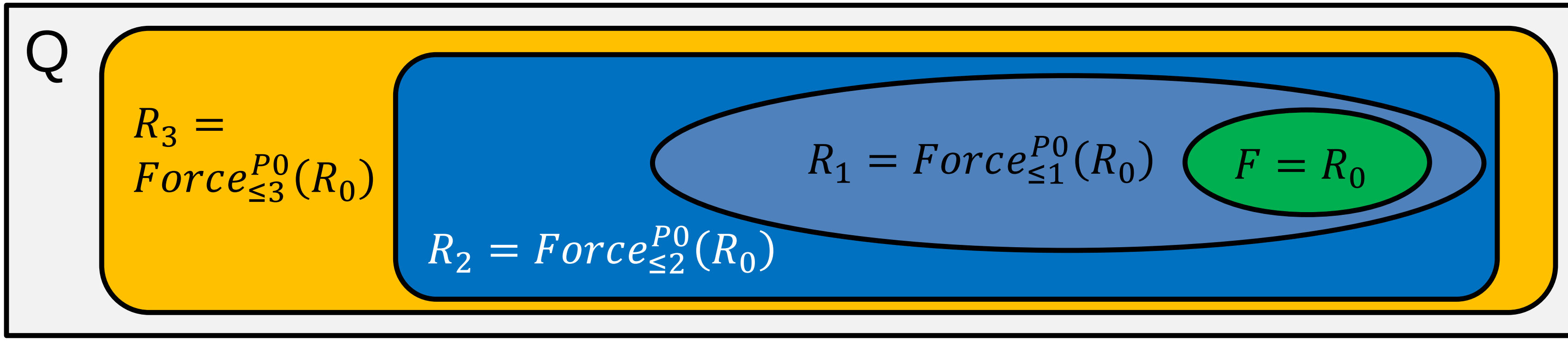
Winning Region



- $R_i(F) = \{q \in Q$ | Player 0 can enforce to visit F in $\leq i$ steps$\}$
- $R_\infty(F) = W = \{q \in Q$ | Player 0 can enforce to visit F$\}$
- Algorithm to compute W

Reachability Game:

Winning Region

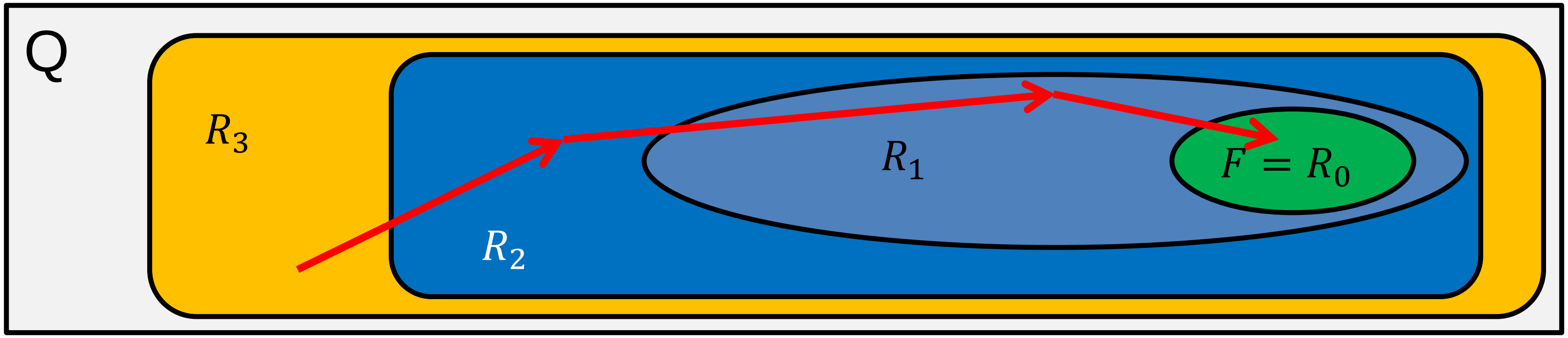


- $R_i(F) = \{q \in Q \mid$ Player 0 can enforce to visit F in $\leq i$ steps$\}$
- $R_\infty(F) = W = \{q \in Q \mid$ Player 0 can enforce to visit F$\}$
- Algorithm to compute W

```
W {
  R = {F}
  while(R changes)
    R = R ∪ Force_1^P0 (R)
return R
}
```

