ARTIST2 - MOTIVES

Trento - Italy, February 19-23, 2007

Session on Schedulability and Controller Synthesis

Controller Synthesis

Jean-François Raskin

Université Libre de Bruxelles

Information Society Technologies

Content

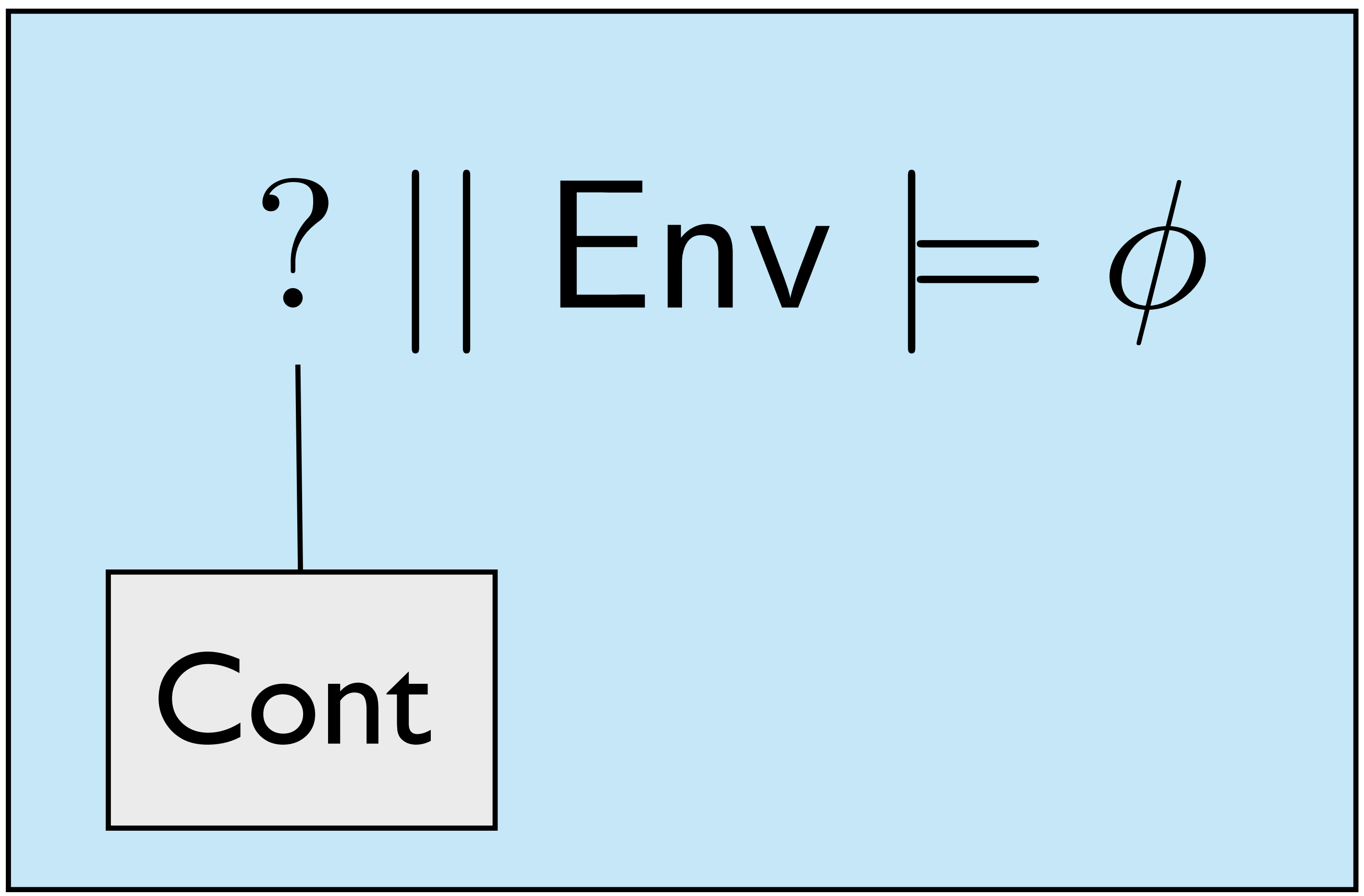
- Controller synthesis problem
- Two-player game structures
- Safety games (of perfect information)
- Imperfect information: motivations
- The lattice of antichains
- CPre over the lattice of antichains
- Application to discrete time control of RHA
- Application to the universality problem of NFA
- Conclusion & perspectives

The synthesis problem

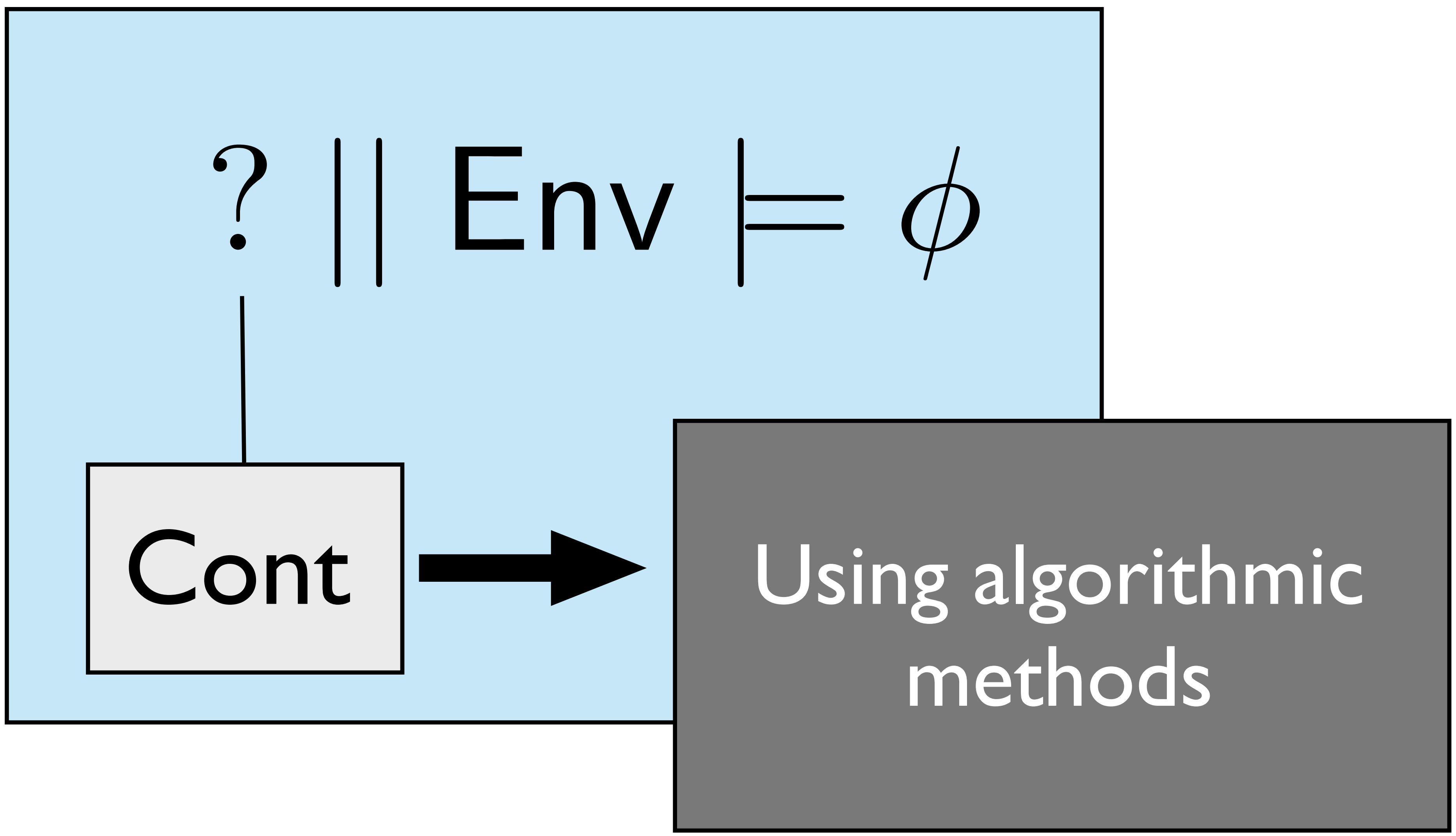
The synthesis problem

$$? \; || \; \text{Env} \models \phi$$

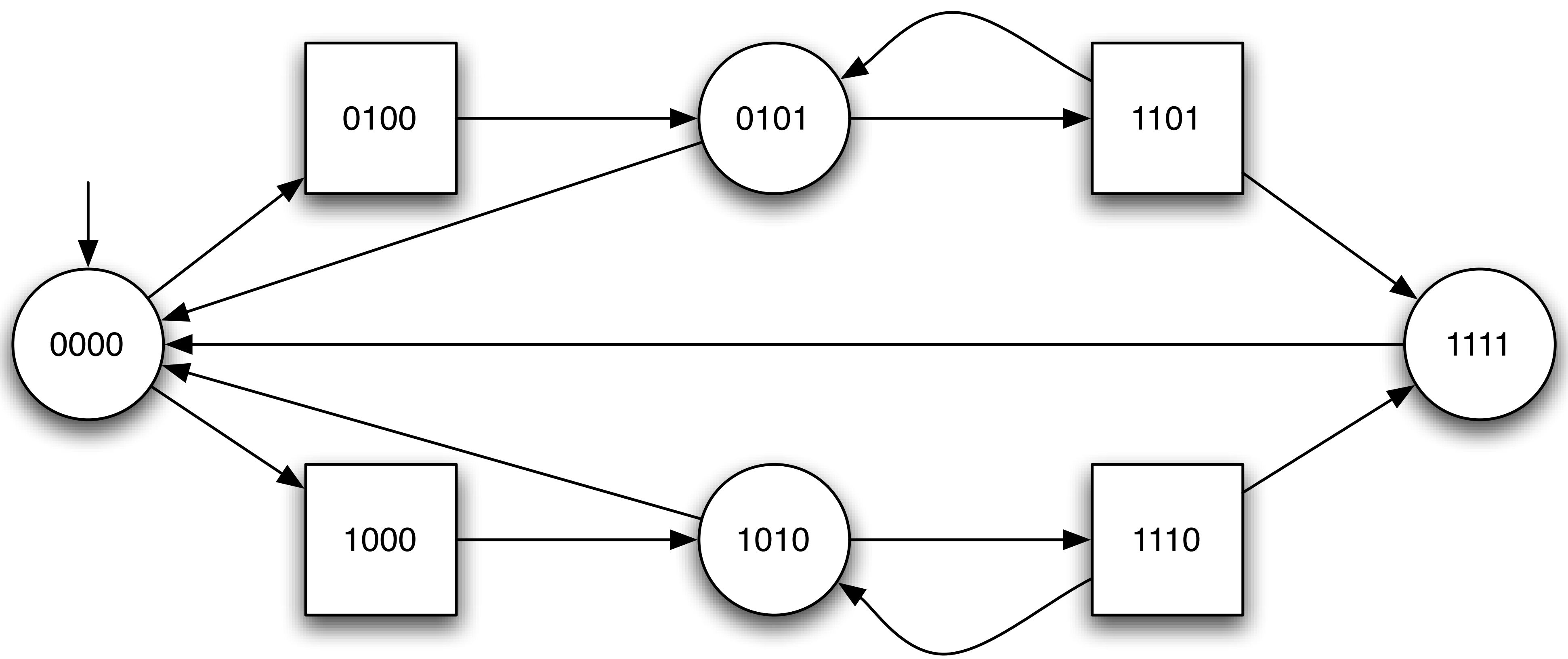
The synthesis problem

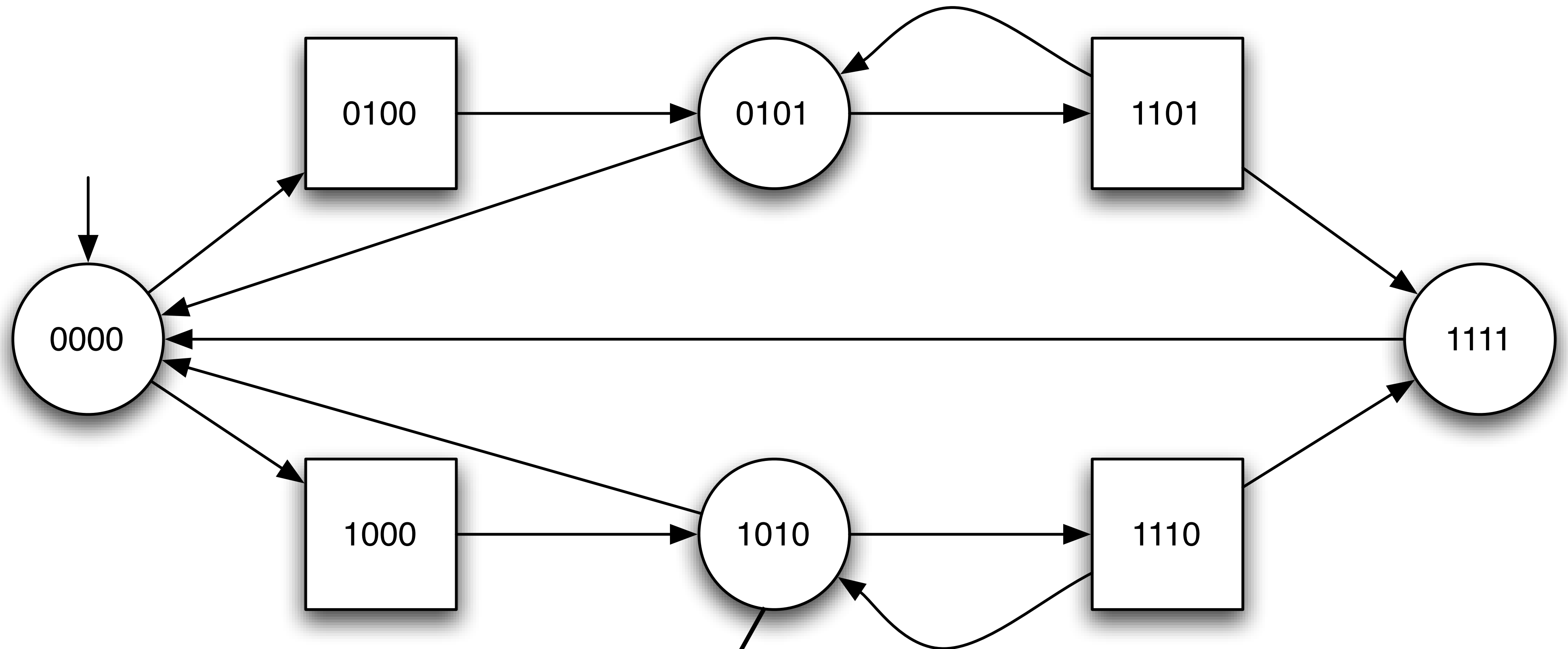


The synthesis problem



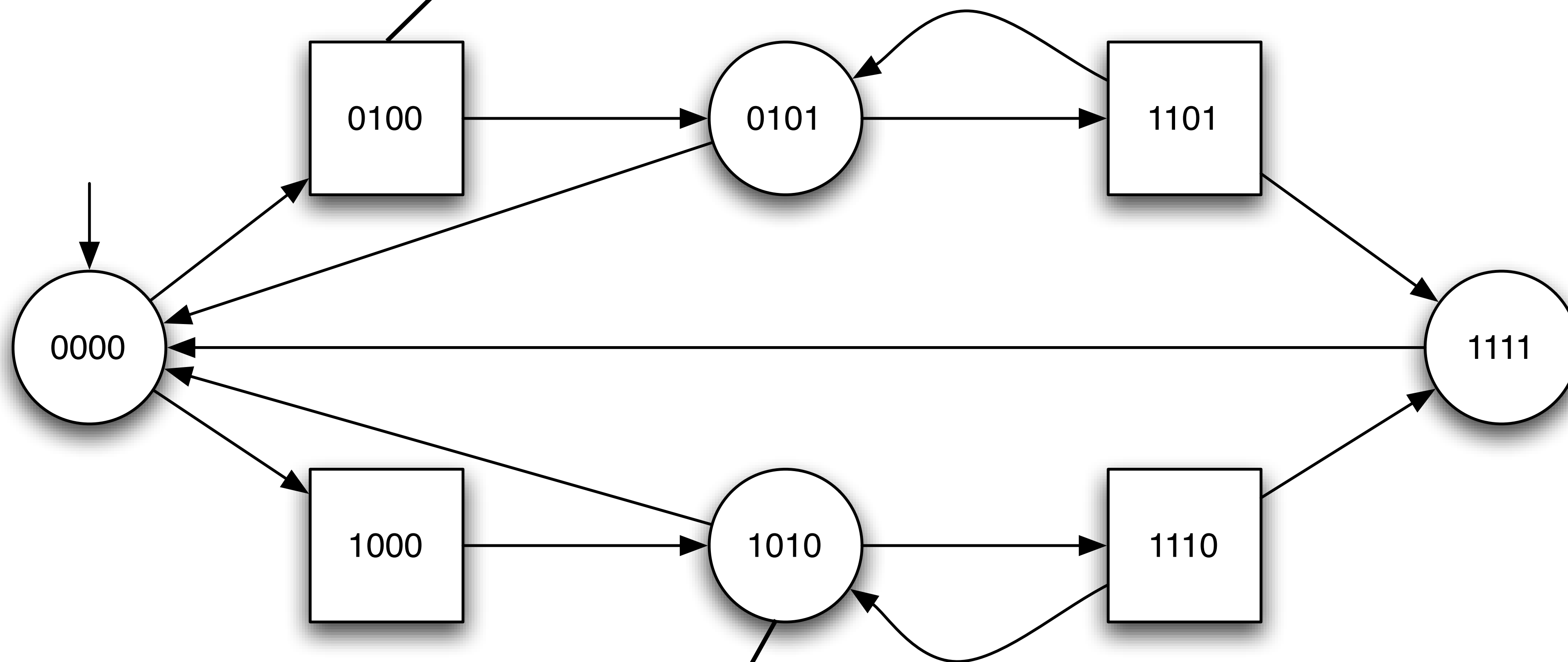
Two-player game structures


0100
0101
1101
0000
1111
1000
1010
1110



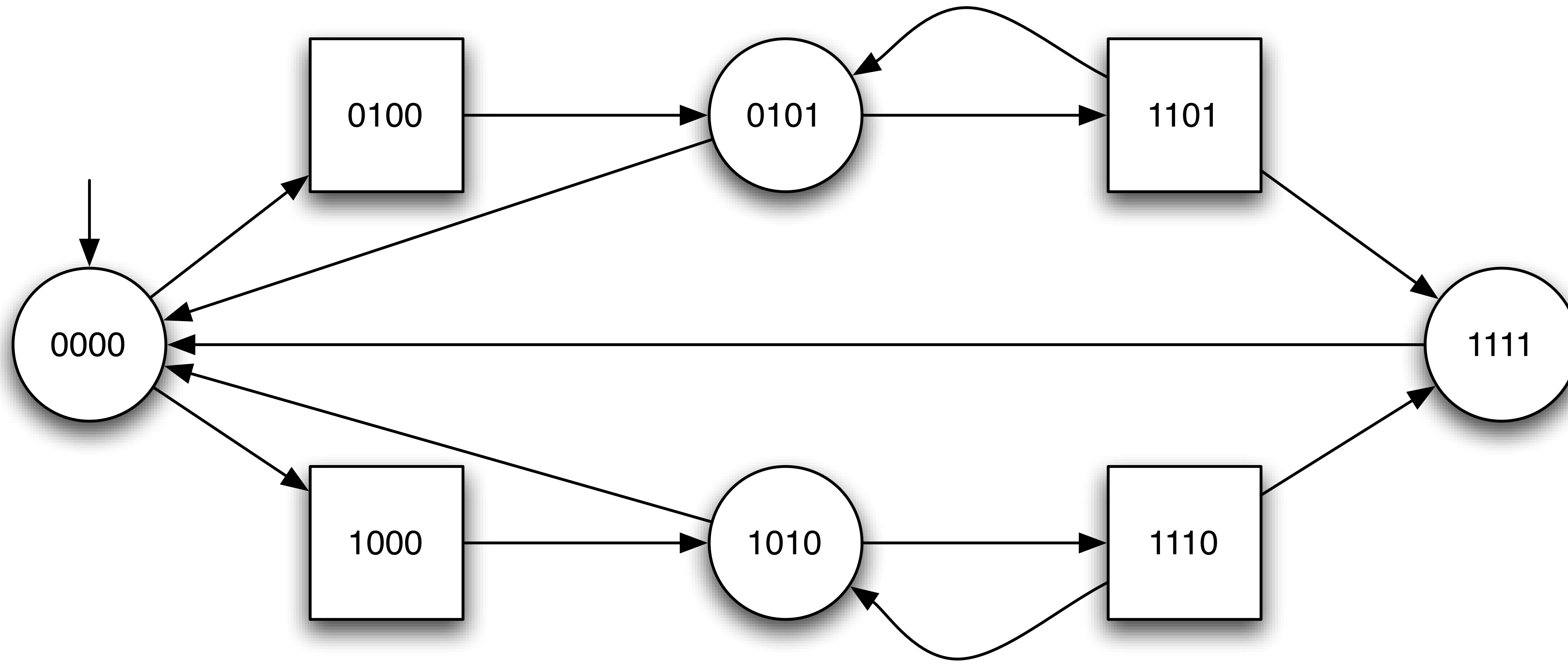
Rounded positions belong to Player 1

Square positions belong to Player 2

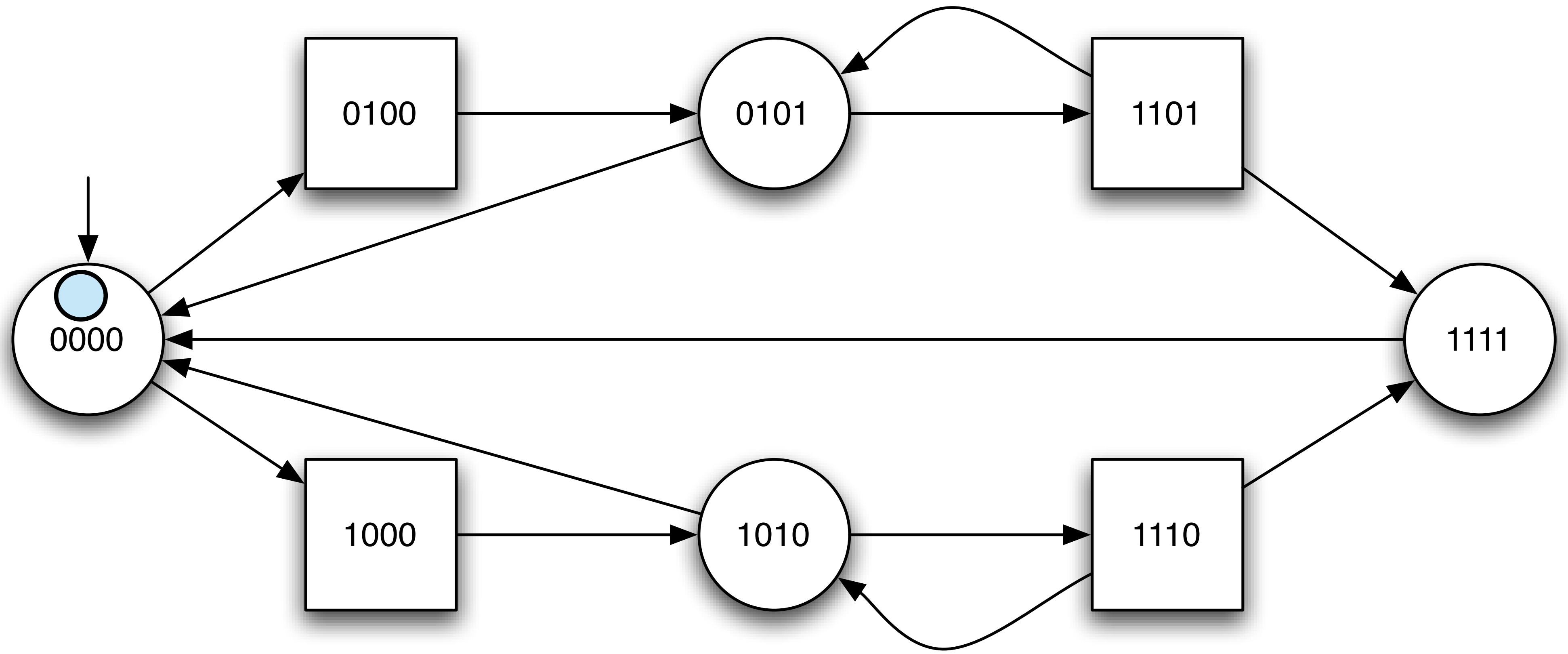


Rounded positions belong to Player 1

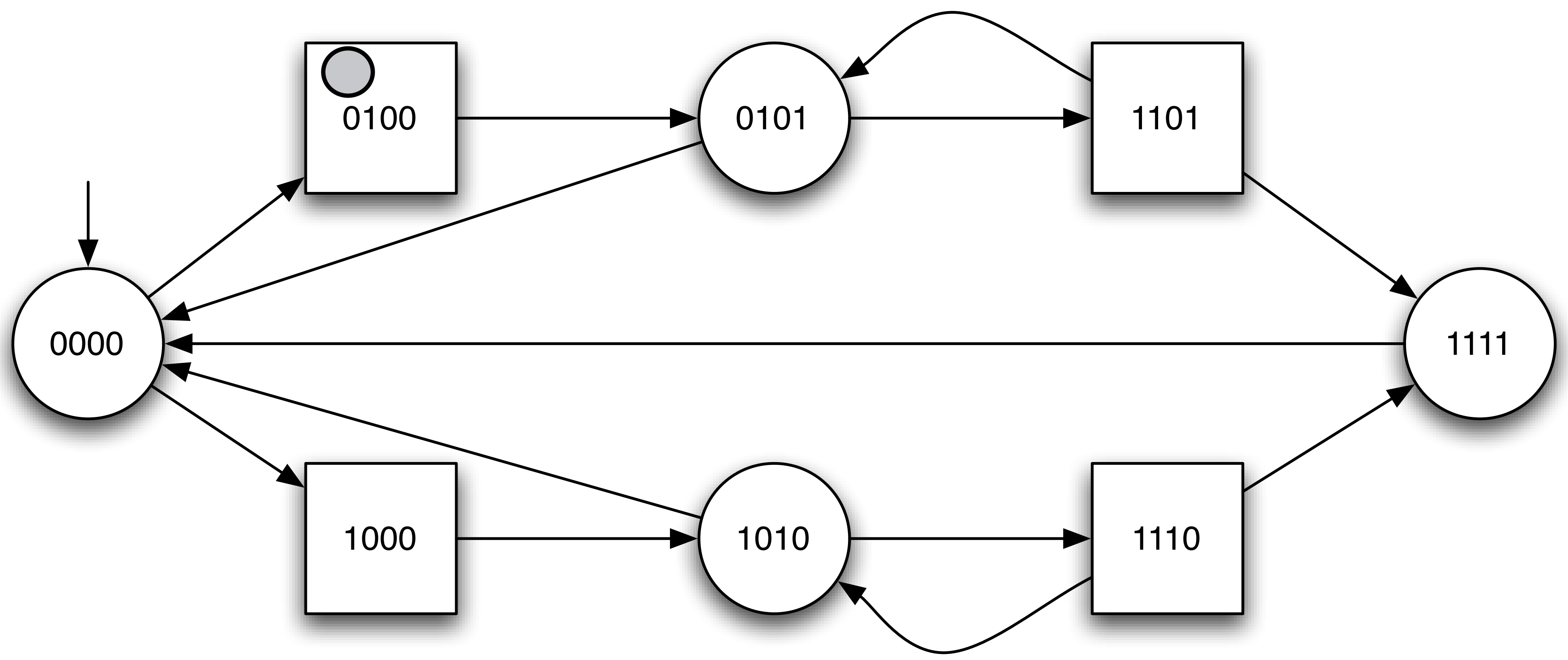
Rounded positions belong to Player I
Square positions belong to Player 2



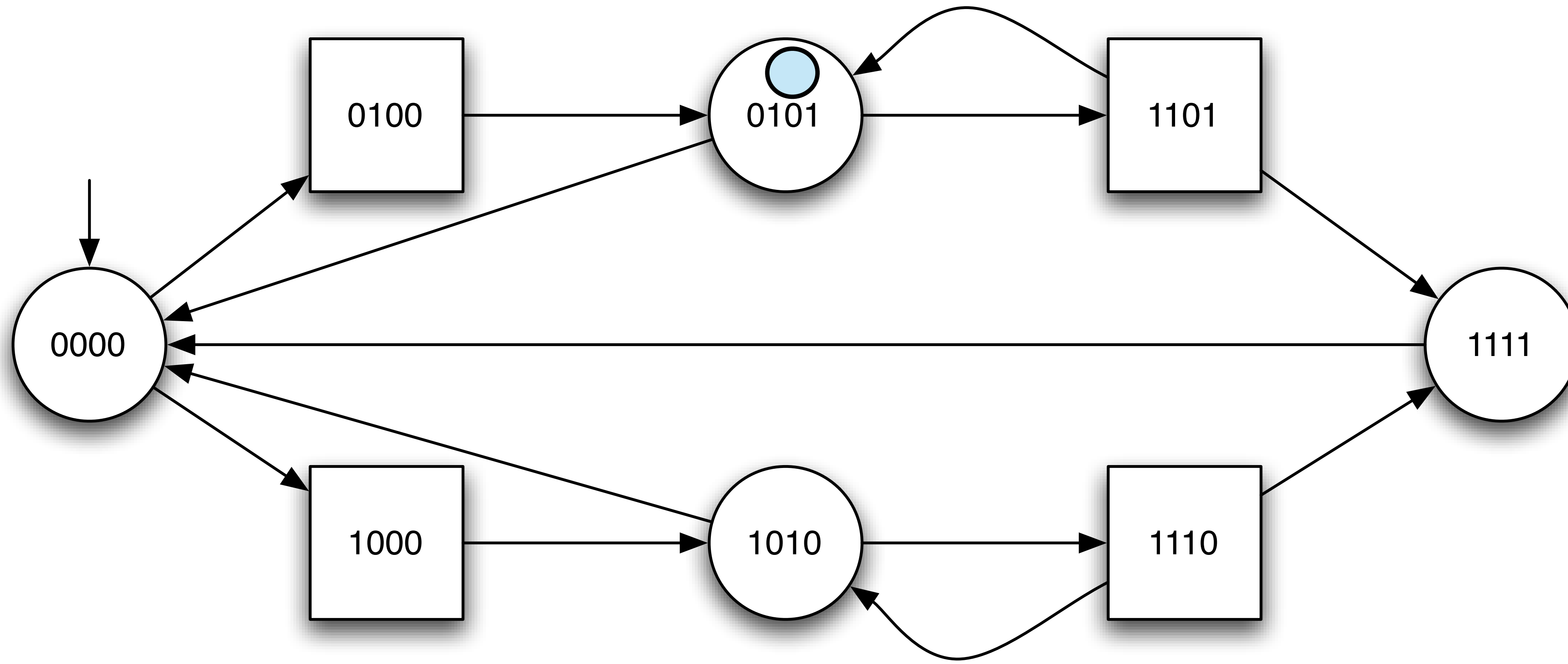
A game is played as follows: in each **round**, the game is in a **position**, if the game is in a rounded position, Player I resolves the **choice** for the next state, if the game is in a square position, Player 2 resolves the choice. The game is played for an **infinite number of rounds**.



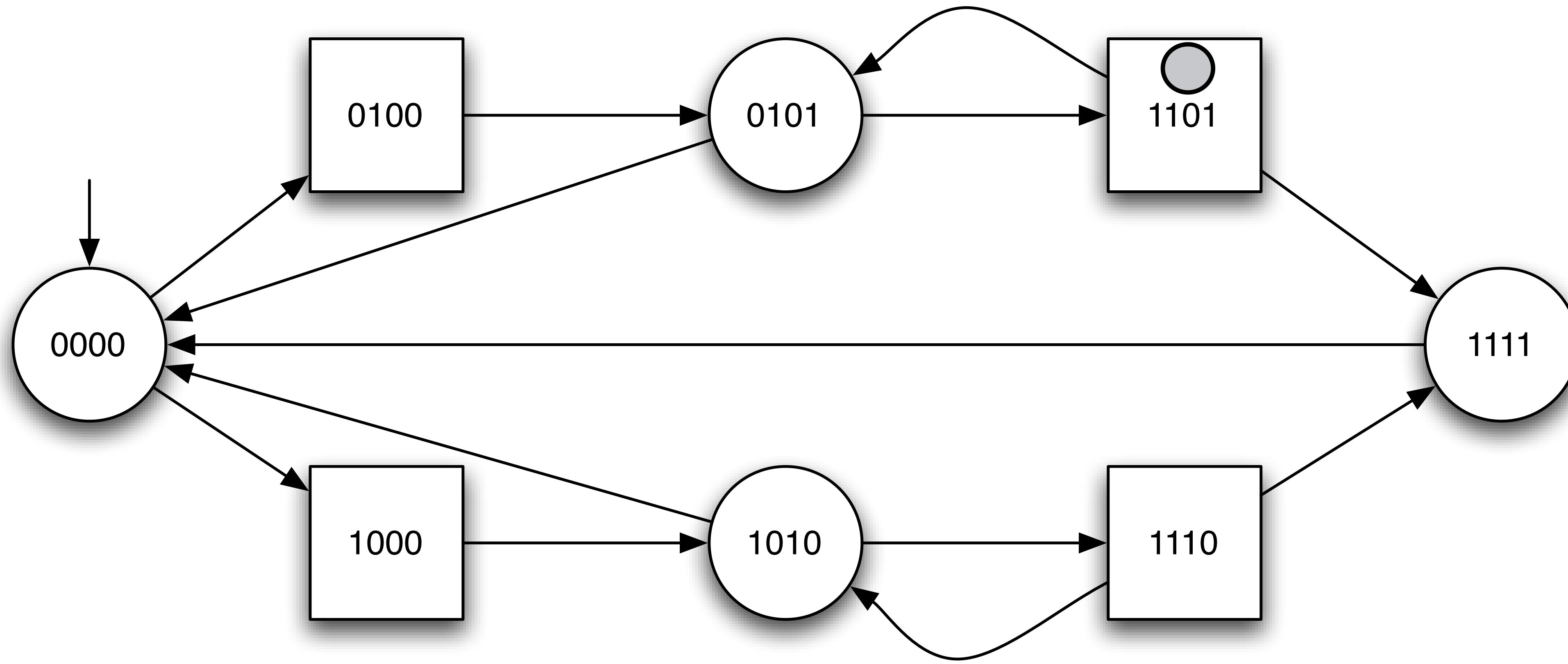
Play : 0000



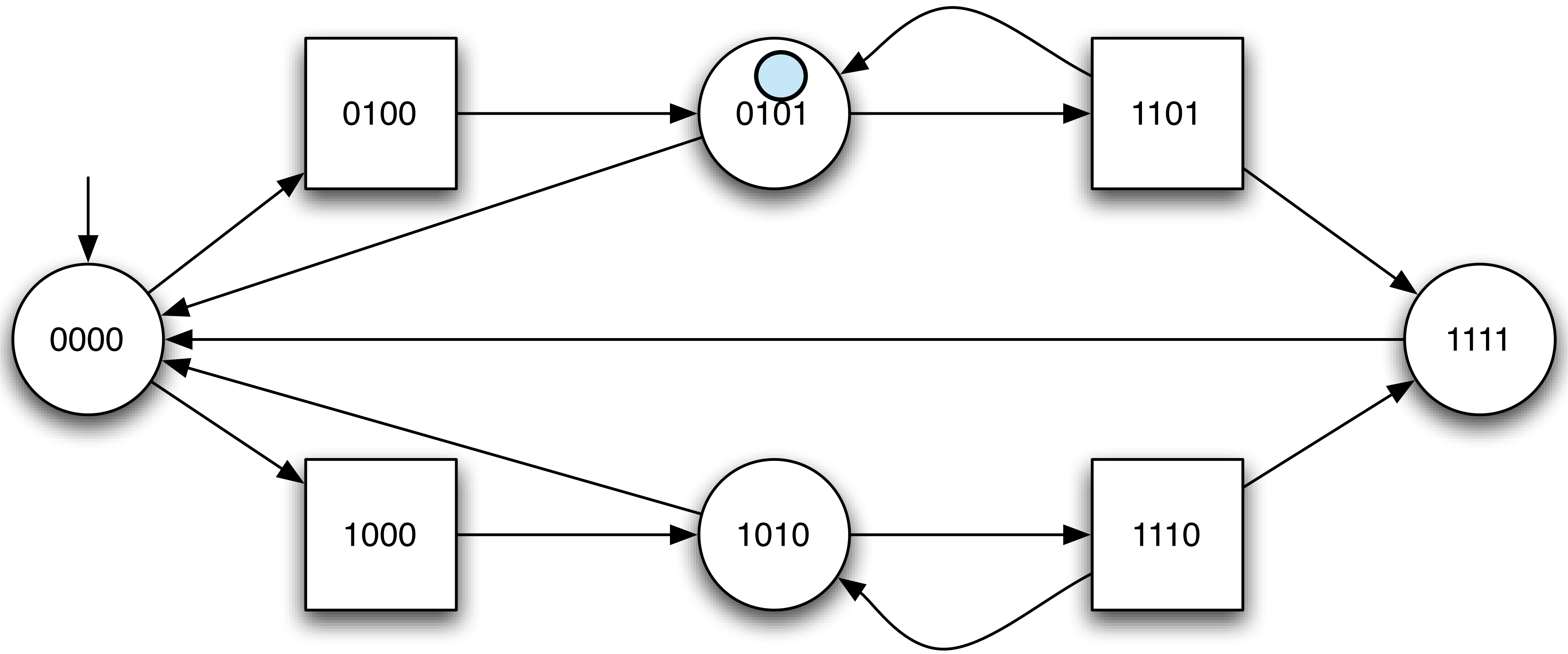
Play : 0000 0100



Play : 0000 0100 0101



Play : 0000 0100 0101 1101



Play : 0000 0100 0101 1101 ...

Two-player Game Structure

A **two-player game structure** is a tuple $G = \langle Q_1, Q_2, \iota, \delta \rangle$ where:

Q_1 and Q_2 are two (finite and) disjoint sets of **positions**

$\iota \in Q_1 \cup Q_2$ is the **initial** position of the game

$\delta \subseteq (Q_1 \cup Q_2) \times (Q_1 \cup Q_2)$ is the **transition relation** of the game

We assume that $\forall q \in Q_1 \cup Q_2 : \exists q' \in Q_1 \cup Q_2 : \delta(q, q')$

Plays, Prefixes of Plays

Let $G = \langle Q_1, Q_2, \iota, \delta \rangle$,

$w = q_0 q_1 \dots q_n \dots$ is a **play** in G if

Plays, Prefixes of Plays

Let $G = \langle Q_1, Q_2, \iota, \delta \rangle$,

$w = q_0 q_1 \ldots q_n \ldots$ is a **play** in G if

$$\forall i \geq 0 : q_i \in Q_1 \cup Q_2$$

Plays, Prefixes of Plays

Let $G = \langle Q_1, Q_2, \iota, \delta \rangle$,

$w = q_0 q_1 \dots q_n \dots$ is a **play** in G if

Notations

Let $w = q_0 q_1 \dots q_n \dots$:

$w(i)$ denotes position *i*

$w(0,i)$ denotes the prefix up to position *i*

$last(w(0,i)) = w(i)$

Plays, Prefixes of Plays

Let $G = \langle Q_1, Q_2, \iota, \delta \rangle$,

$w = q_0 q_1 \dots q_n \dots$ is a **play** in G if

1) $w(0) = \iota$
2) $\forall i \geq 0 : \delta(w(i), w(i+1))$

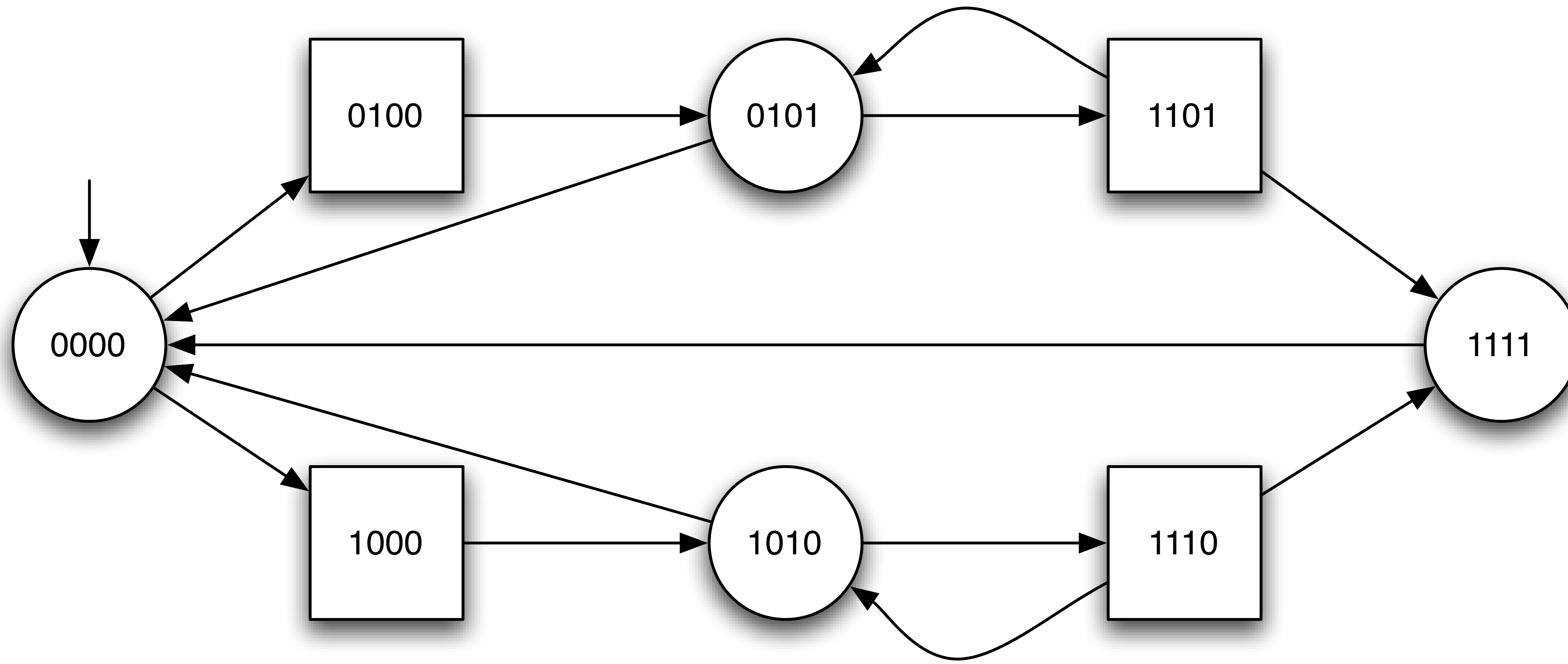
We denote the set of plays in G by : $\mathsf{Plays}(G)$

and

$$\mathsf{PrefPlays}(G) = \{q_0 q_1 \dots q_n \mid \exists w \in \mathsf{Plays}(G) \wedge \forall 0 \leq i \leq n : w(i) = q_i\}$$

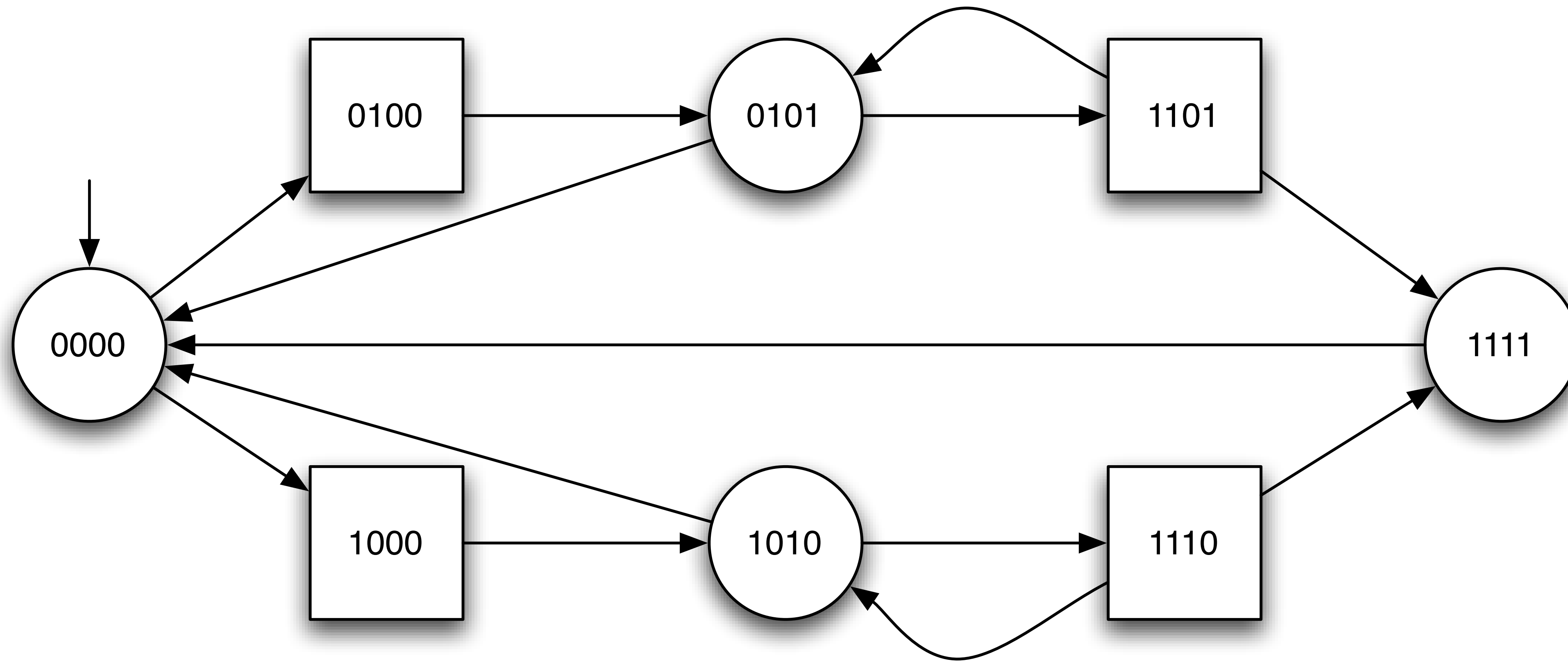
$$\mathsf{PrefPlays}_k(G) = \{w \in \mathsf{PrefPlays}(G) \wedge \mathit{last}(w) \in Q_k\}$$

Who is winning ?



Play : 0000 0100 0101 1101 ...