Reachability Game:

Winning Strategy

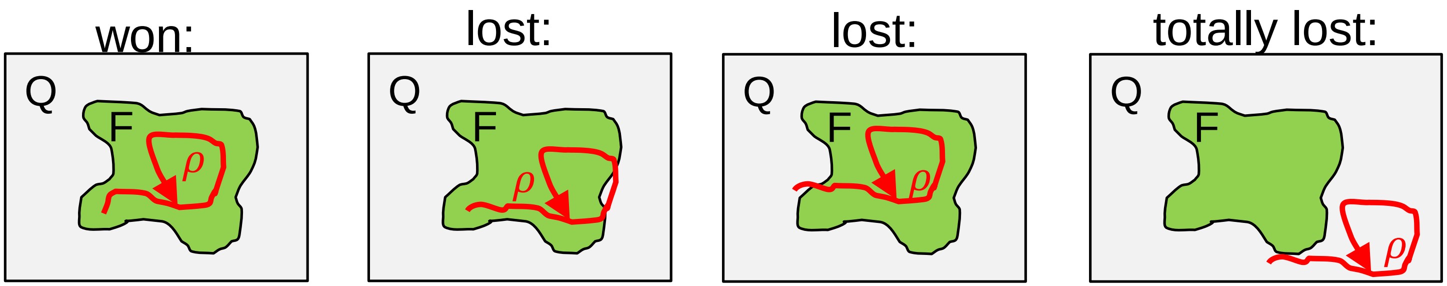


From $R_i \setminus R_{i-1}$ go to R_{i-1}

Winning Conditions:

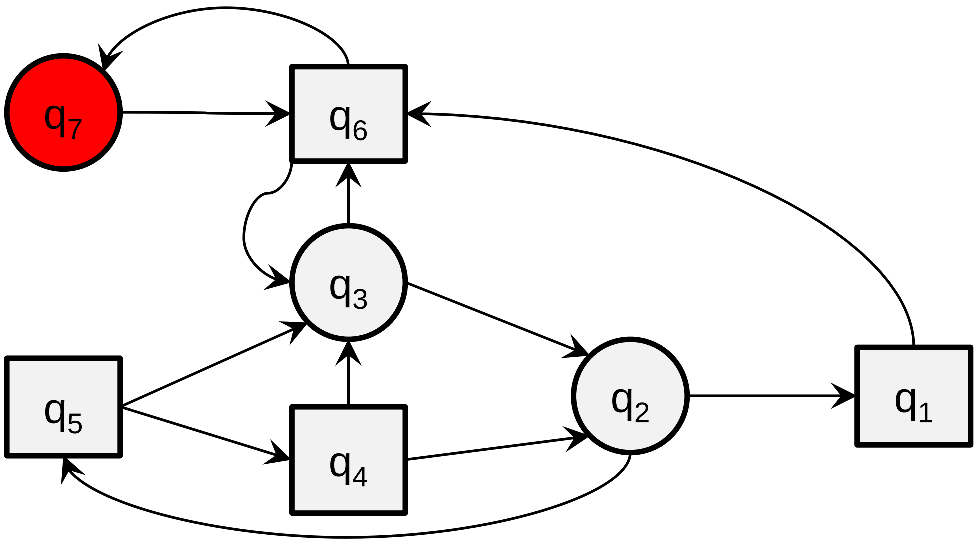
Safety Games

- set **F** of "safe states"
- Player 0 wins a play iff **it stays in F**



Safety Game

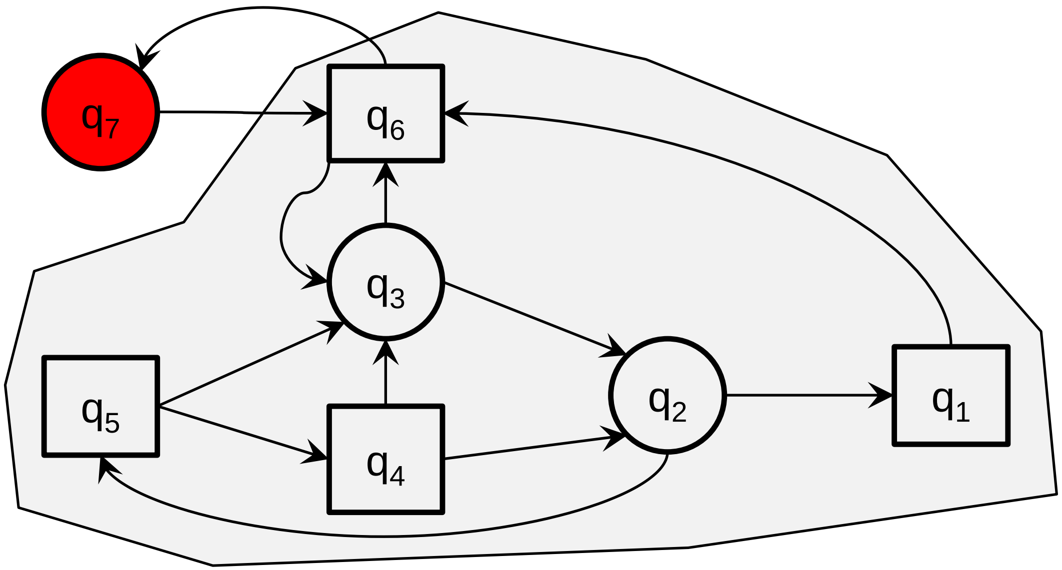
- Compute the winning region of Player 0



- For the safety game with $F = \{q_1, q_2, q_3, q_4, q_5, q_6\}$

Safety Game

- Compute the winning region of Player 0

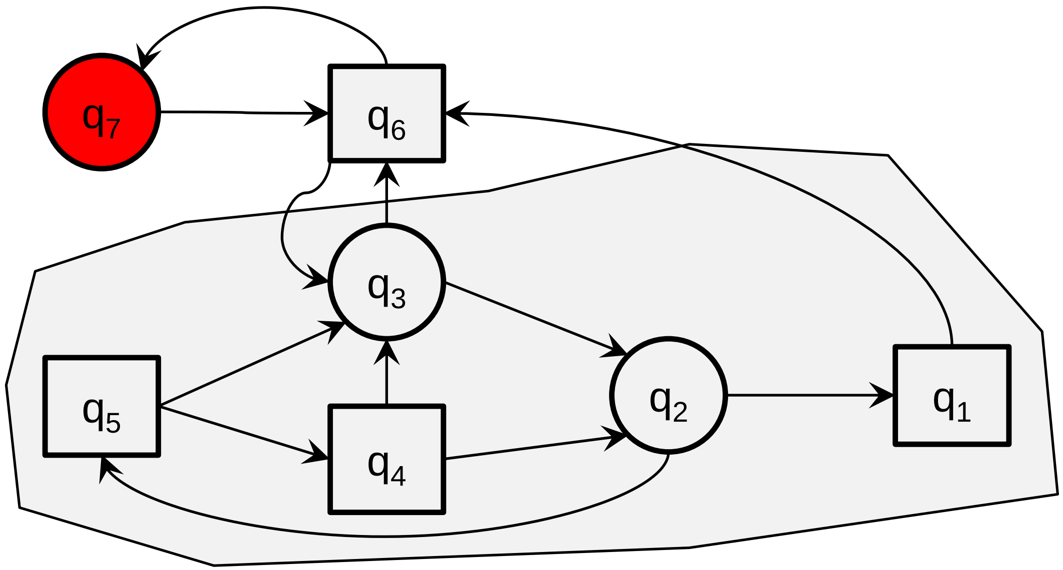


Let’s start with all safe states and see from which states we can fall out of the safe region.