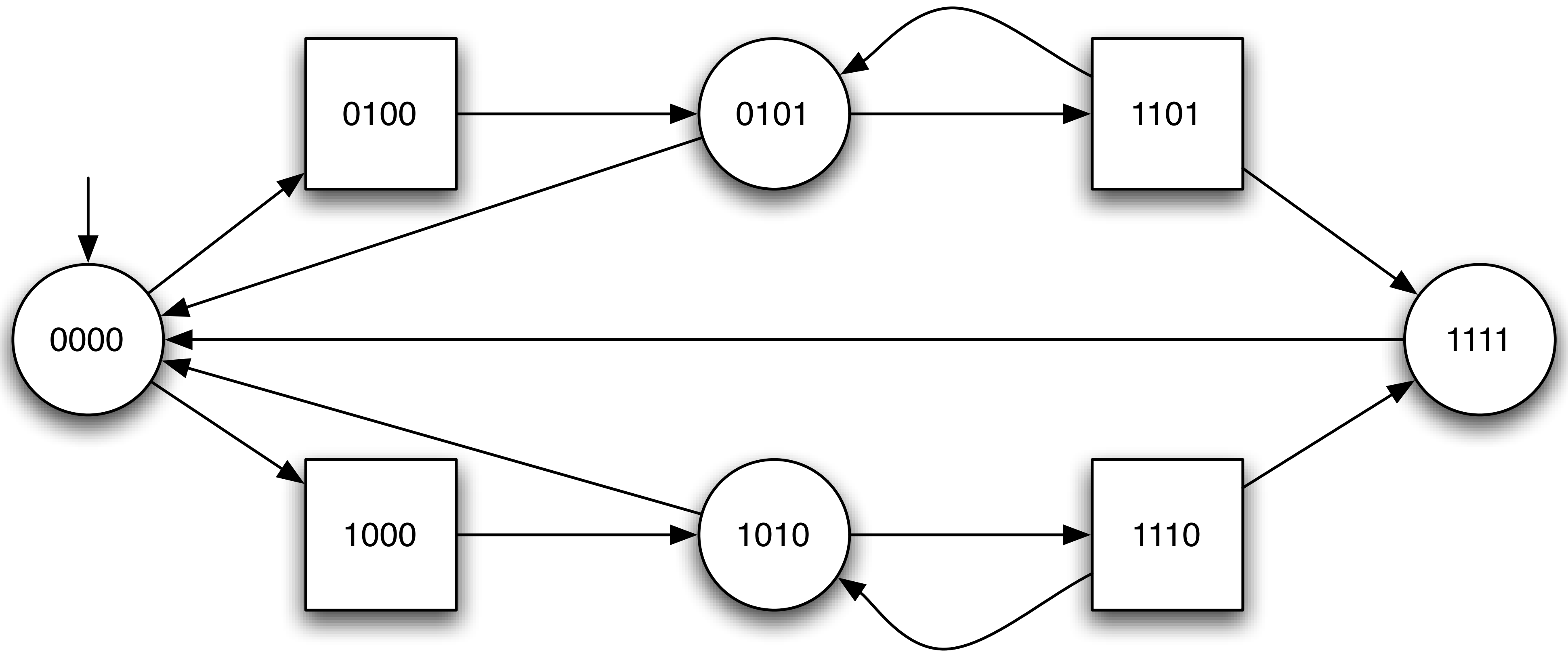
Who is winning ?



Play : 0000 0100 0101 1101 ...

Is this a **good** or a **bad** play for **Player *k*** ?

Who is winning ?



A **winning condition** (for Player *k*)
is a set of plays

$$W \subseteq (Q_1 \cup Q_2)^\omega$$

Game

=

Two-player game structure

+

Winning condition for Player *k*

Strategies

Players are playing **according to strategies.**

A **Player *k* strategy** in *G* is a function:

$$\lambda : \text{PrefPlays}_k(G) \rightarrow Q_1 \cup Q_2$$

with the restriction that:

$$\forall w \in \text{PrefPlays}_k(G) : \delta(last(w), \lambda(w))$$

Outcome of a strategy

w is a possible **outcome** of the Player *k* strategy λ if

$$\forall i \geq 0 : w(i) \in Q_k : w(i+1) = \lambda(w(0,i))$$

w is a play where Player k plays according to strategy λ

Outcome of a strategy

w is a possible **outcome** of the Player *k* strategy λ if

$$\forall i \geq 0 : w(i) \in Q_k : w(i+1) = \lambda(w(0,i))$$

The set of plays that have this property is denoted

$$\text{Outcome}_k(G, \lambda)$$

Winning strategy

- Given a pair (G, W)
- We say that Player *k* wins the game (G, W) if and only if:

$$\exists \lambda : \mathsf{Outcome}_k(G, \lambda) \subseteq W$$

Winning strategy

- Given a pair (G, W)
- We say that Player *k* wins the game (G, W) if and only if:

$$\exists \lambda : \text{Outcome}_k(G, \lambda) \subseteq W$$

That is, no matter how the other player resolves his choices, when player *k* **plays according to** λ, the resulting play belongs to *W*. Player *k* can **force** the play to be in *W*.

Winning strategy

- Given a pair (G, W)
- We say that Player *k* wins the game (G, W) if and only if:

$$\exists \lambda : \text{Outcome}_k(G, \lambda) \subseteq W$$

We say λ that is a **winning strategy** for player *k* in the game (G, W)

Winning strategies

=

Controllers that enforce winning plays

Safety Games

Safety Game

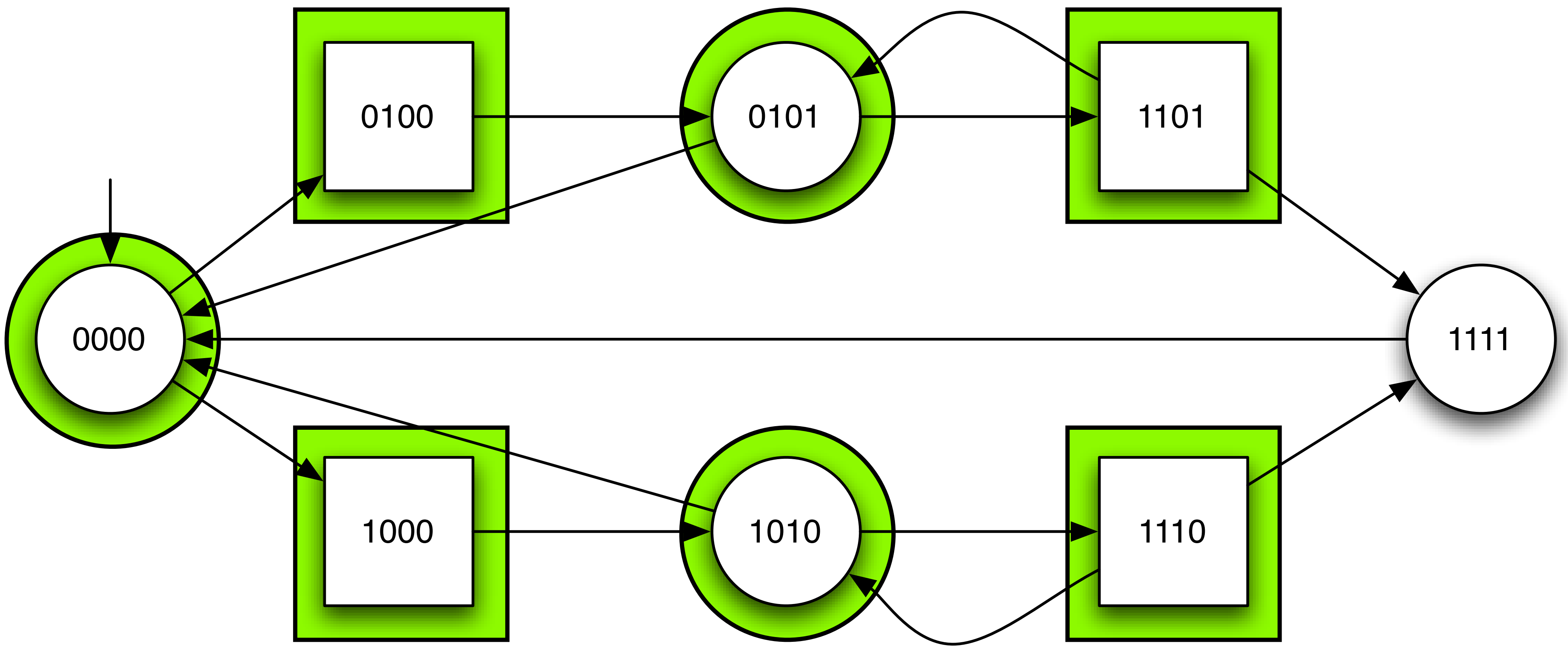
(G, W) is a **safety game** if

$$\exists Q \subseteq Q_1 \cup Q_2 : W = \{w \in \mathsf{Plays}(G) \mid \forall i \geq 0 : w(i) \in Q\}$$

That is *W* is the set of plays that stay within a given set of positions *Q*.

$\mathsf{Safe}(G, Q)$

A Safety Game



Does Player I, who owns the rounded positions, have a strategy (against any choices of Player II) to stay within the set of states

$Q \setminus \{1111\}$?

Symbolic algorithms to solve games

Complete lattices

A **complete lattice** is a partially ordered set $(L,\leq)$ where every subset of L has a **least upper bound** (often called join or supremum) and a **greatest lower bound** (often called meet or infimum).

Given $M \subseteq L$, **lub**(M) is a value of L such that :

(i) for all $m \in M$: $m \leq$ **lub**(M) and

(ii) for all $m' \in L$,

if for all $m \in M$: $m \leq m'$ then **lub**(M) $\leq m'$

Given $M \subseteq L$, **glb**(M) is a value of L such that :

(i) for all $m \in M$: **glb**(M) $\leq m$ and

(ii) for all $m' \in L$,

if for all $m \in M$: $m' \leq m$ then $m' \leq$ **glb**(M)

Example of complete lattice

2^S, the set of subsets of a set *S*, ordered by set inclusion $\subseteq$

forms a complete lattice.

Its *least upper bound* is given by union :

$$\mathbf{lub}\{S_1, S_2, \ldots, S_n\} = \cup\{S_1, S_2, \ldots, S_n\}$$

Its *greatest lower bound* is given by intersection :

$$\mathbf{glb}\{S_1, S_2, \ldots, S_n\} = \cap\{S_1, S_2, \ldots, S_n\}$$

The *least element* of the lattice is $\emptyset$ and the *largest element* is *S*.
The powerset complete lattice is noted

$$\langle 2^S, \subseteq, \cup, \cap, S, \emptyset \rangle$$

Monotone functions and fixed points

Let $\langle L, \sqsubseteq, \sqcup, \sqcap, \top, \bot \rangle$ be a complete lattice, let $f : L \rightarrow L$.
We say that *f* is **monotone** iff

$$\forall l_1, l_2 \in L : l_1 \sqsubseteq l_2 \Rightarrow f(l_1) \sqsubseteq f(l_2)$$

f is **Scott- continuous** iff $\sqcup\{f(l) \mid l \in X\} = f(\sqcup X)$
for any chain *X*.
We say that *l* is a fixed point of *f* iff $l = f(l)$

Any monotone function *f* over a complete lattice *L* has:

a **least fixed point**: $\mathrm{lfp} f = \sqcap\{l \mid l = f(l)\}$

a **greatest fixed point**: $\mathrm{gfp} f = \sqcup\{l \mid l = f(l)\}$

Monotone functions and fixed points

Let $\langle L, \sqsubseteq, \sqcup, \sqcap, \top, \bot \rangle$ be a complete lattice, let $f : L \to L$.
We say that *f* is **monotone** iff

$$\forall l_1, l_2 \in L : l_1$$

f is **Scott- continuou**
for any chain X.
We say that *l* is a fixed p

Any monotone function *f*

Monotony is equivalent to Scott-continuity on any **finite** complete lattice.

a **least fixed point**: $\mathrm{lfp} f = \sqcap \{ l \mid l = f(l) \}$
a **greatest fixed point**: $\mathrm{gfp} f = \sqcup \{ l \mid l = f(l) \}$

Player *k* Controllable Predecessors

X is a set of positions

$$1\text{CPre}_G(X) = \{q \in Q_1 \mid \exists q' : \delta(q,q') \wedge q' \in X\} \cup \{q \in Q_2 \mid \forall q' : \delta(q,q') : q' \in X\}$$

Set of Player I positions where she has a choice of successor that lies in *X*

Set of Player II positions where all her choices for successors lie in *X*

Player *k* Controllable Predecessors

$$\mathsf{1CPre}_G(X) = \{q \in Q_1 \mid \exists q' : \delta(q,q') \wedge q' \in X\} \cup \{q \in Q_2 \mid \forall q' : \delta(q,q') : q' \in X\}$$

Symmetrically

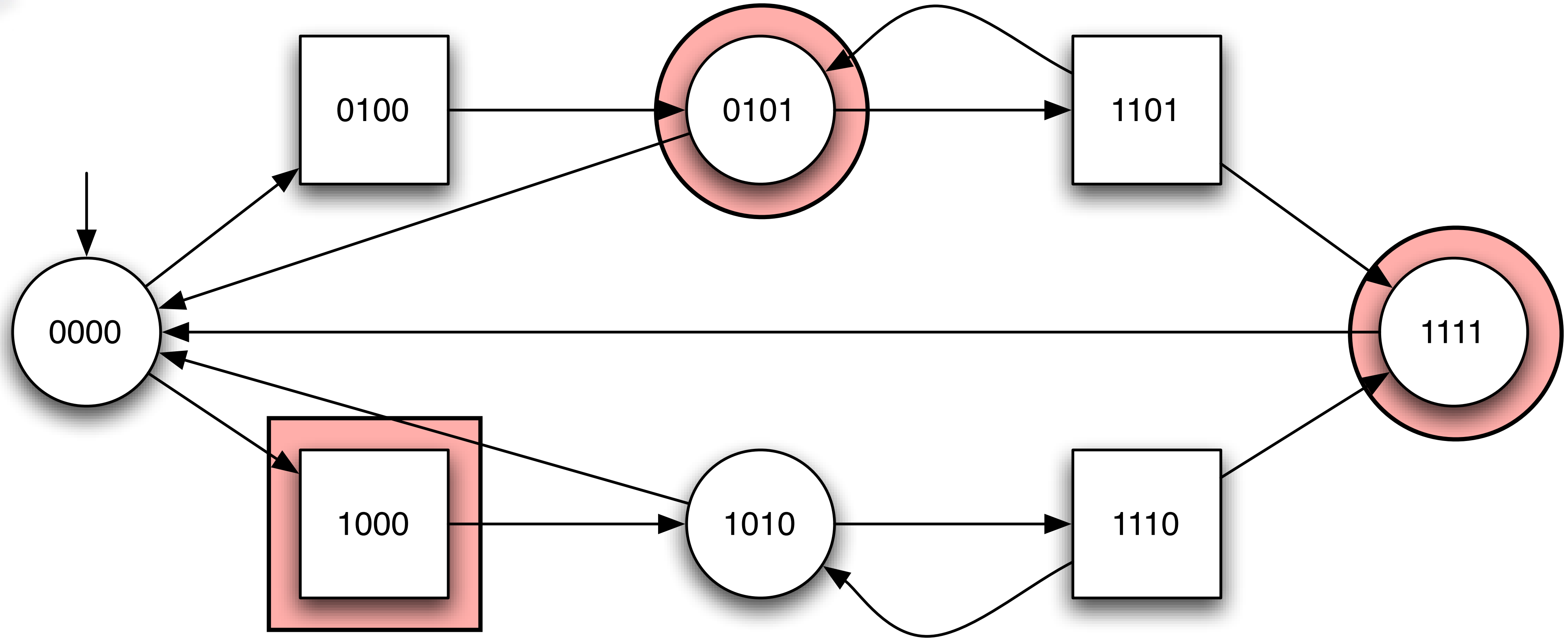
$$\mathsf{2CPre}_G(X) = \{q \in Q_2 \mid \exists q' : \delta(q,q') \wedge q' \in X\} \cup \{q \in Q_1 \mid \forall q' : \delta(q,q') : q' \in X\}$$

Player *k* Controllable Predecessors

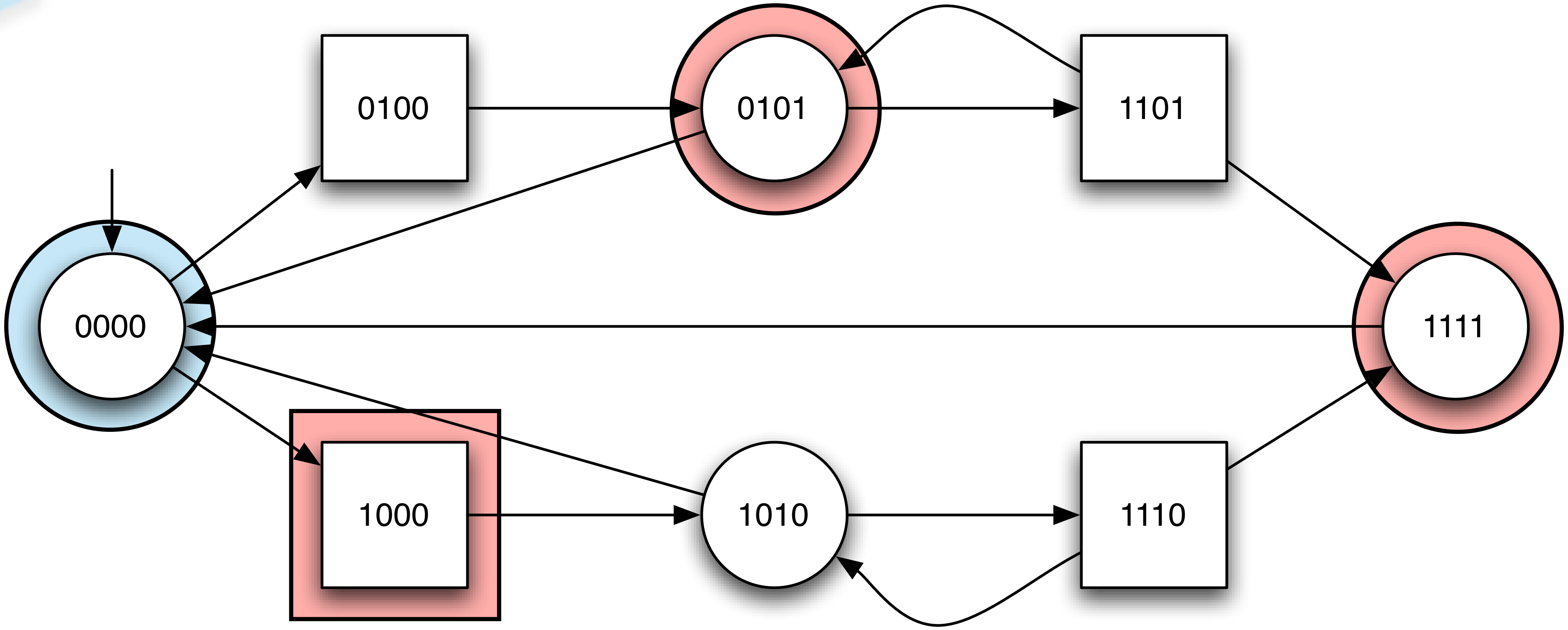
$$\mathsf{1CPre}_G(X) = \{q \in Q_1 \mid \exists q' : \delta(q,q') \wedge q' \in X\} \cup \{q \in Q_2 \mid \forall q' : \delta(q,q') : q' \in X\}$$

Monotonic functions over $\langle 2^{Q_1 \cup Q_2}, \subseteq \rangle$

$$\mathsf{2CPre}_G(X) = \{q \in Q_2 \mid \exists q' : \delta(q,q') \wedge q' \in X\} \cup \{q \in Q_1 \mid \forall q' : \delta(q,q') : q' \in X\}$$



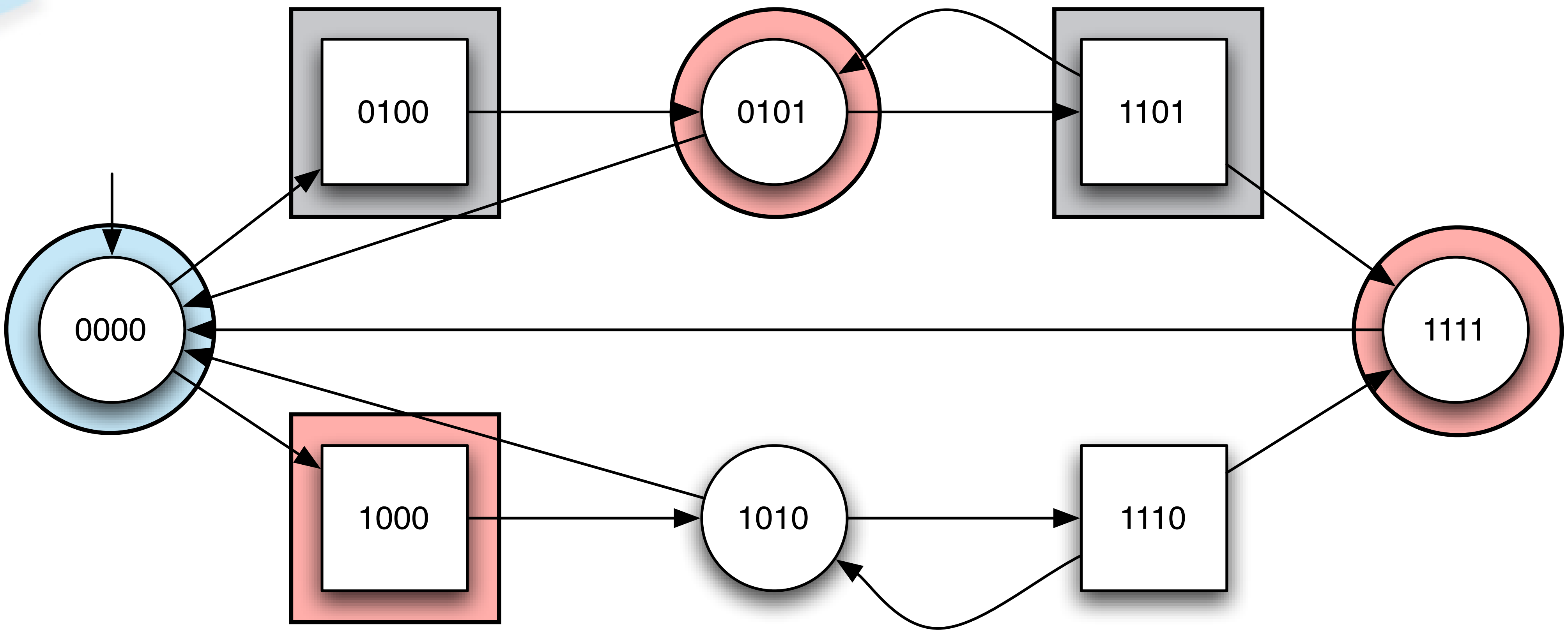
$$X = \boxed{\{1000, 0101, 1111\}}$$



$$X = \{1000, 0101, 1111\}$$

$$\text{1CPre}(X) = \{0000\} \cup \{0100, 1101\}$$

Rounded positions,
there exists a red successor



$$X = \{1000, 0101, 1111\}$$

$$1\mathsf{CPre}(X) = \{0000\} \cup \{0100, 1101\}$$

Rounded positions,
there exists a red successor

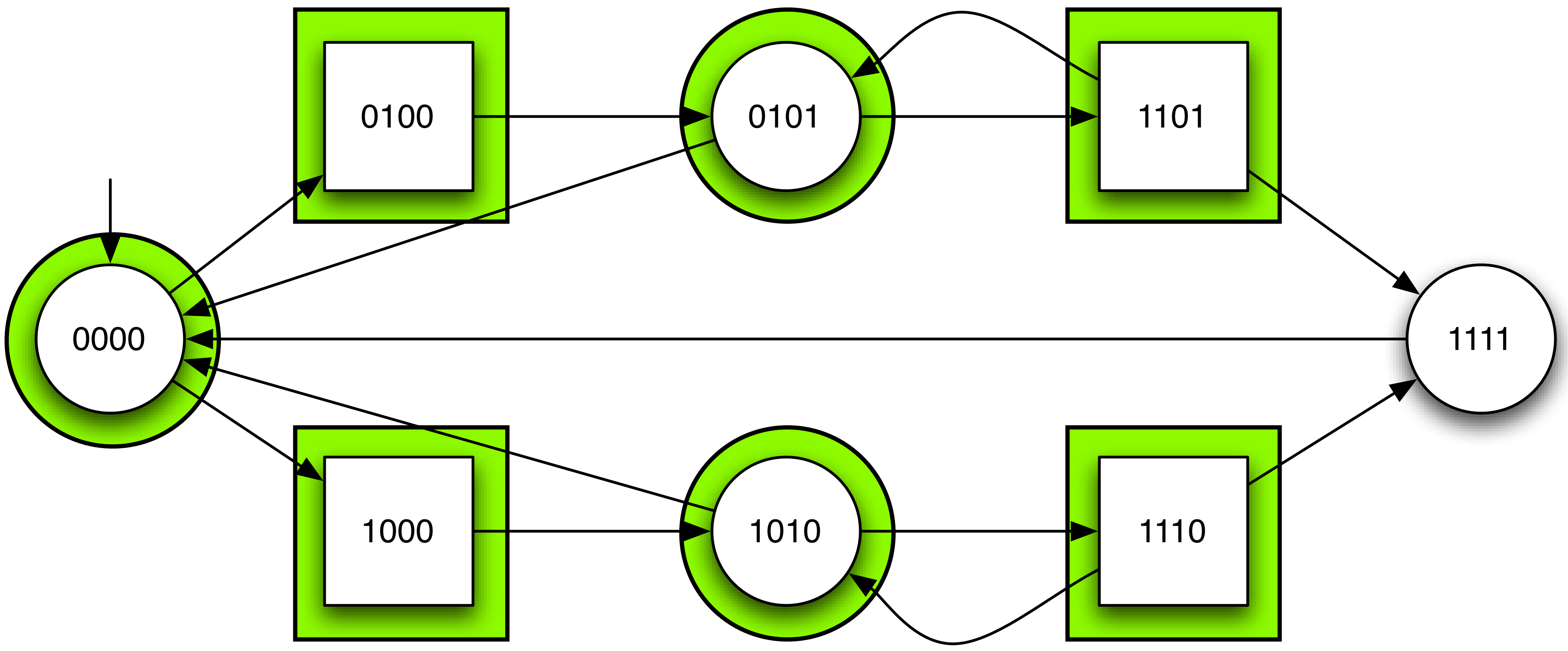
Squared positions,
all successors are red

Fixed points to solve games

Let Q be a set of safe states, the states in which Player 1 can force the game to within Q is given by the following fixed point expression :

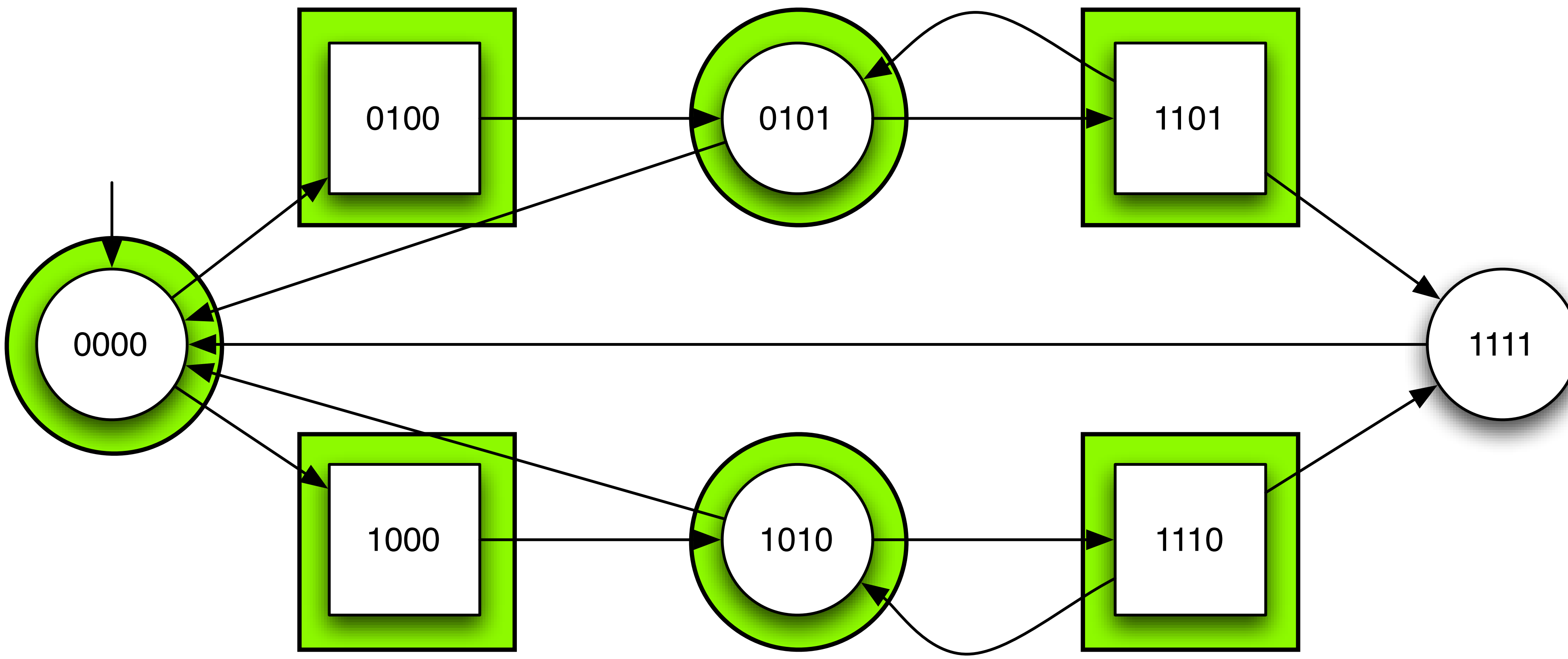
$$\cup\{R \mid R = Q \cap \mathsf{CPre}_1(R)\}$$

Fixed points to solve games



Does Player I, who owns the rounded positions, have a strategy to stay within the set of states $Q \setminus \{1111\}$?

Fixed points to solve games



We must compute

$$\cup\{R \mid R = (Q_1 \cup Q_2) \setminus \{1111\} \cap \mathsf{CPre}_1(R)\}$$

To do that, we use the Tarski fixpoint theorem.

Tarski-Kleene Theorem

Let $\langle L, \sqsubseteq, \sqcup, \sqcap, \top, \bot \rangle$ be a complete lattice, the *f* be a Scott-continuous function on *L*, then

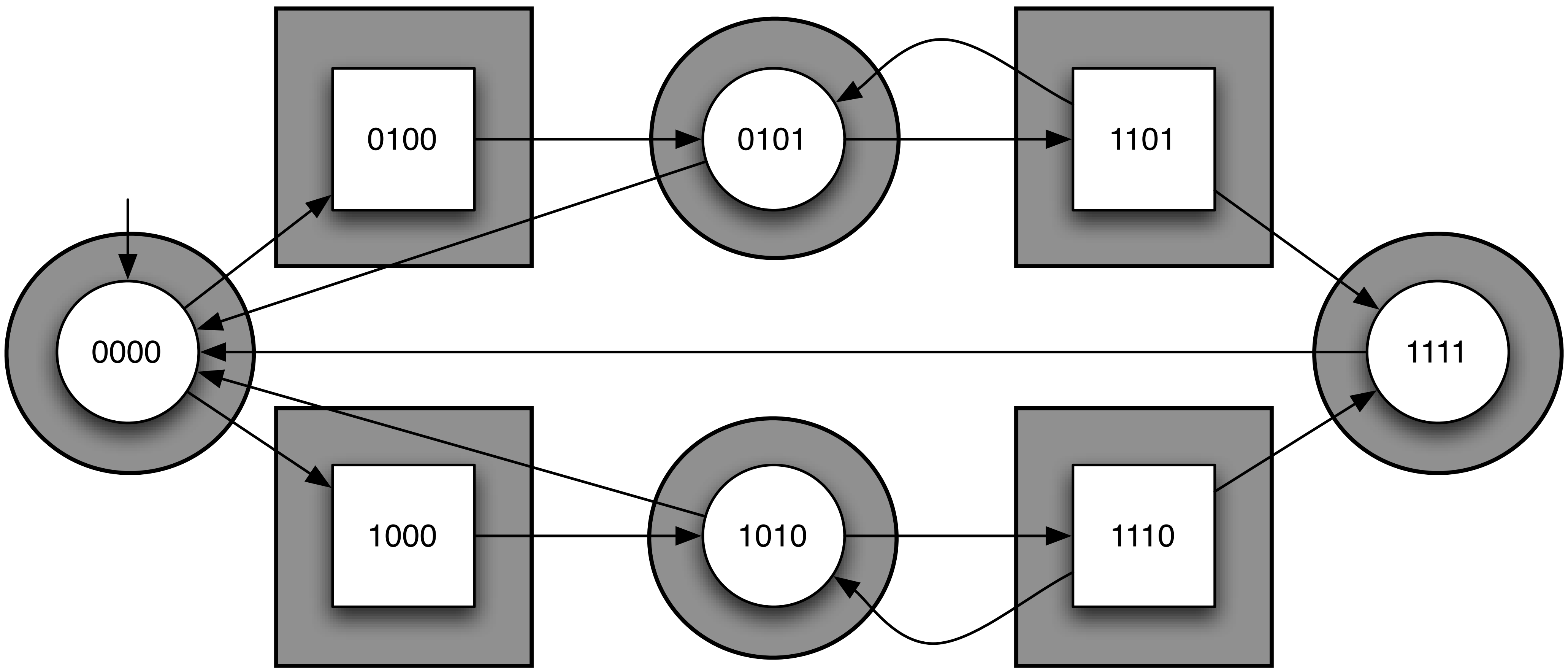
lfp *f* is the limit of the sequence :

$f(\bot), f(f(\bot)), ..., f(... f(\bot)...), ...$

gfp *f* is the limit of the sequence :

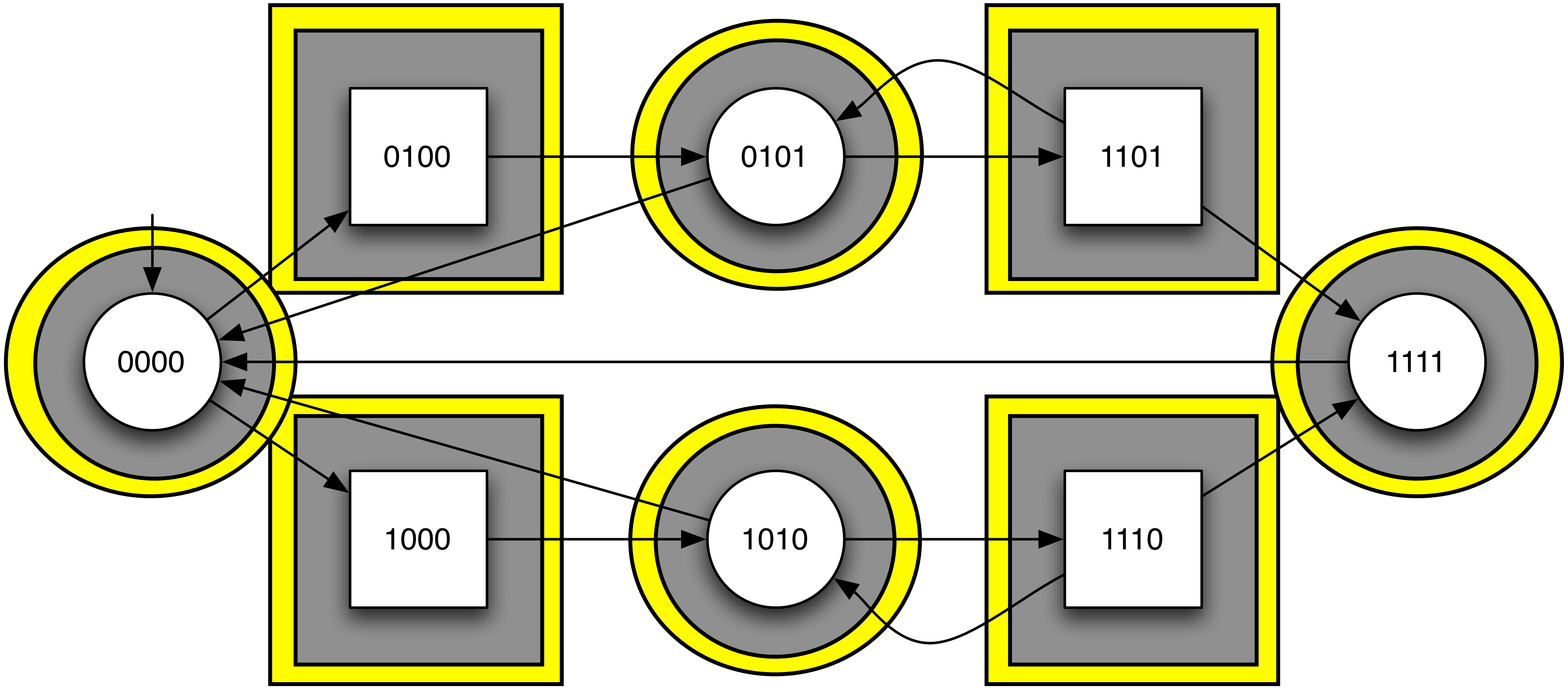
$f(T), f(f(T)), ..., f(....f(T)...), ...$

Fixpoint for a safety game



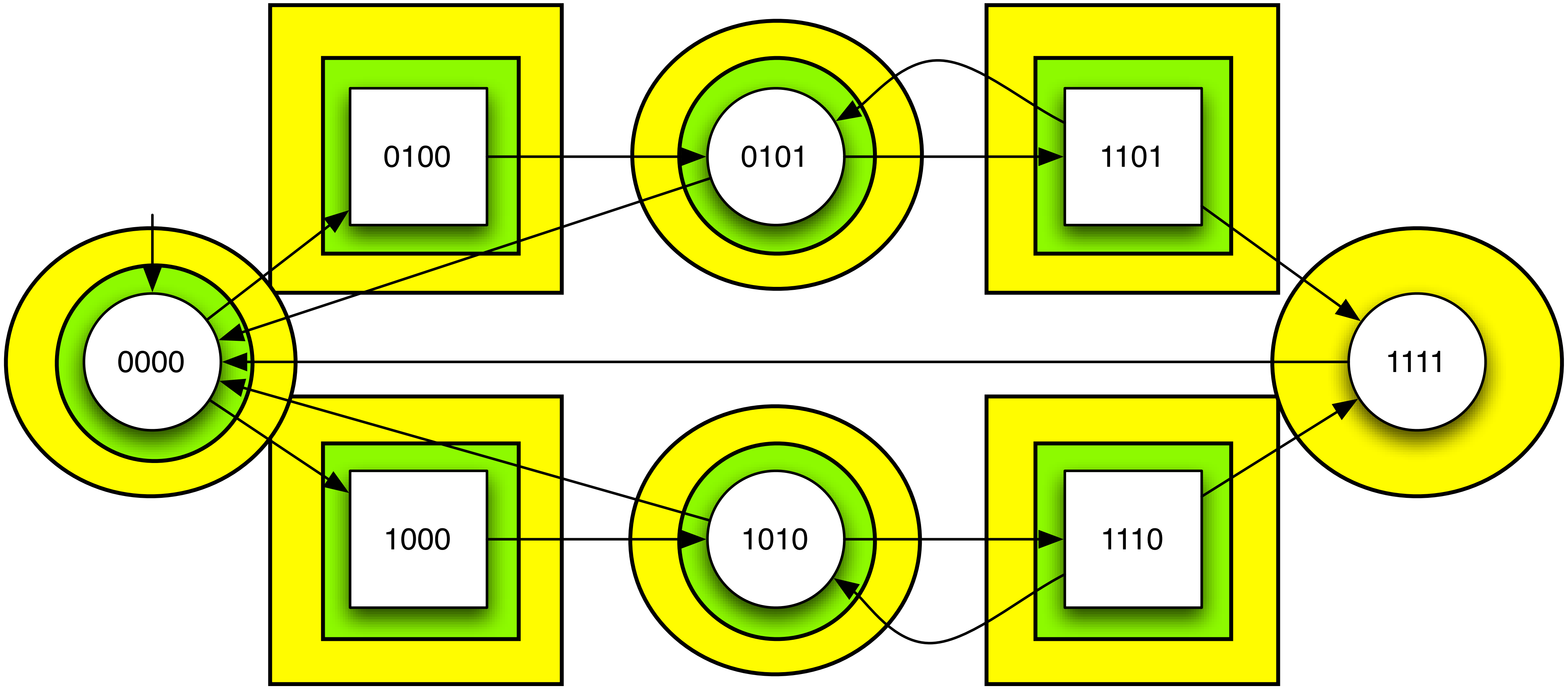
$$X_0 = (Q \setminus \{1111\}) \cap \text{1CPre}(Q)$$

Fixpoint for a safety game



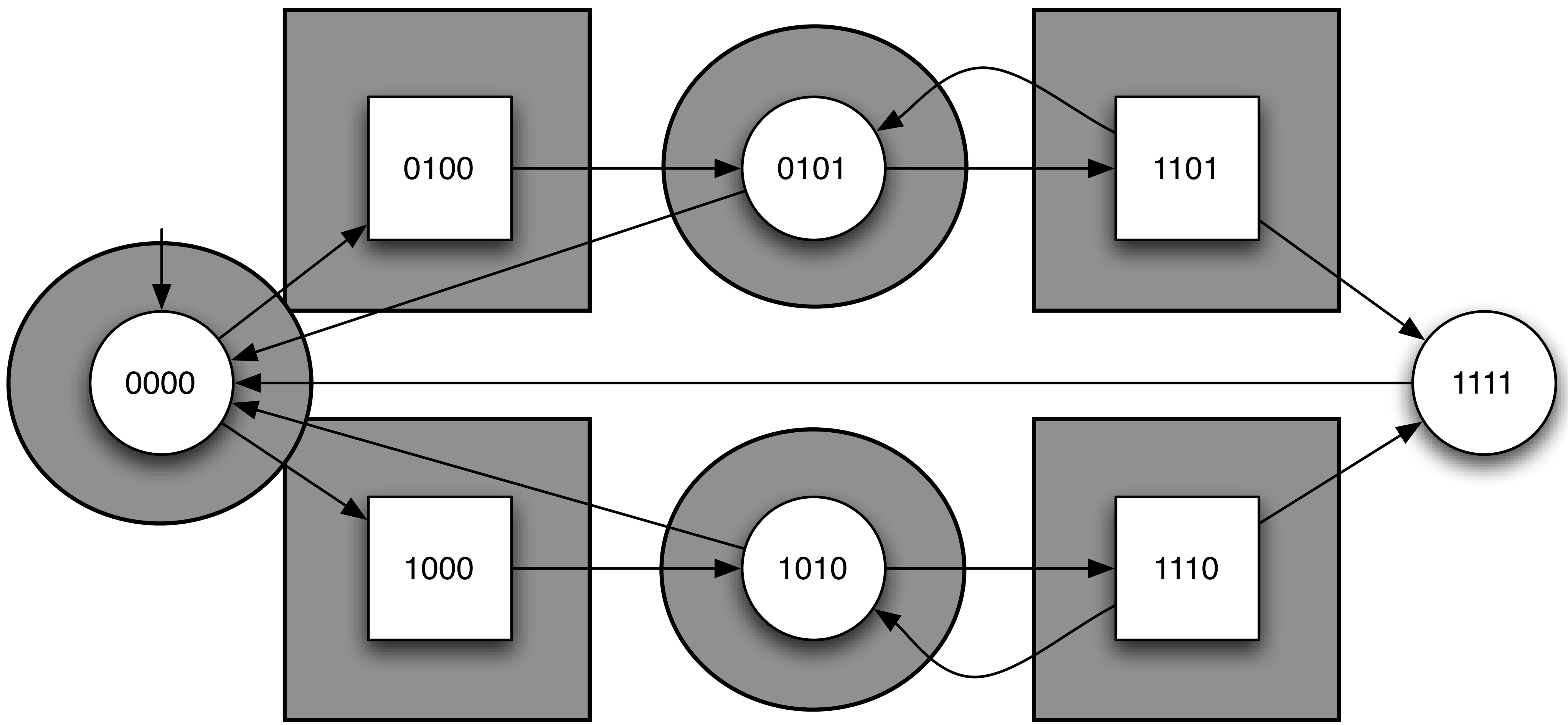
$$X_0 = (Q \setminus \{1111\}) \cap \text{1CPre}(Q)$$

Fixpoint for a safety game



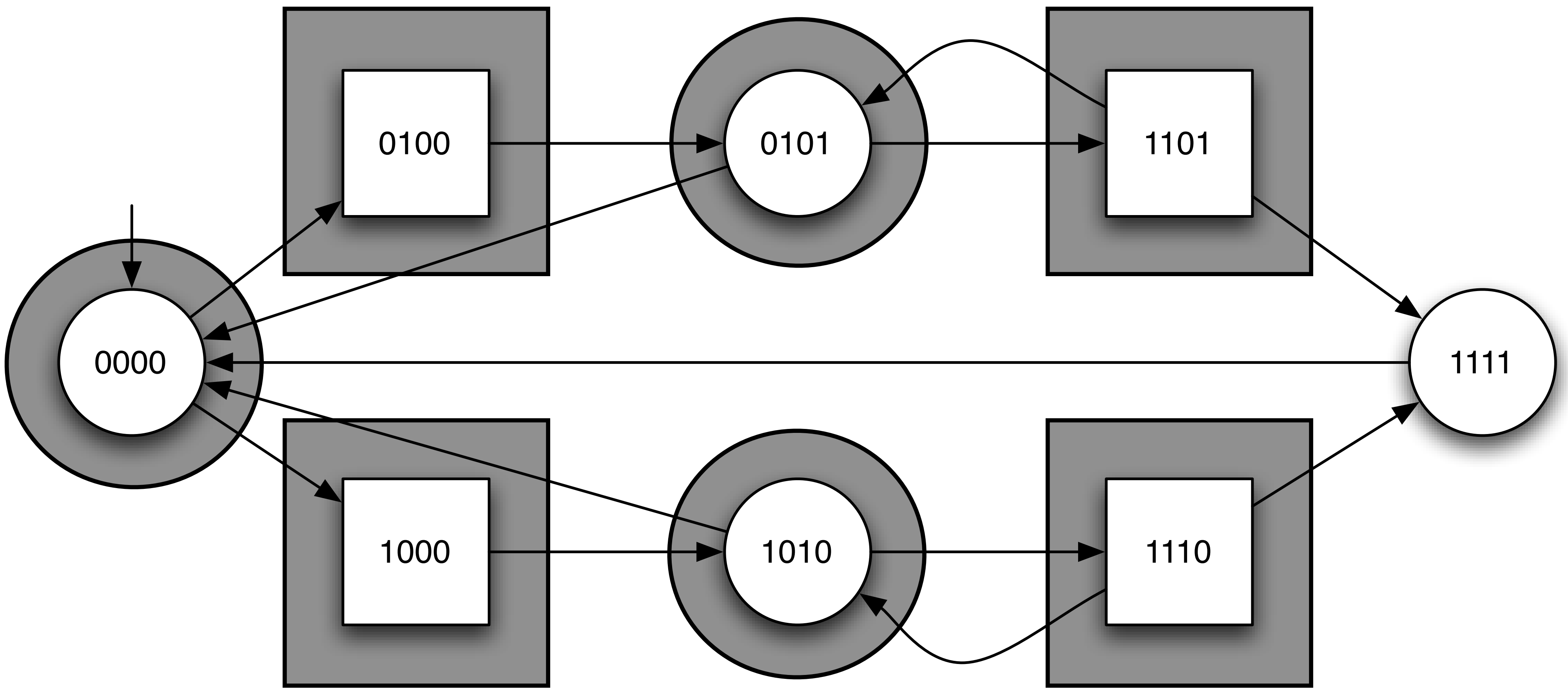
$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