- For the safety game with $F = \{q_1, q_2, q_3, q_4, q_5, q_6\}$

Exercise: Safety Game

- Compute the winning region of Player 0

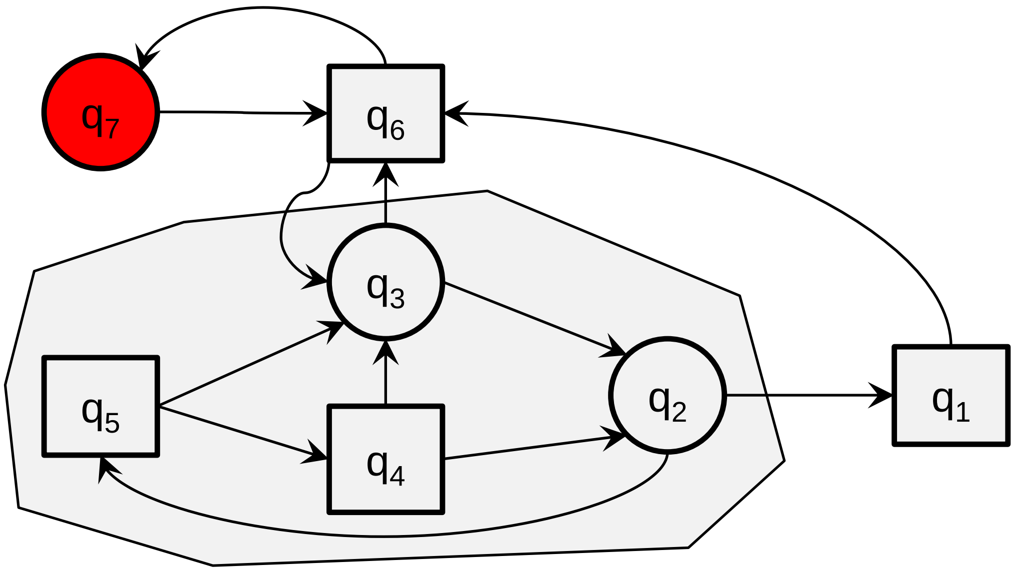


If we ever get to q_6 we are in trouble
➔
q_6 is not in the winning region

- For the safety game with $F = \{q_1, q_2, q_3, q_4, q_5, q_6\}$

Exercise: Safety Game

- Compute the winning region of Player 0

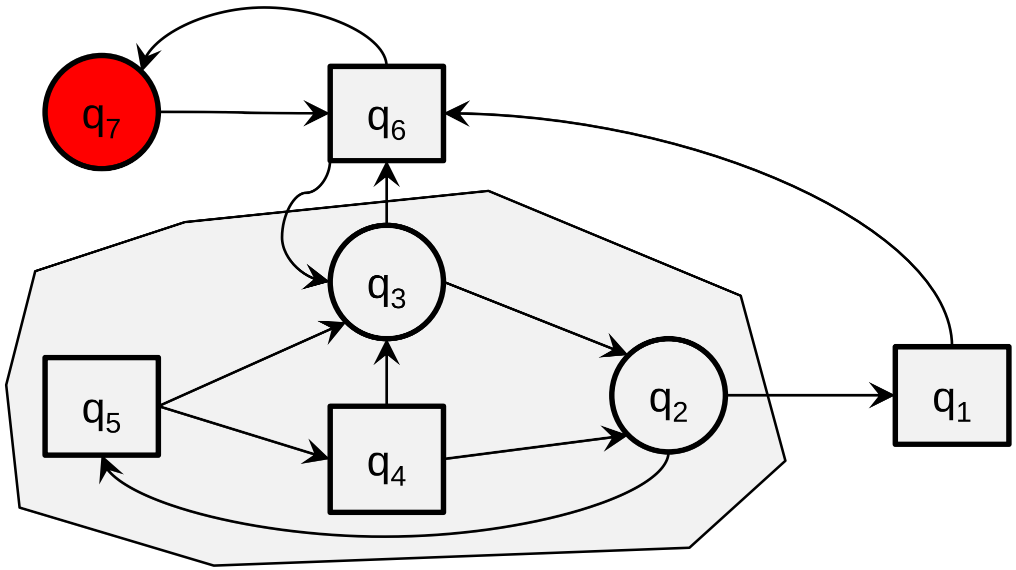


If we ever get to q_1 we are in trouble
➔
q_1 is not in the winning region

- For the safety game with $F = \{q_1, q_2, q_3, q_4, q_5, q_6\}$

Exercise: Safety Game