Fixpoint for a safety game



$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

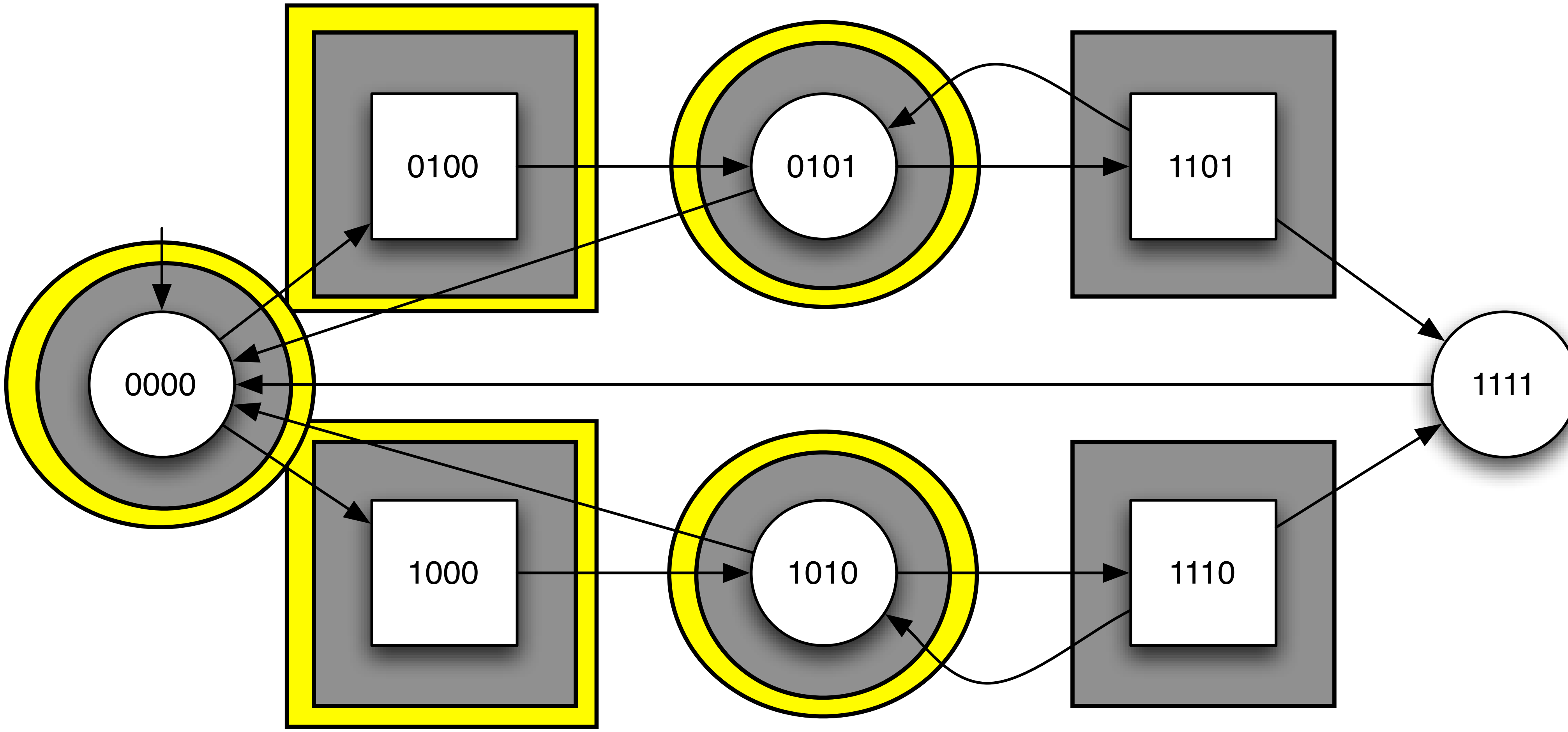
Fixpoint for a safety game



$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_0)$$

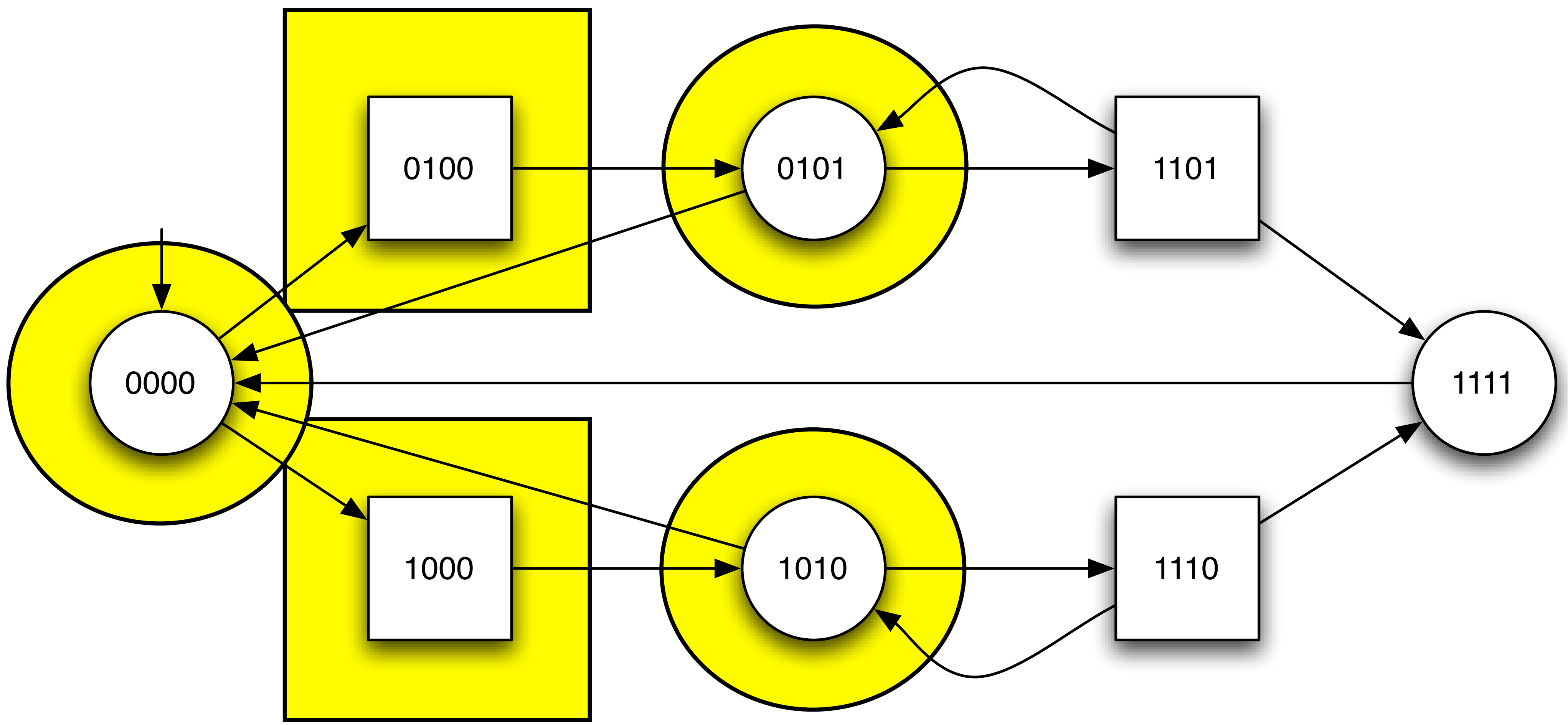
Fixpoint for a safety game



$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_0)$$

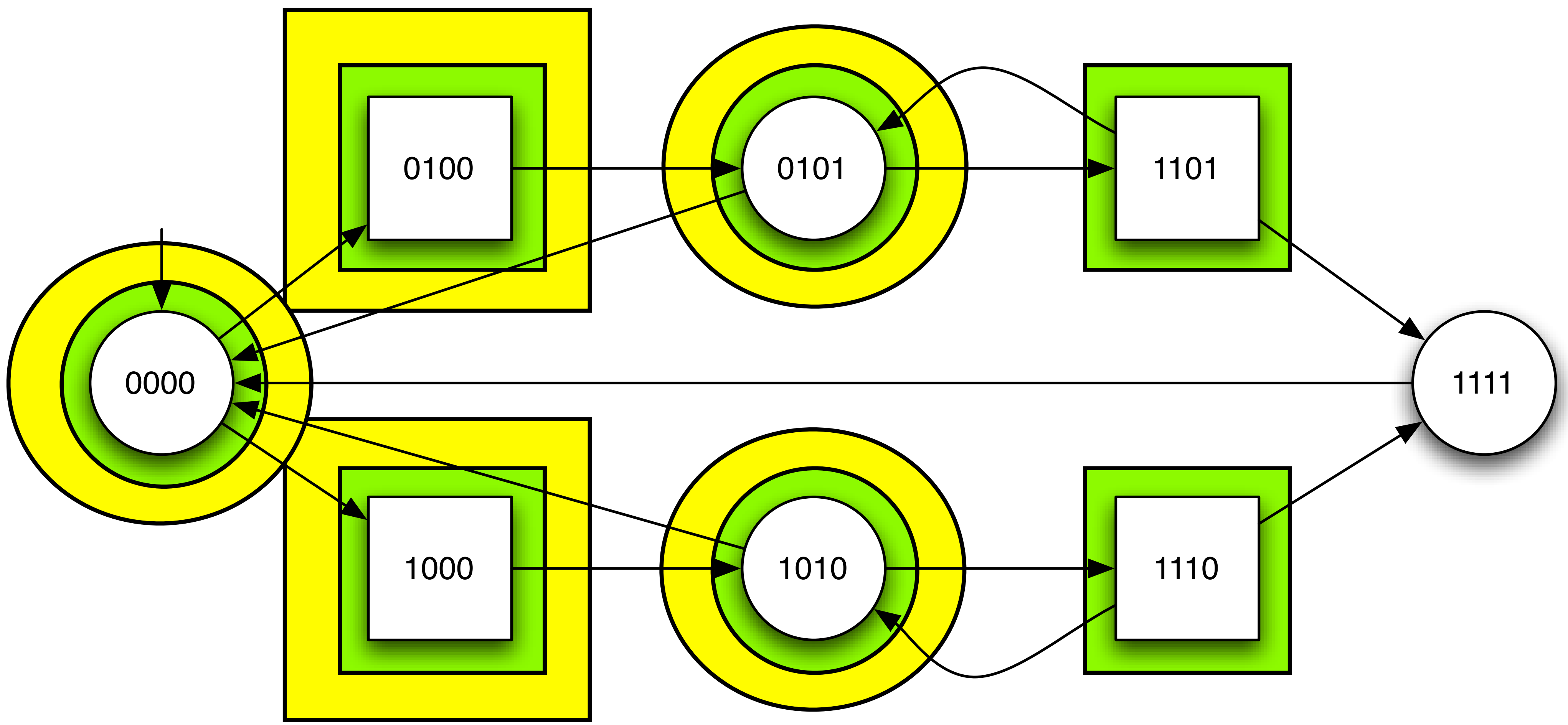
Fixpoint for a safety game



$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_0)$$

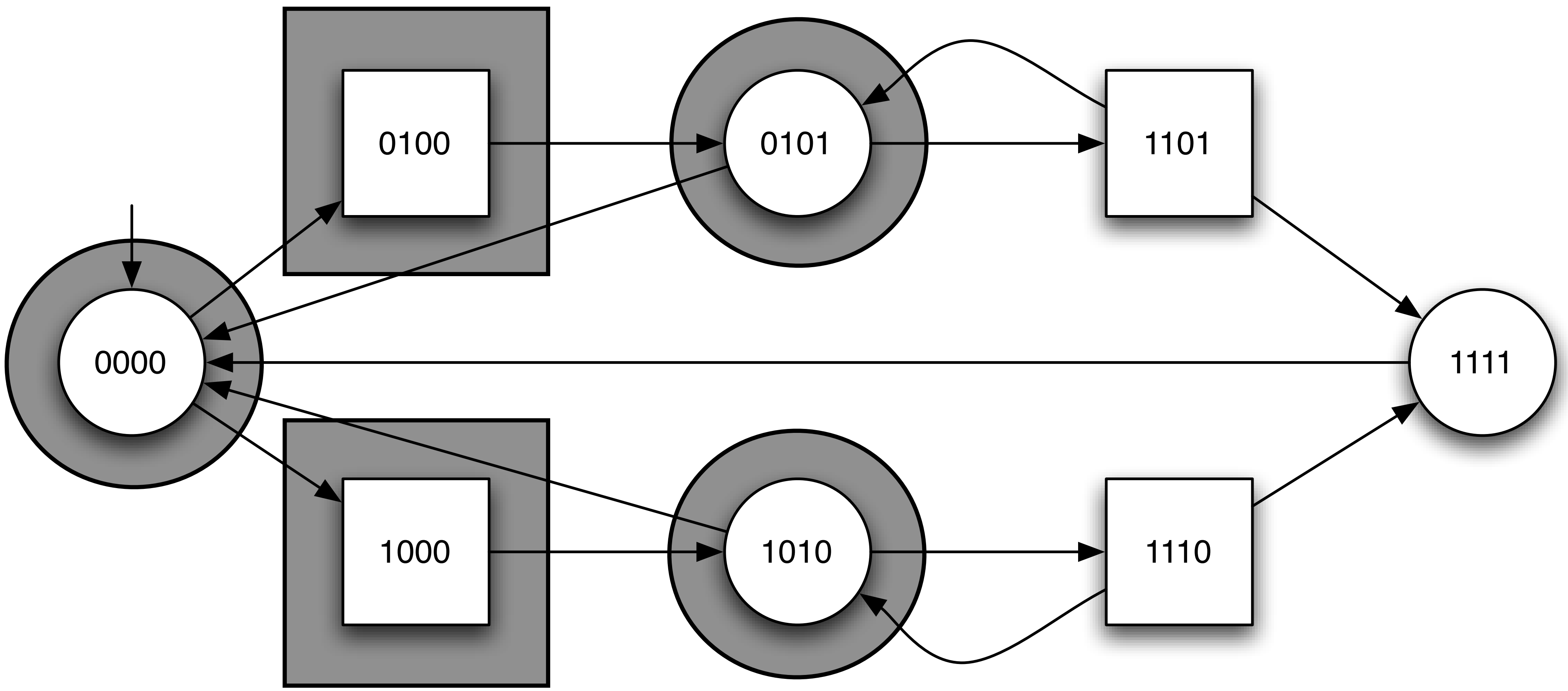
Fixpoint for a safety game



$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_0)$$

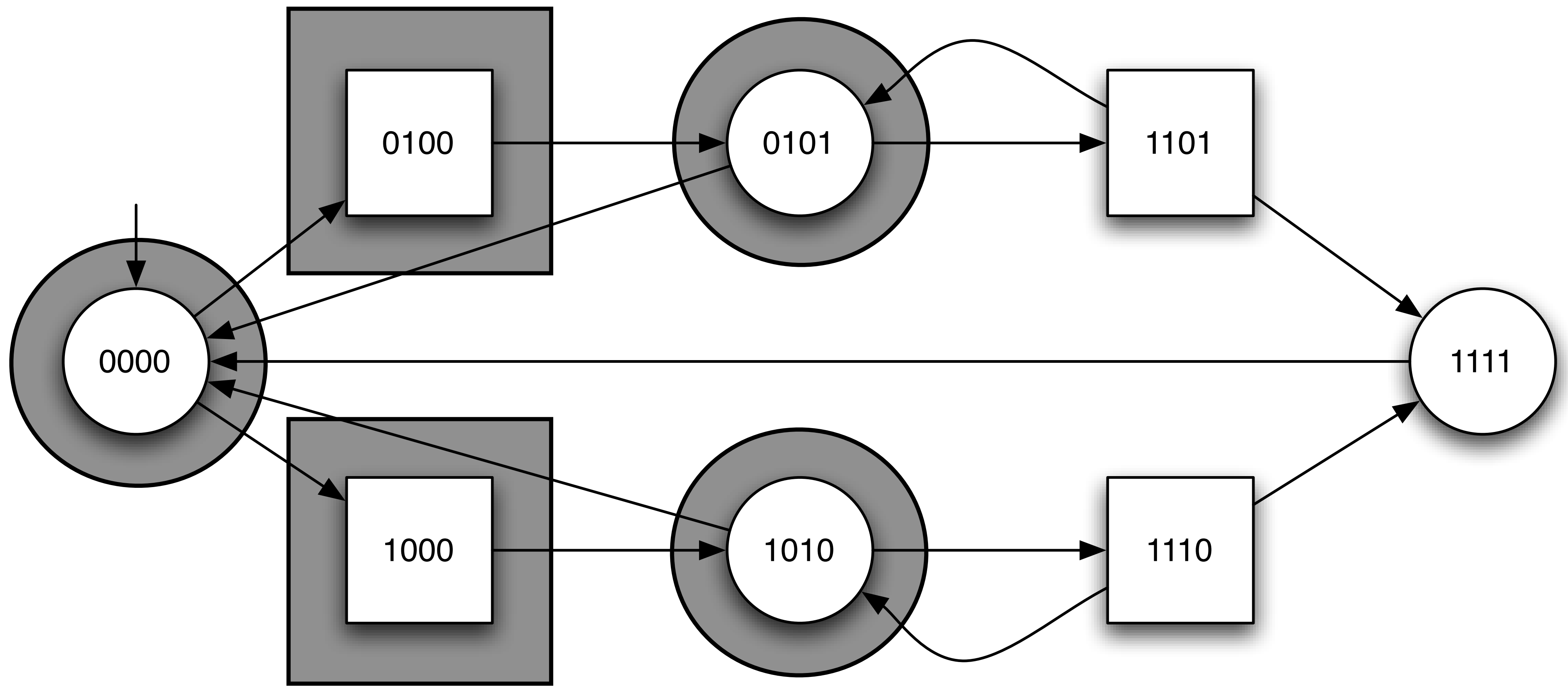
Fixpoint for a safety game



$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_0)$$

Fixpoint for a safety game

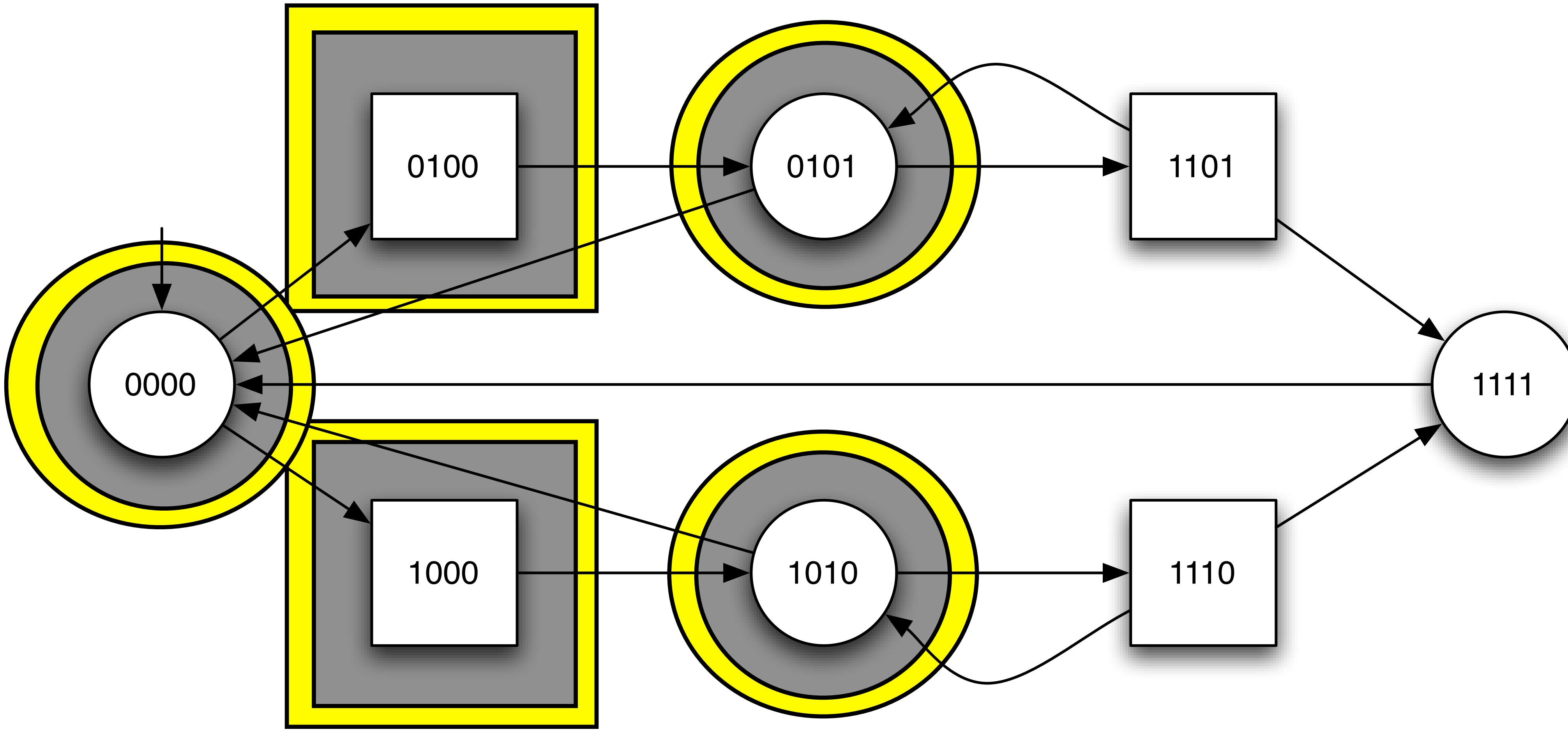


$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_0)$$

$$X_2 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_1)$$

Fixpoint for a safety game

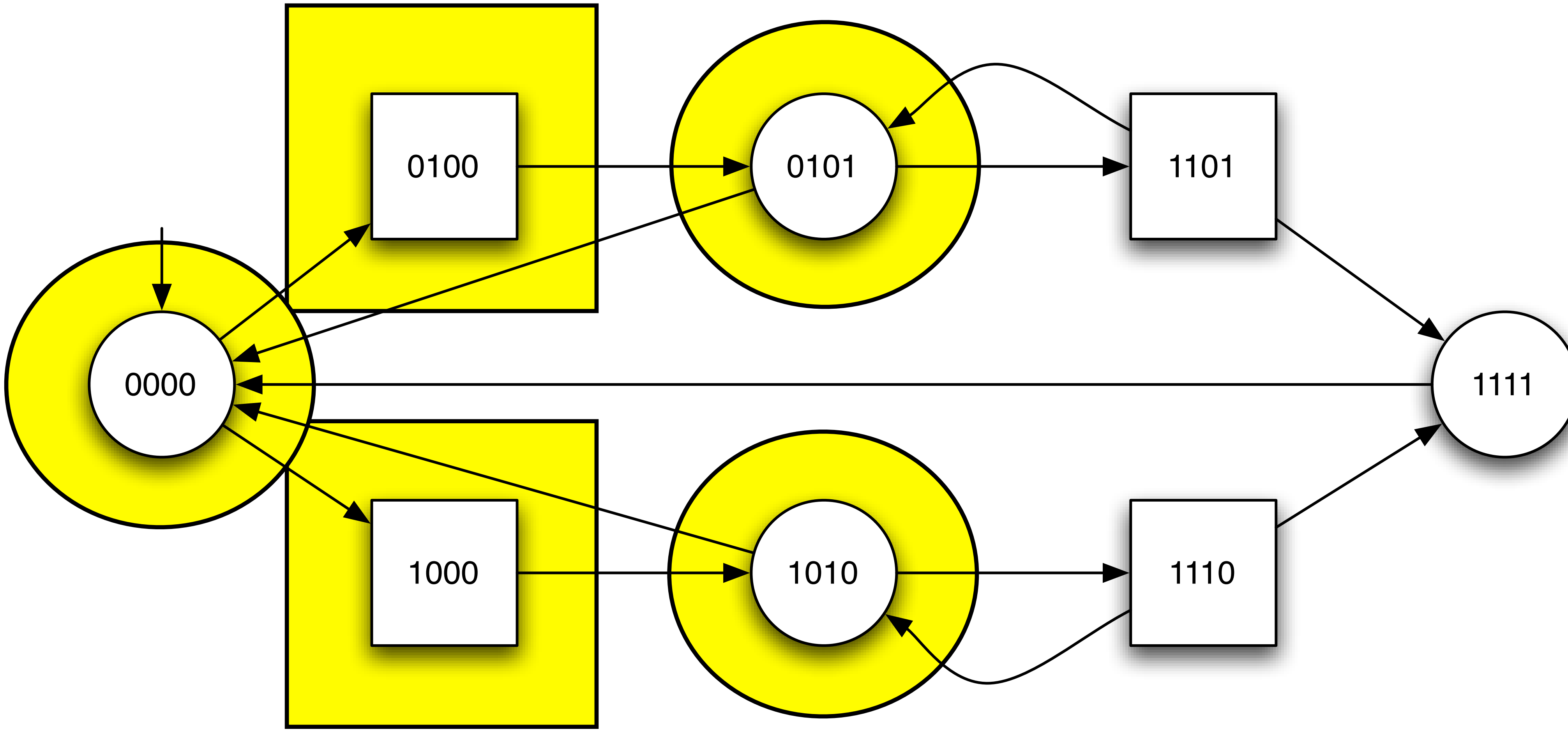


$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_0)$$

$$X_2 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_1)$$

Fixpoint for a safety game

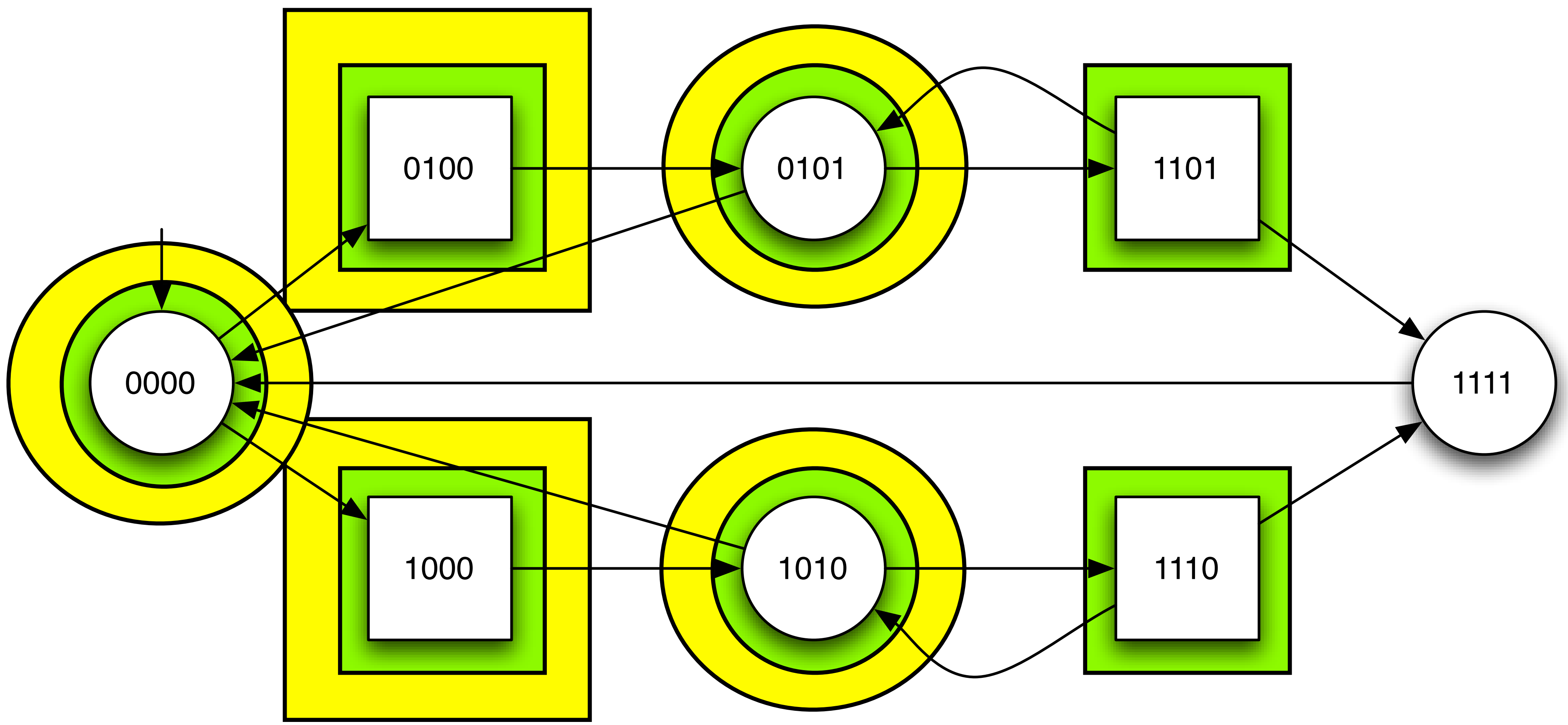


$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_0)$$

$$X_2 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_1)$$

Fixpoint for a safety game

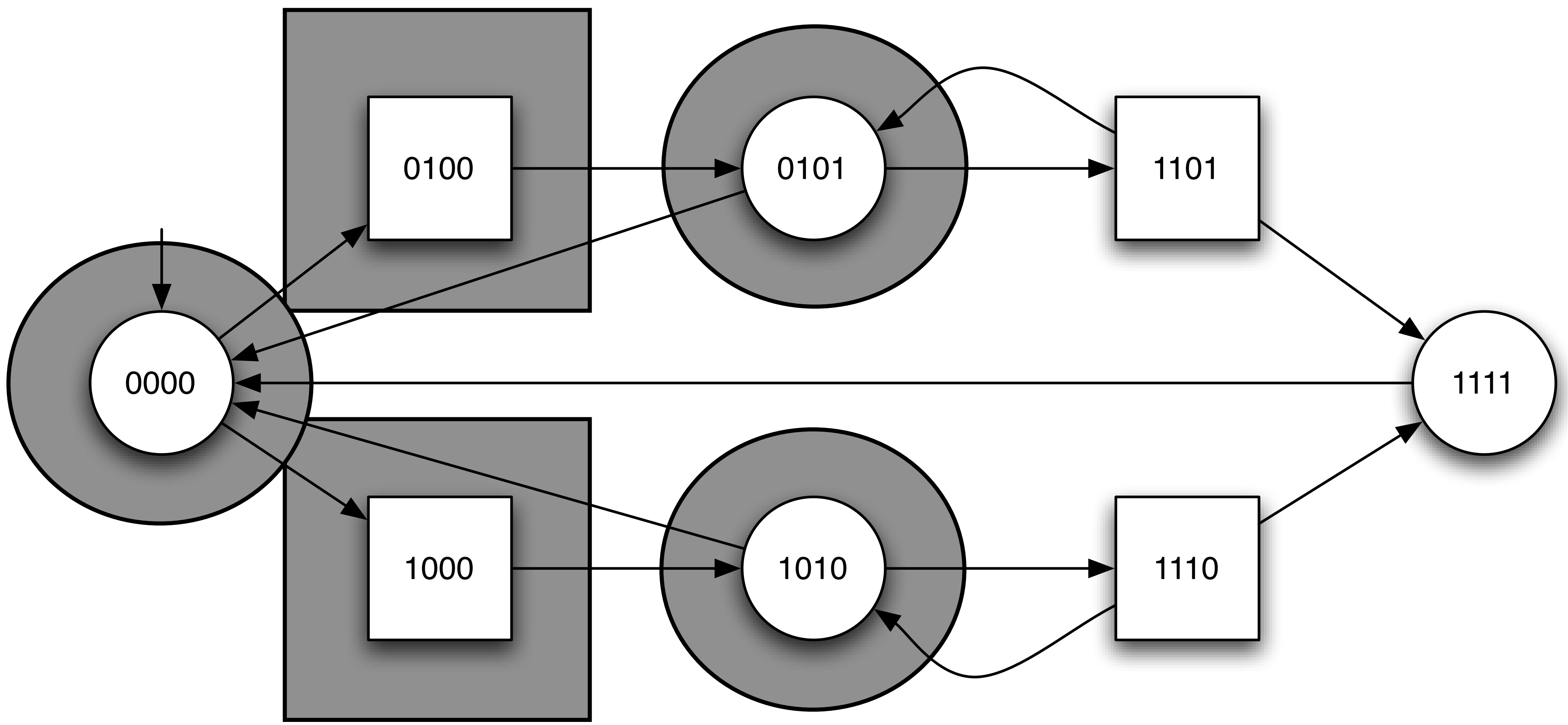


$$X_0 = (Q \setminus \{1111\}) \cap \text{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \text{1CPre}(X_0)$$

$$X_2 = (Q \setminus \{1111\}) \cap \text{1CPre}(X_1)$$

Fixpoint for a safety game

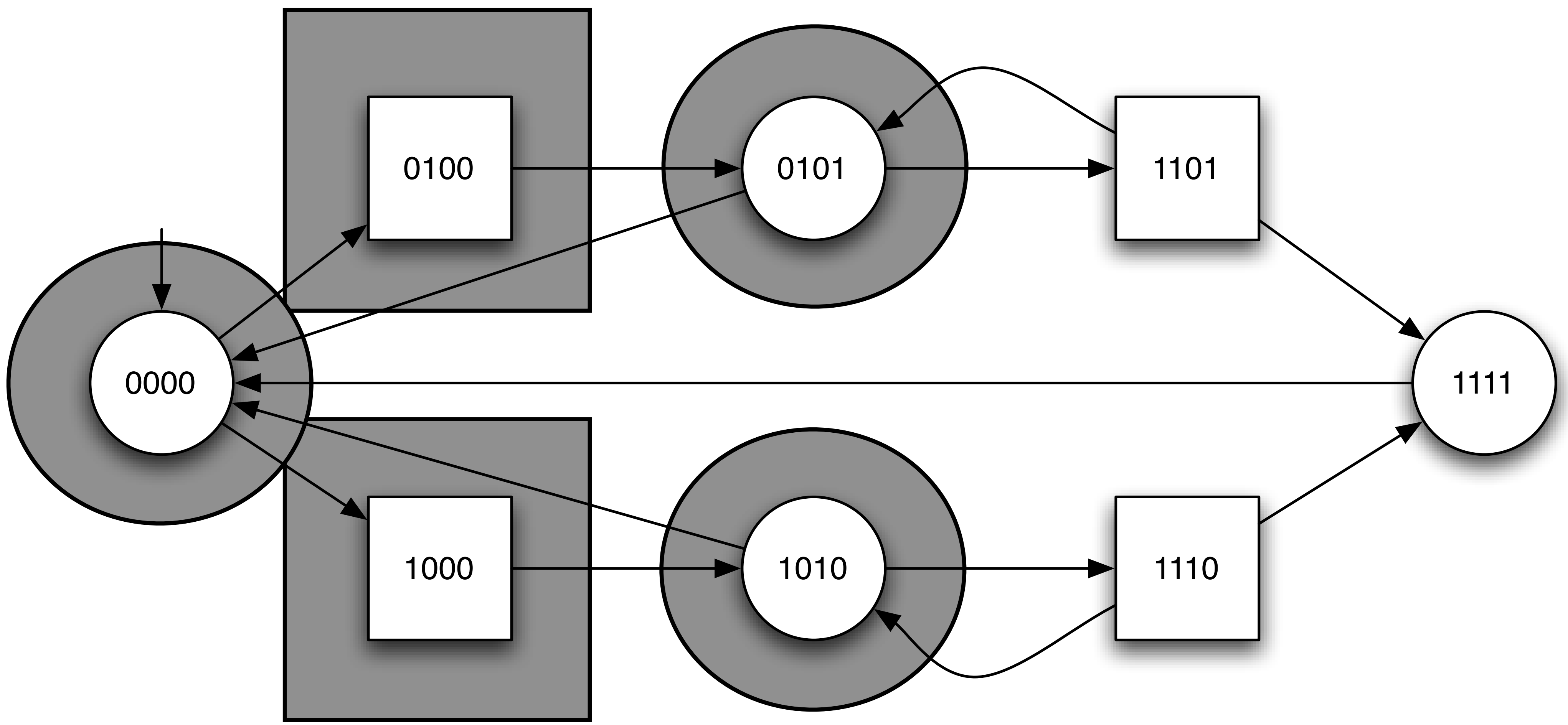


$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_0)$$

$$X_2 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_1)$$

Fixpoint for a safety game



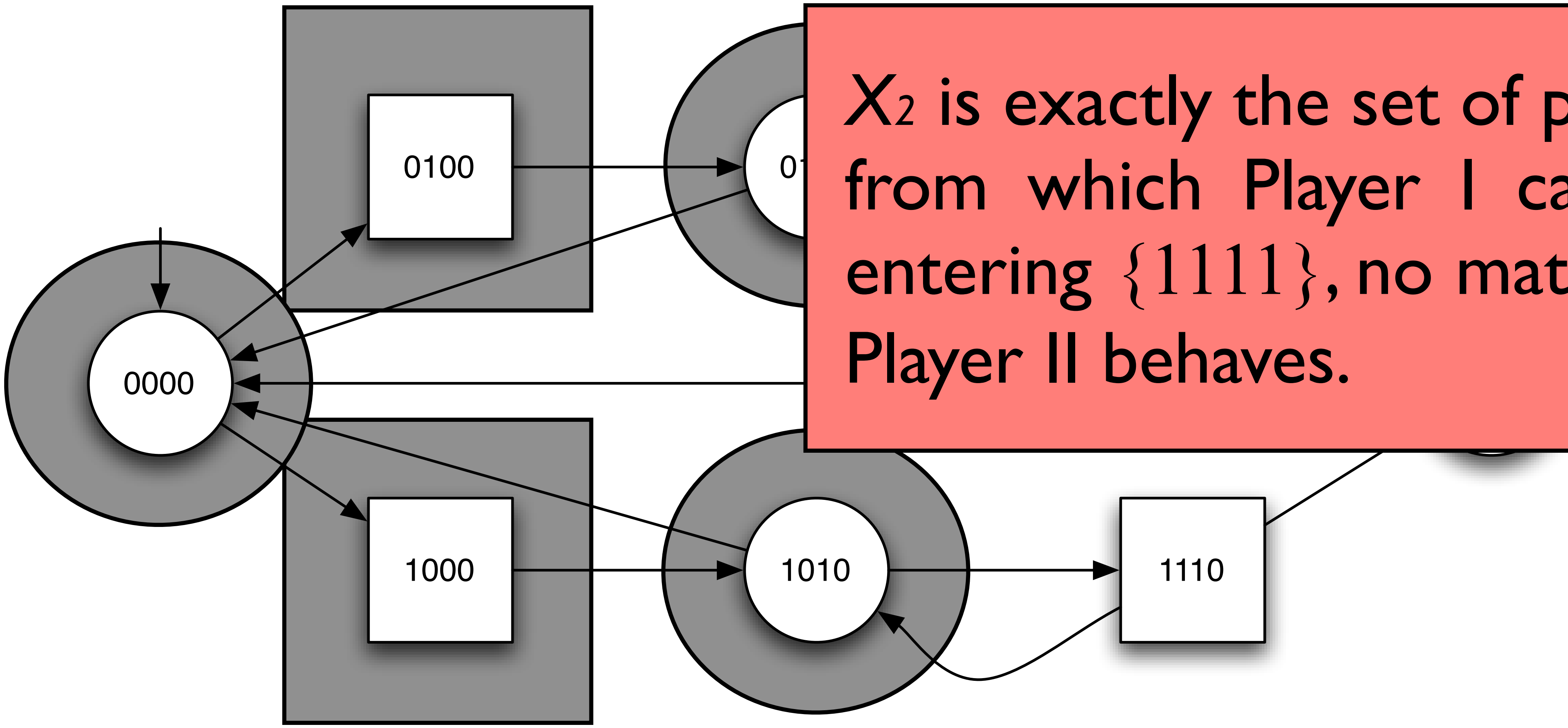
$$X_0 = (Q \setminus \{1111\}) \cap \text{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \text{1CPre}(X_0)$$

$$X_2 = (Q \setminus \{1111\}) \cap \text{1CPre}(X_1) = X_1$$

This is the greatest fixed point

Fixpoint for a safety game



X_2 is exactly the set of positions from which Player I can avoid entering $\{1111\}$, no matter how Player II behaves.

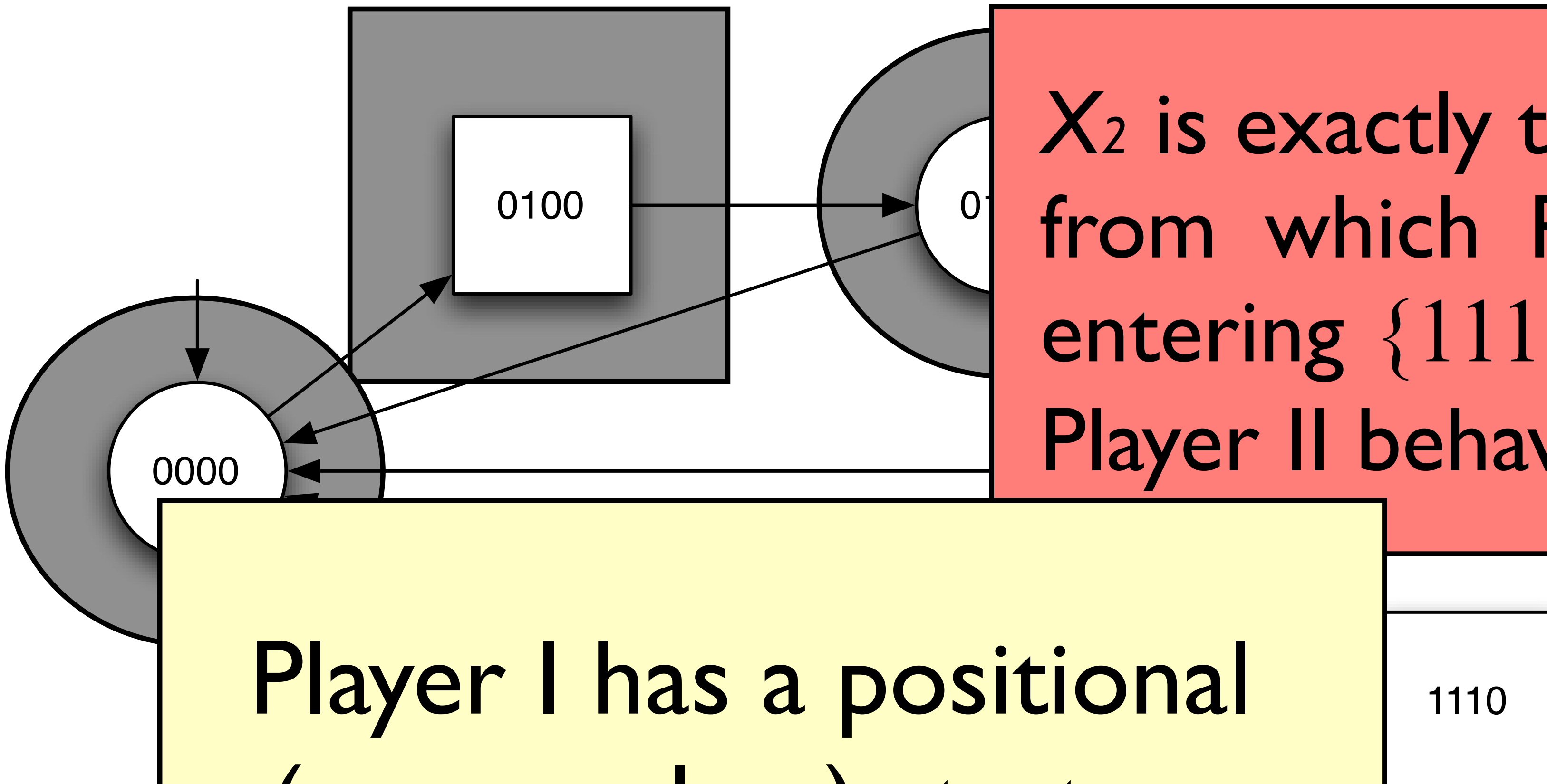
This is the greatest fixed point

$$X_0 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(Q)$$

$$X_1 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_0)$$

$$X_2 = (Q \setminus \{1111\}) \cap \mathsf{1CPre}(X_1) = X_1$$

Fixpoint for a safety game



X_2 is exactly the set of positions from which Player I can avoid entering $\{1111\}$, no matter how Player II behaves.