- Compute the winning region of Player 0



Now we can make sure that the grey area is not left

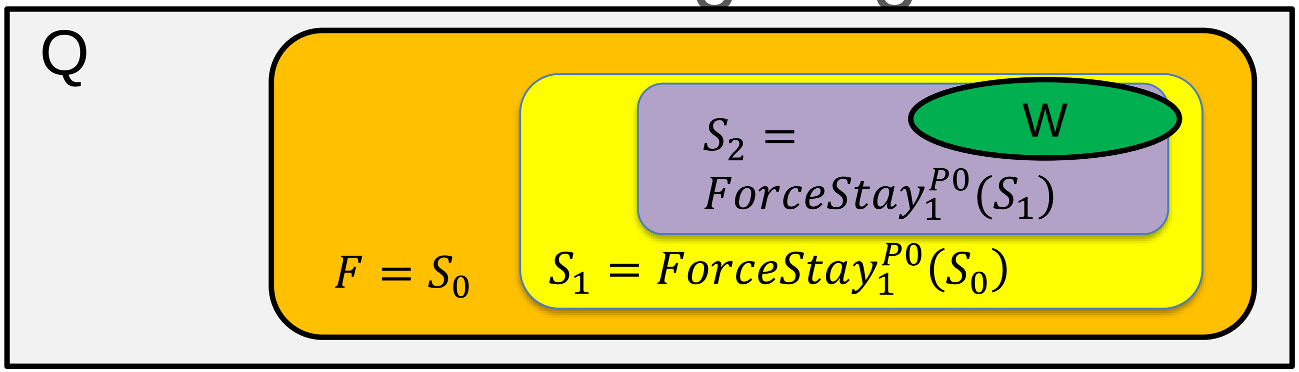
➔

We are done! The winning region is $\{q_2, q_3, q_4, q_5\}$

- For the safety game with $F = \{q_1, q_2, q_3, q_4, q_5, q_6\}$

Safety Game:

Winning Region

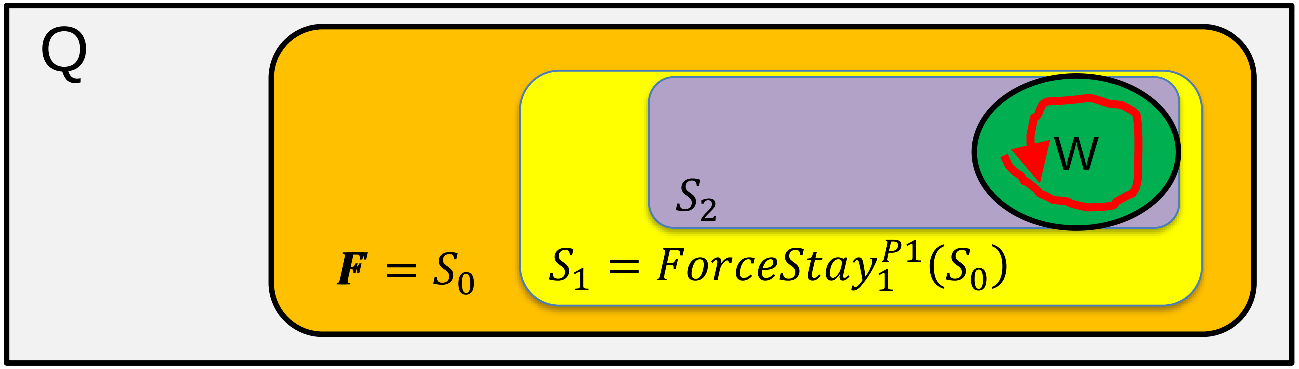


- $S_i(X) =$ $\{q \in Q$ | Player 0 can enforce to stay in F for $\geq i$ steps$\}$
- $S_\infty(X)$ =W= $\{q \in Q$ | Player 0 can enforce to stay in F forever$\}$

- Algorithm to compute W

```
W {
  S = F
  while(S changes)
    S = F ∩ Force_1^{P0} (S)
  return S
}
```

Safety Game:
Winning Strategy for Player 0

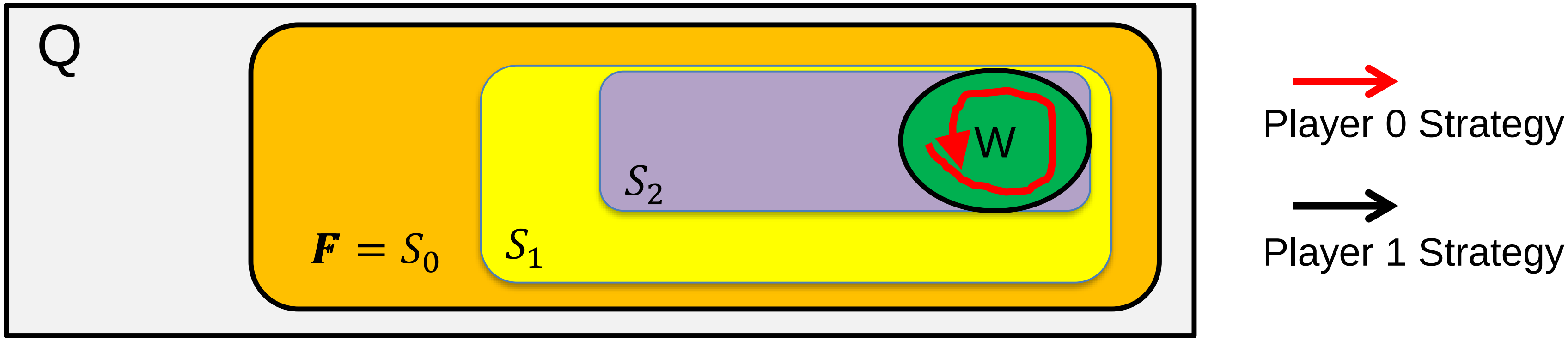


From W go to W

- In Q_0 states: pick one such edge
- In Q_1 states: Possible because W is constructed in such a way

Safety Game:

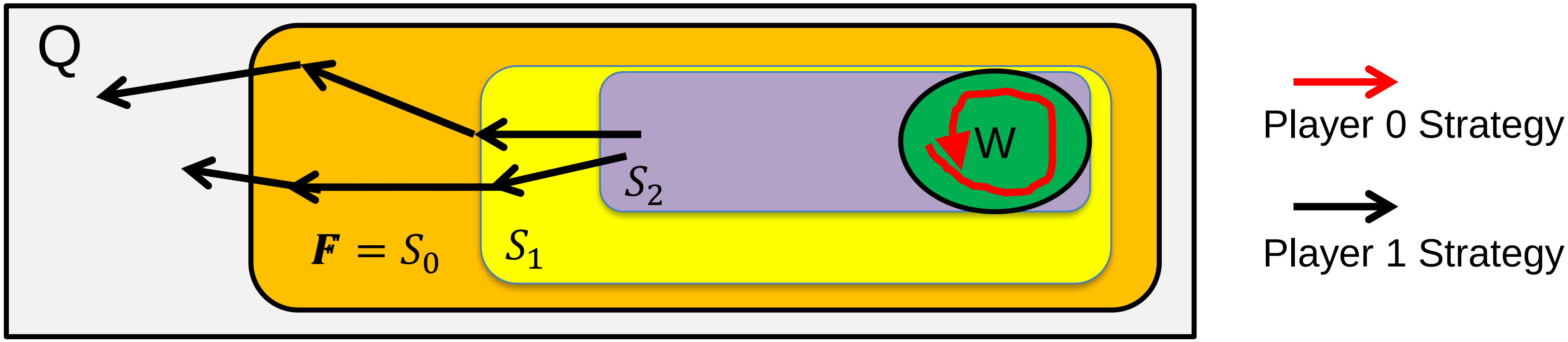
Winning Strategy For Player 0



- 

  - In which states exists a winning strategy for player 1?
  - Which game is P1 playing?
  - What is the strategy of P1?

Safety Game:
Winning Strategy For Player 0



- In which states exists a winning strategy for player 1?
  - $W_1 = Q \setminus W_0$
- Which game is P1 playing?
  - Reachability Game
- What is the strategy of P1?


THANK
YOU!