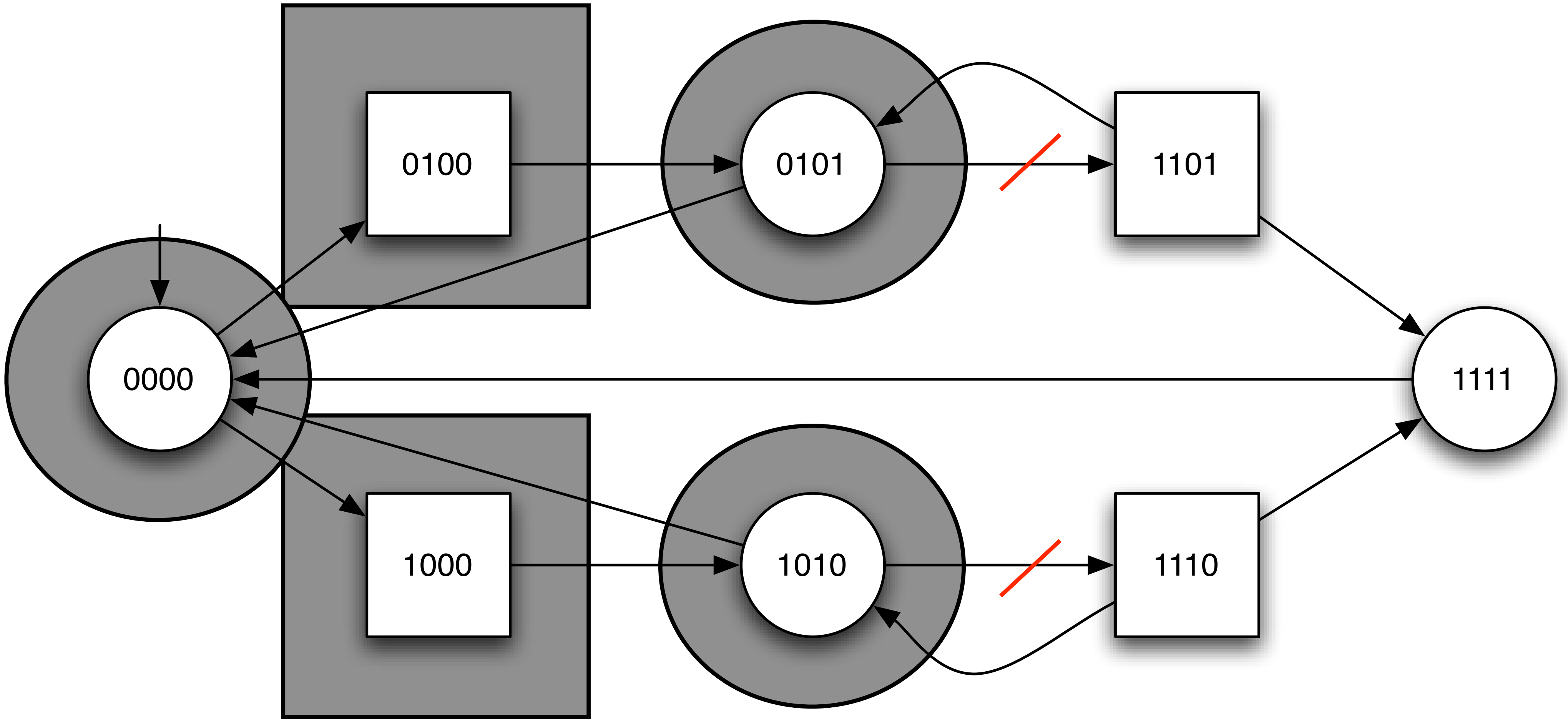
Player I has a positional (memoryless) strategy to win the game

This is the greatest fixed point

Pre(Q)

$$X_1 = (Q \setminus \{1111\}) \cap 1\text{CPre}(X_0)$$

$$X_2 = (Q \setminus \{1111\}) \cap 1\text{CPre}(X_1) = X_1$$


0100
0101
1101
0000
1111
1000
1010
1110

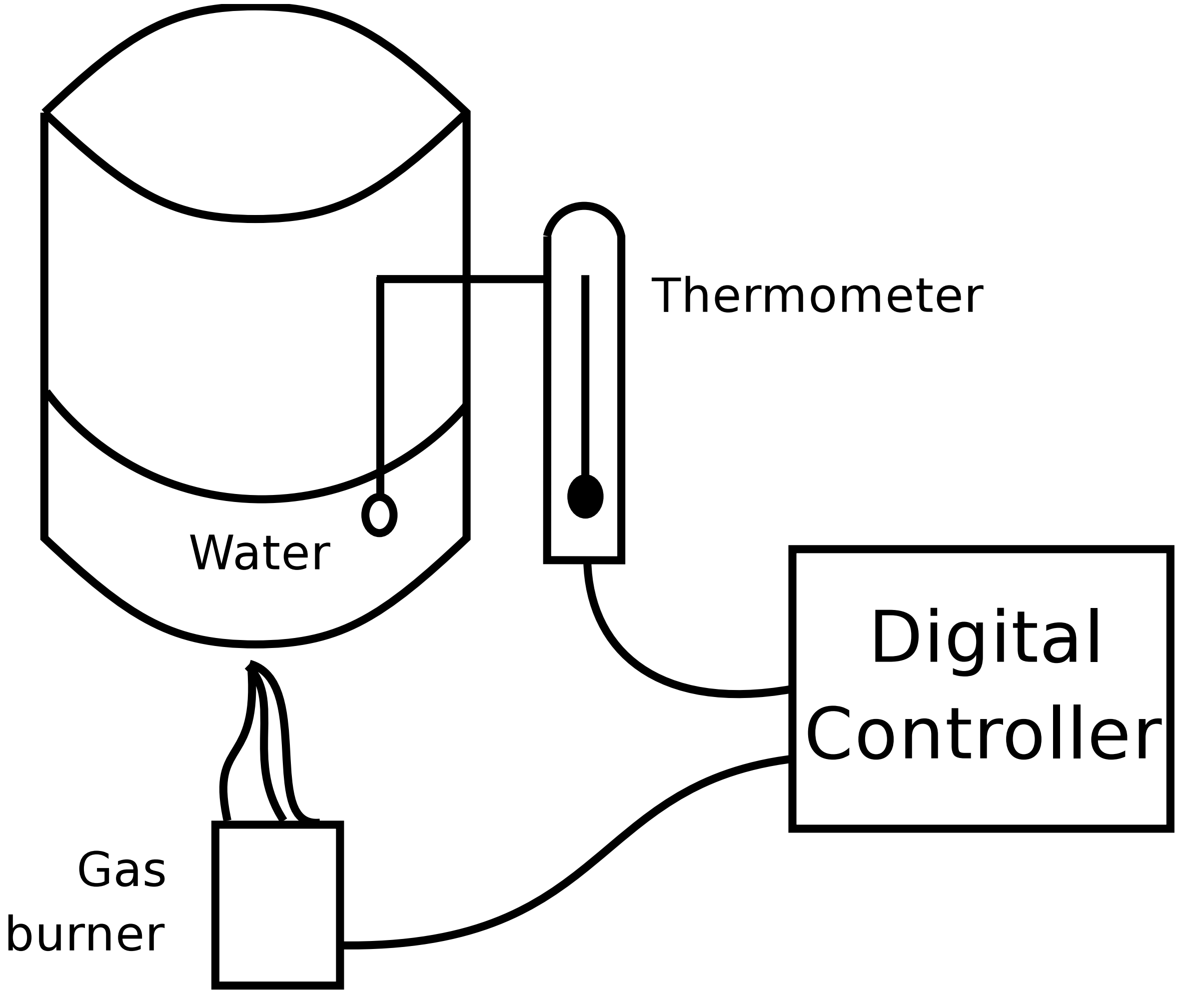
Let $G = \langle Q_1, Q_2, \iota, \delta \rangle$ be a TGS, let $\mathsf{Reach}(G, Q)$ be a **reachability** game defined on G, Player I has a winning strategy for this game iff

$$\iota \in \cap\{R \mid R = Q \cup \mathsf{CPre}_1(R)\}$$

Let $G = \langle Q_1, Q_2, \iota, \delta \rangle$ be a TGS, let $\mathsf{Safe}(G, Q)$ be a **safety** game defined on G, Player I has a winning strategy for this game iff

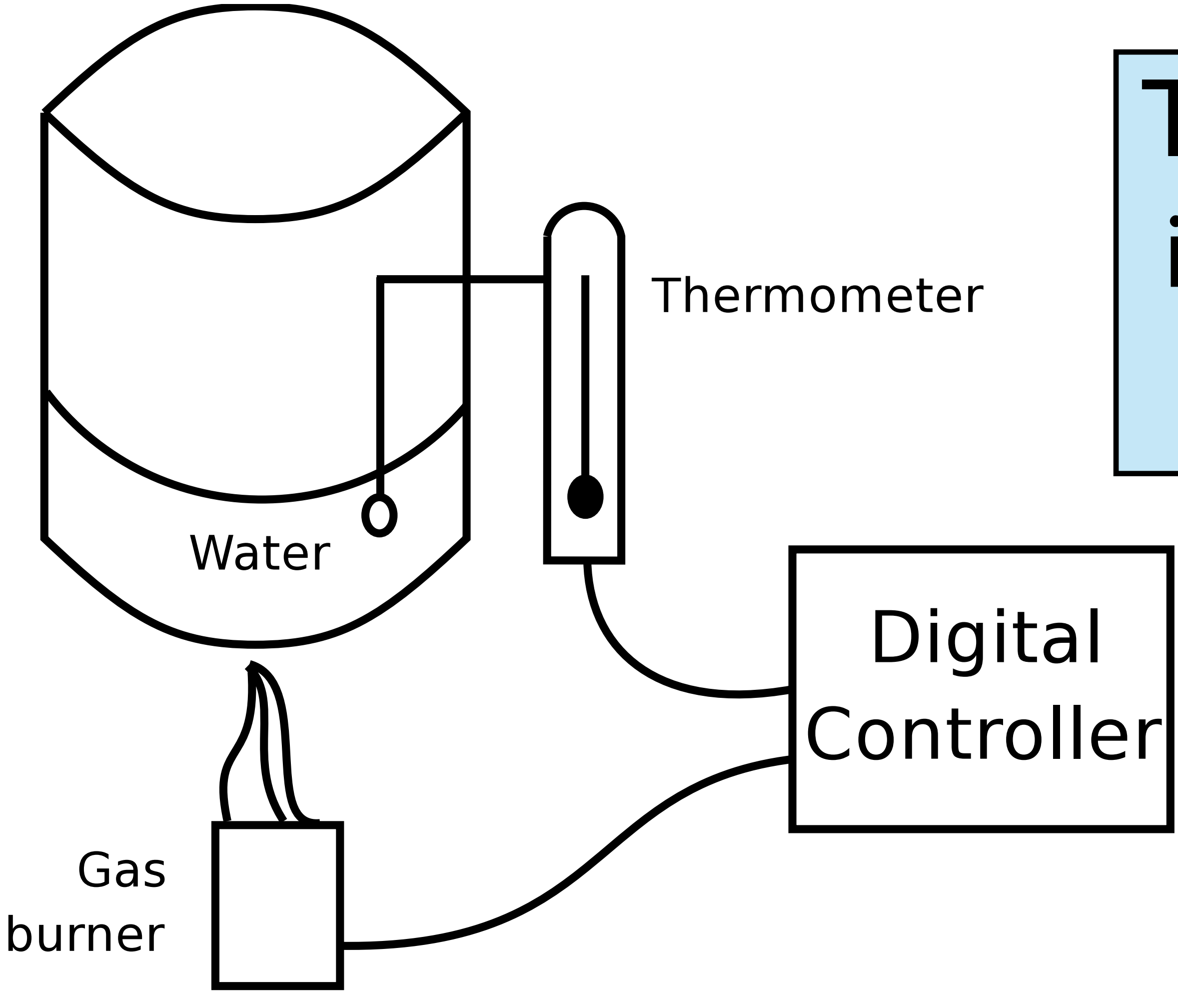
$$\iota \in \cup\{R \mid R = Q \cap \mathsf{CPre}_1(R)\}$$

Games of imperfect information



Typical hybrid system

Perfect information hypothesis?

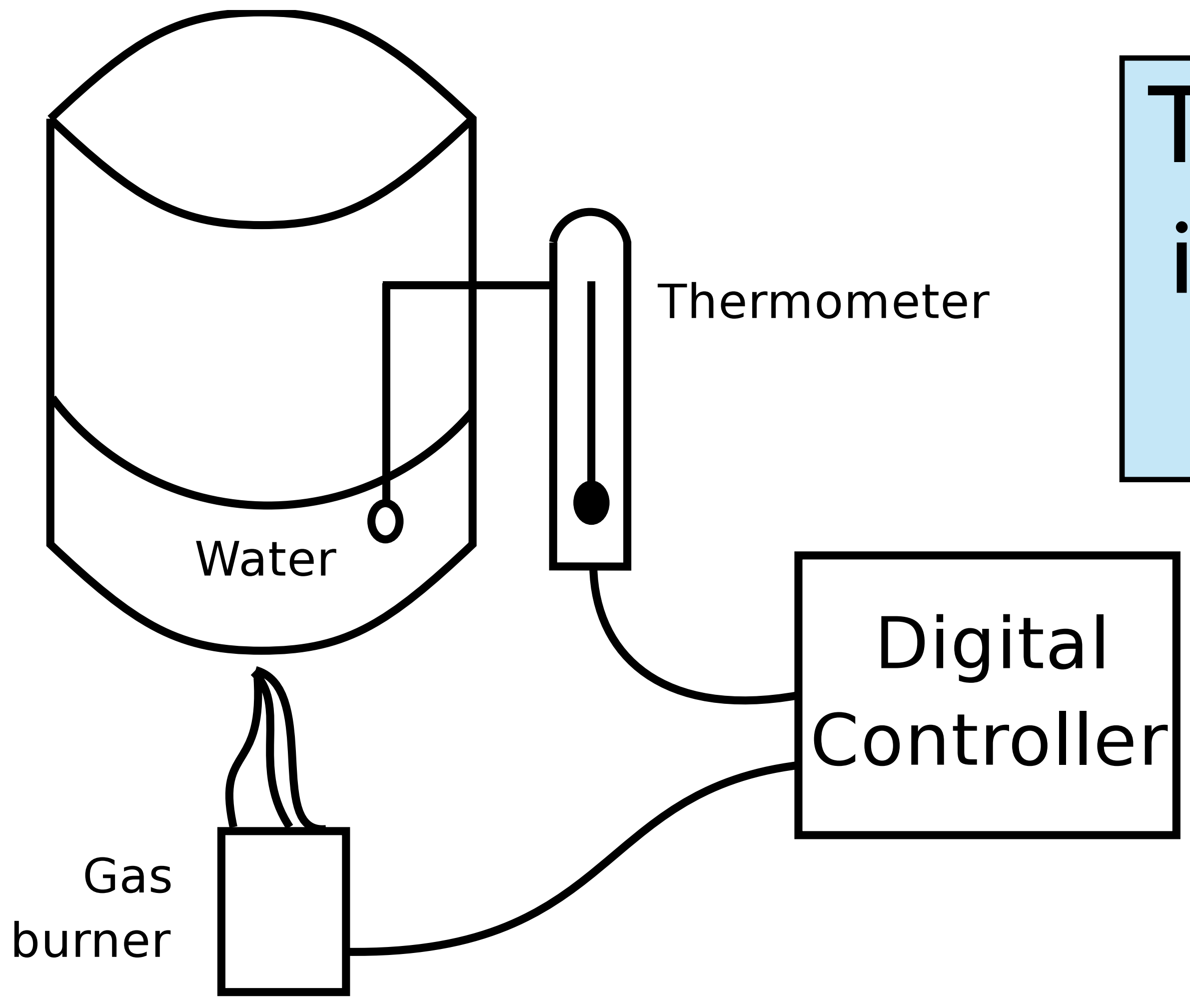


The temperature is in the interval $(c-1, c+1)$

Typical hybrid system

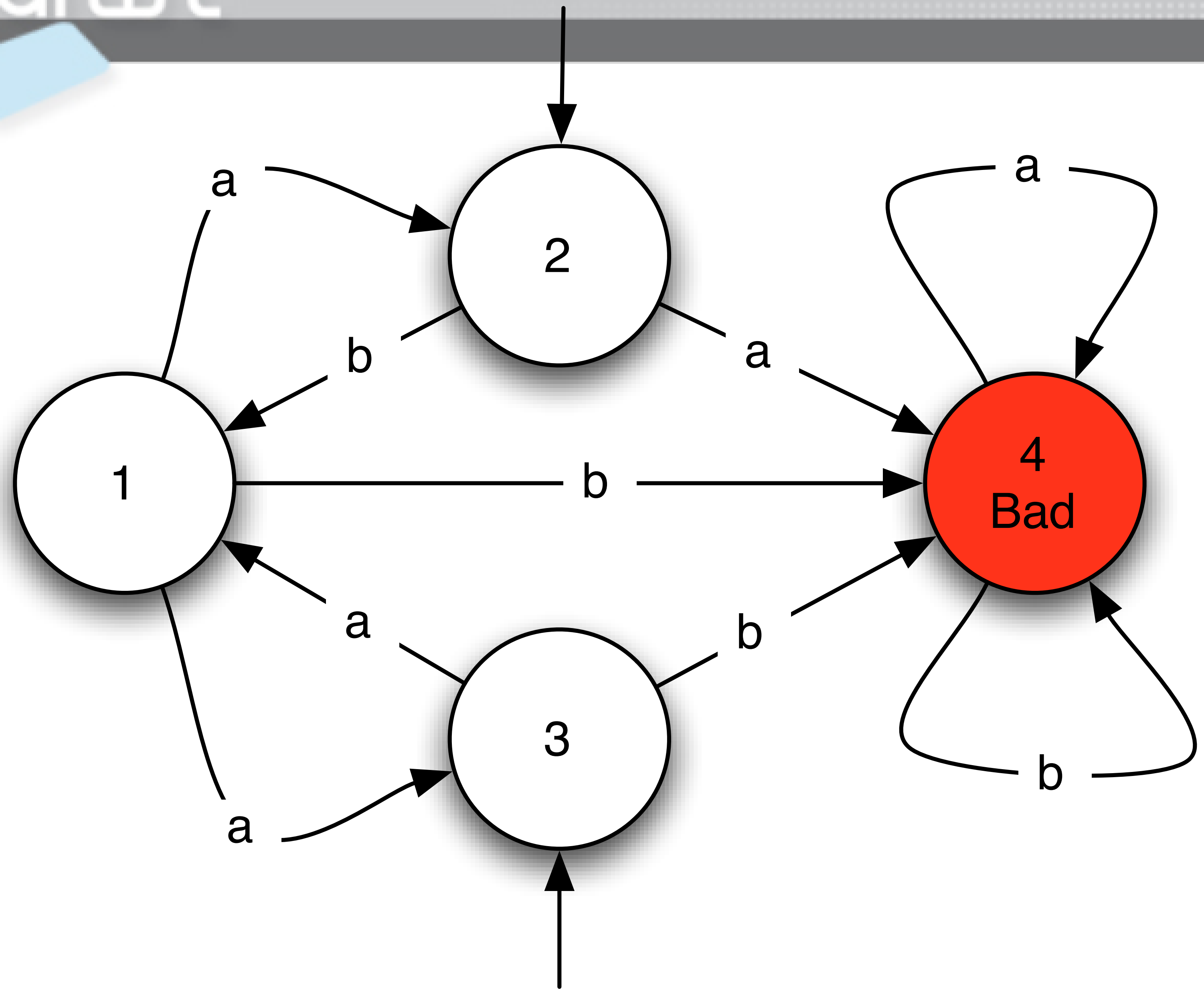
Perfect information hypothesis?

Finite precision = imperfect information

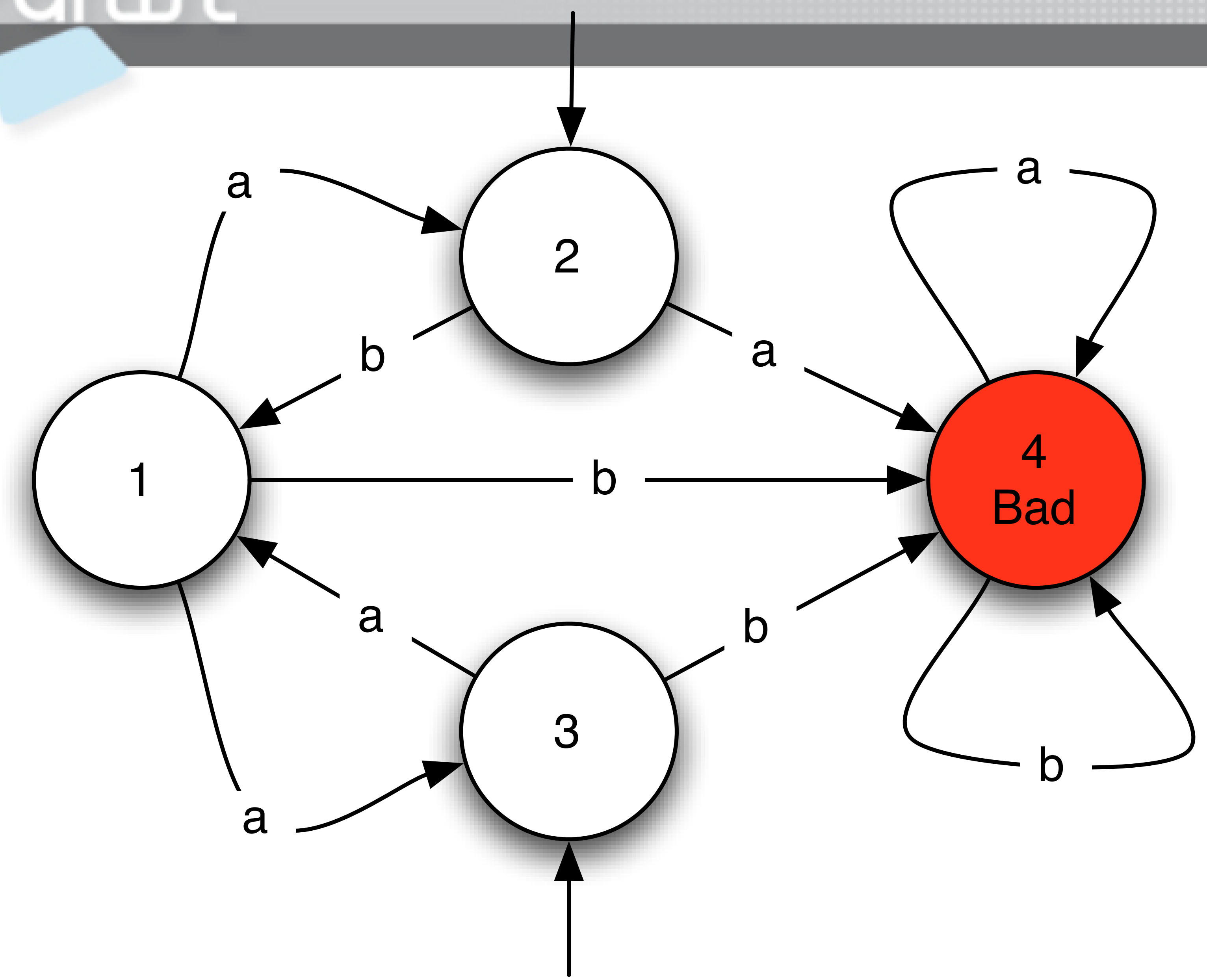


The temperature is in the interval $(c-1, c+1)$

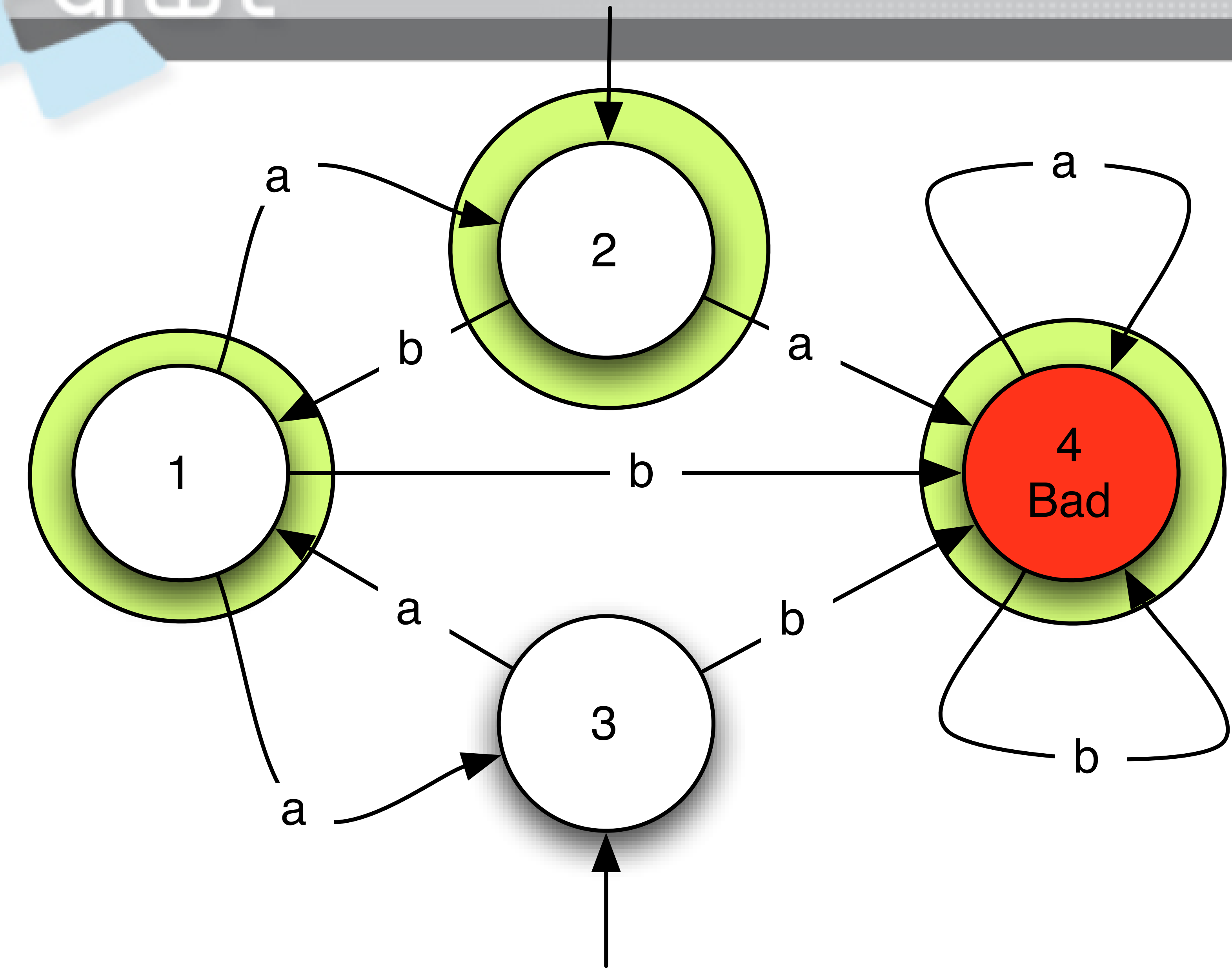
Typical hybrid system



Player 0 chooses a letter
Player 1 resolves nondeterminism

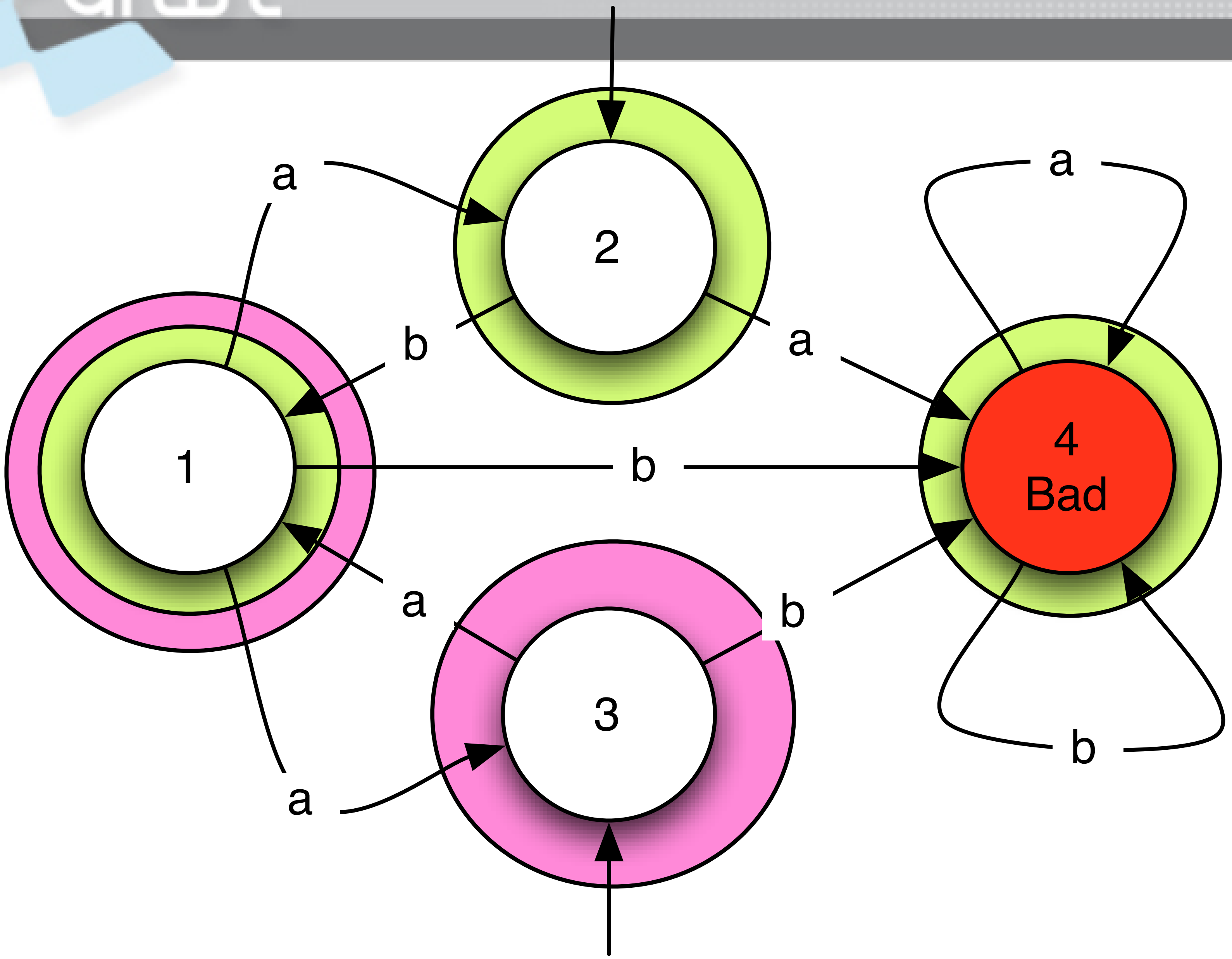


Imperfect information



Obs 0

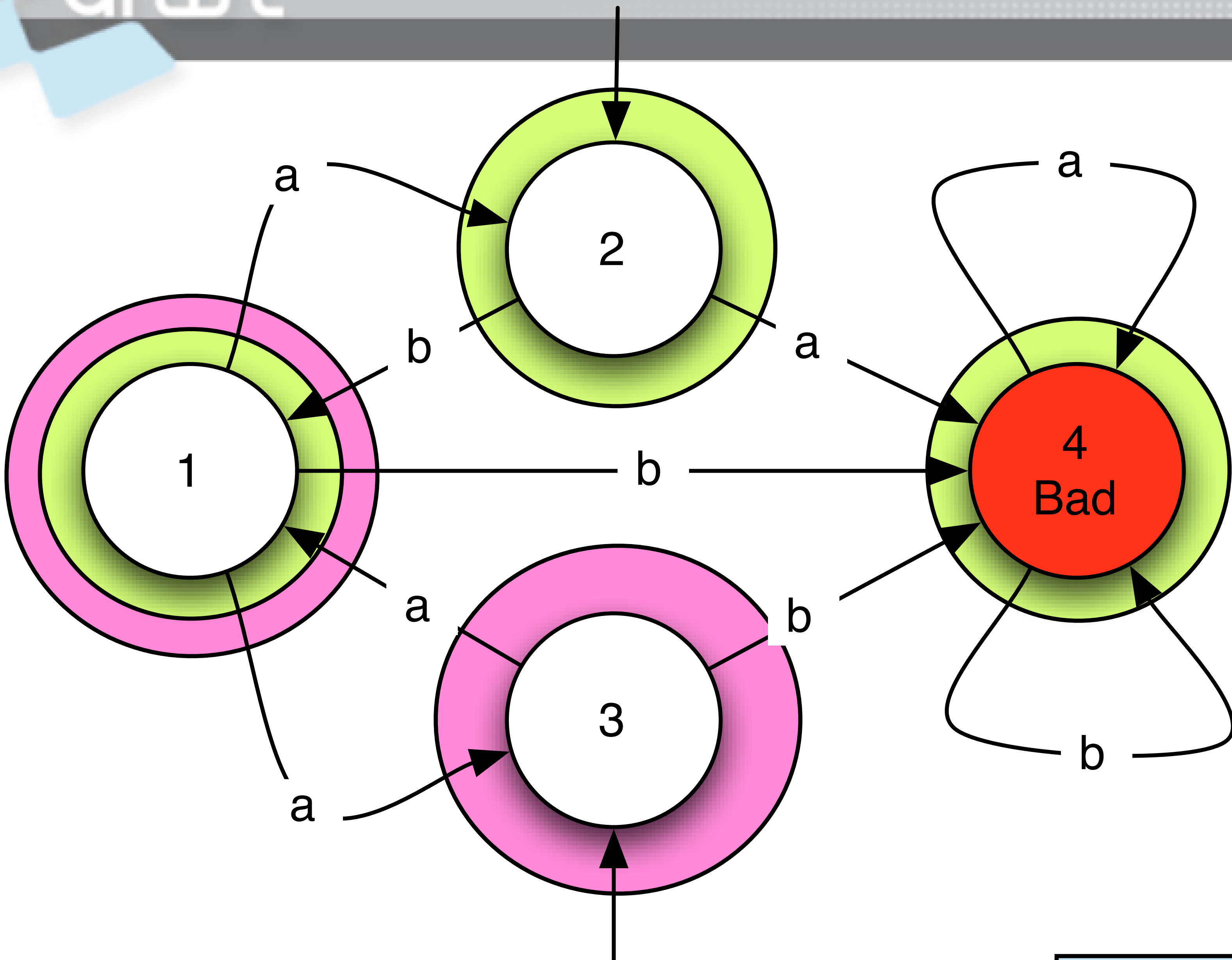
Imperfect information



Obs 0

Obs 1

Imperfect information

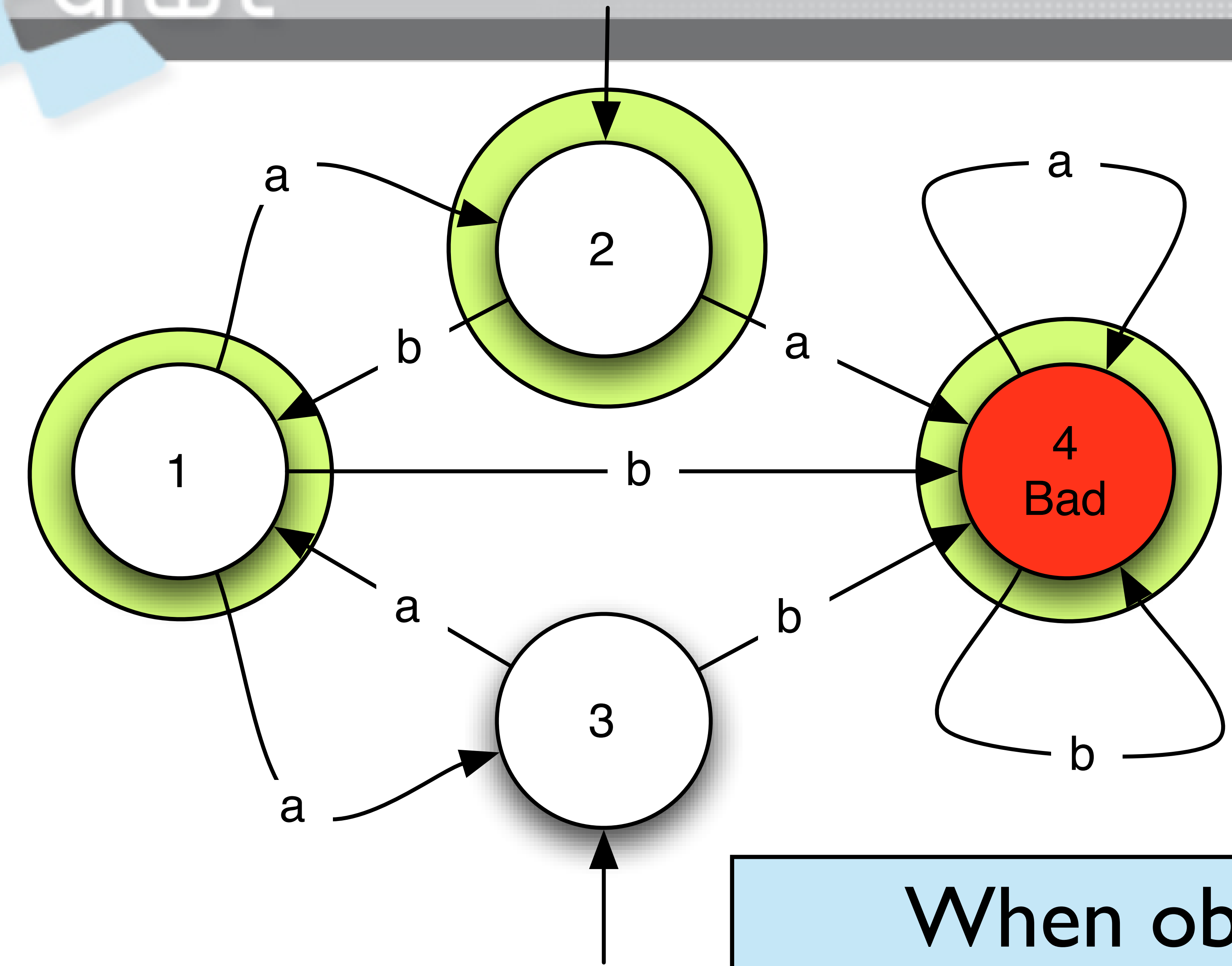


Obs 0

Obs 1

Slight generalization of incomplete information

Imperfect information



Obs 0

When observing Obs 0,
there is no unique good choice:
memory is necessary

Imperfect information

- A game of *imperfect information*:
game structure + observation structure

-*Observation structure* : **(Obs,γ)** where **Obs** is a finite set of observations and γ maps every observation to a set of states (we require that every state has at least one observation).

-A *observation based strategy* is a function that maps every sequence $o_1 \sigma_1 o_2 \ldots o_n$ to a letter in Σ.

Our objective is to find an algorithm to construct **observation based strategies** that avoid Bad.

Notation: a game structure of imperfect information is a tuple $(S,S_0,\Sigma,\rightarrow,Obs,\gamma)$.

...ation structure

observation structure : **(Obs,γ)** where **Obs** is a finite set of observations and **γ** maps every observation to a set of states (we require that every state has at least one observation).

-A *observation based strategy* is a function that maps every sequence $o_1\sigma_1 o_2 \ldots o_n$ to a letter in Σ.

Our objective is to find an algorithm to construct **observation based strategies** that avoid Bad.

Notation: a game structure of imperfect information is a tup

...tion structure

...s a finite

set of ... ation to

a set ... as at

least ...

Those games generalize games of *perfect information* where **Obs=S** and γ is the identity function

-A *observation based strategy* is a function that maps every sequence $o_1\sigma_1 o_2 \ldots o_n$ to a letter in Σ.

Our objective is to find an algorithm to construct **observation based strategies** that avoid Bad.

Notation: a game structure of imperfect information is a tup[…]

[…]tion structure

[…]s a finite

set of […]ation to

a set […]

least […]

Those games generalize games of p[…]

whe[…]

iden[…]

Those games generalize games of *incomplete information*: in that case **Obs partitions** the state space *S*. [Rei84]

-A *observa*[…]s

every sequ[…]

Our objective is to find an algorithm to construct **observation based strategies** that avoid Bad.

Classical Approaches

- To solve games of perfect information :
  - (elegant) fixed point algorithms using a controllable predecessor operator
- To solve games of imperfect information
  - [Reif84] builds a game of perfect information using a knowledge-based subset construction and then solve this games using classical techniques

Classical Approaches

- To solv
  - (elega
    contr

After a finite prefix of a game, Player I has a partial knowledge of the current state of the game : **a set of states**

- To solve games of imperfect information
  - [Reif84] builds a game of perfect information using a knowledge-based subset construction and then solve this games using classical techniques

Classical Approaches

- To solv
  - (elega
    contr

After a finite prefix of a game, Player I has a partial knowledge of the current state of the game : **a set of states**

We propose here a new solution that avoid the **preliminary** explicit subset construction.

fect information

of perfect information
ed
subset construction and then solve this games using classical techniques

We define a *controllable predecessor* operator for a ***set of sets of states*** *q*

$$\mathsf{CPre}(q) = \{s \subseteq \overline{\mathsf{Bad}} \mid \exists \sigma \in \Sigma \cdot \forall \mathsf{obs} \in \mathsf{Obs} \cdot \exists s' \in q : \mathsf{Post}_\sigma(s) \cap \gamma(\mathsf{obs}) \subseteq s'\}$$

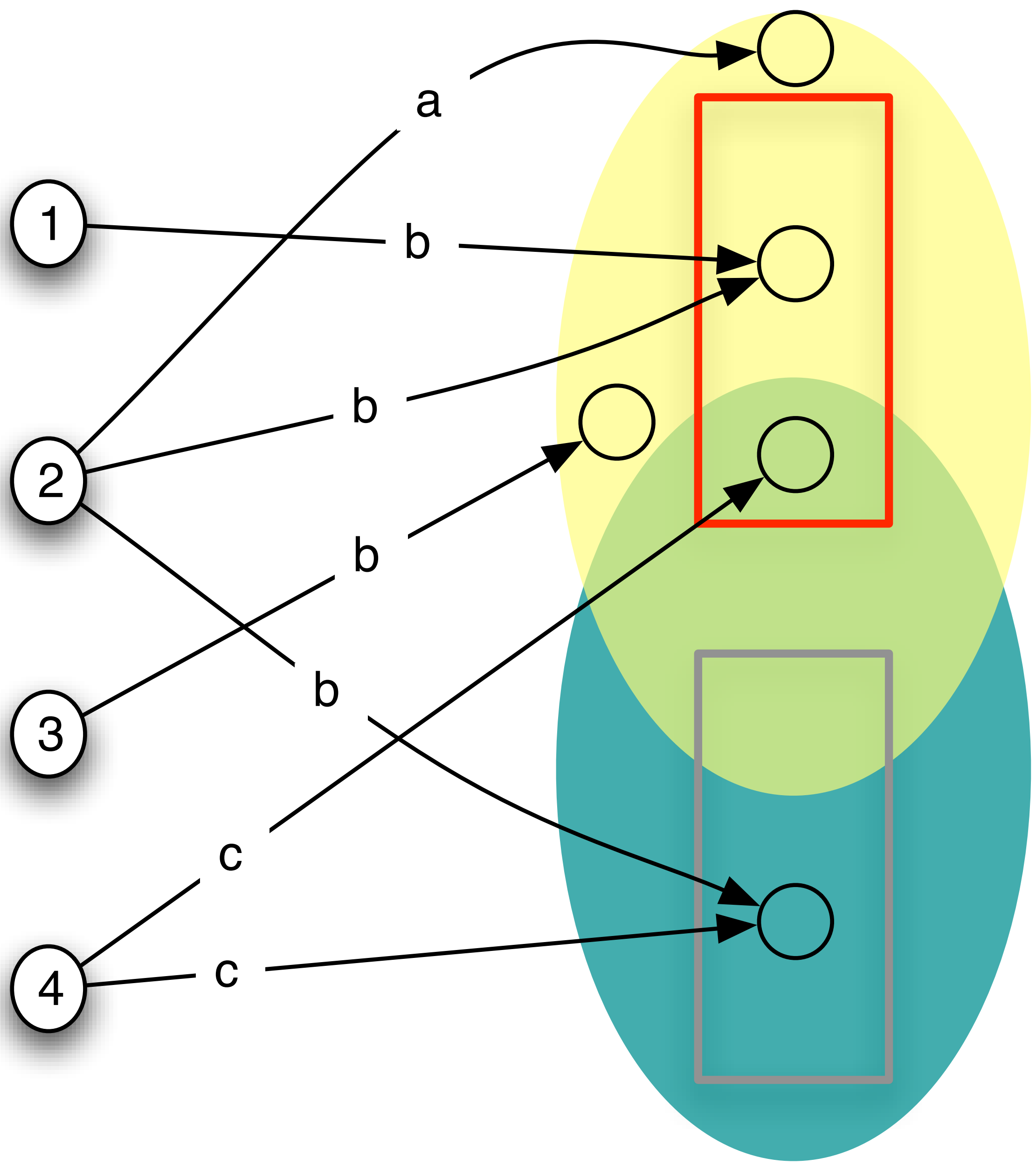
(i) *s* does not intersect with **Bad**,

(ii) there exists σ s.t. the set of possible successors of s by σ is covered by *q*

(a) no matter how the adversary resolves non-determinism,

(b) no matter the compatible observation **Obs**

Example

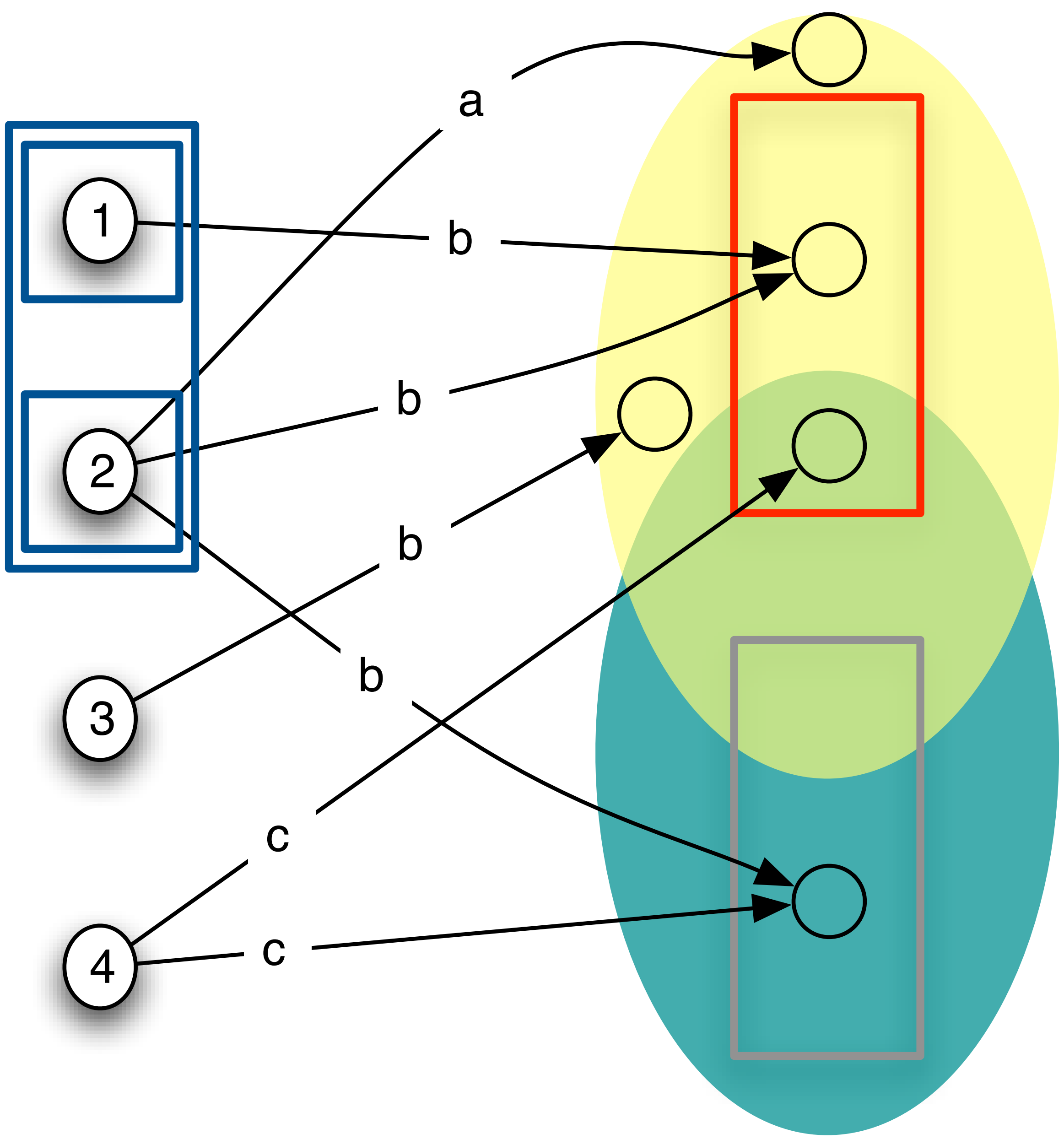


$q = \{A, B\}$

Obs 1

Obs 2

Example



q ={A, B}

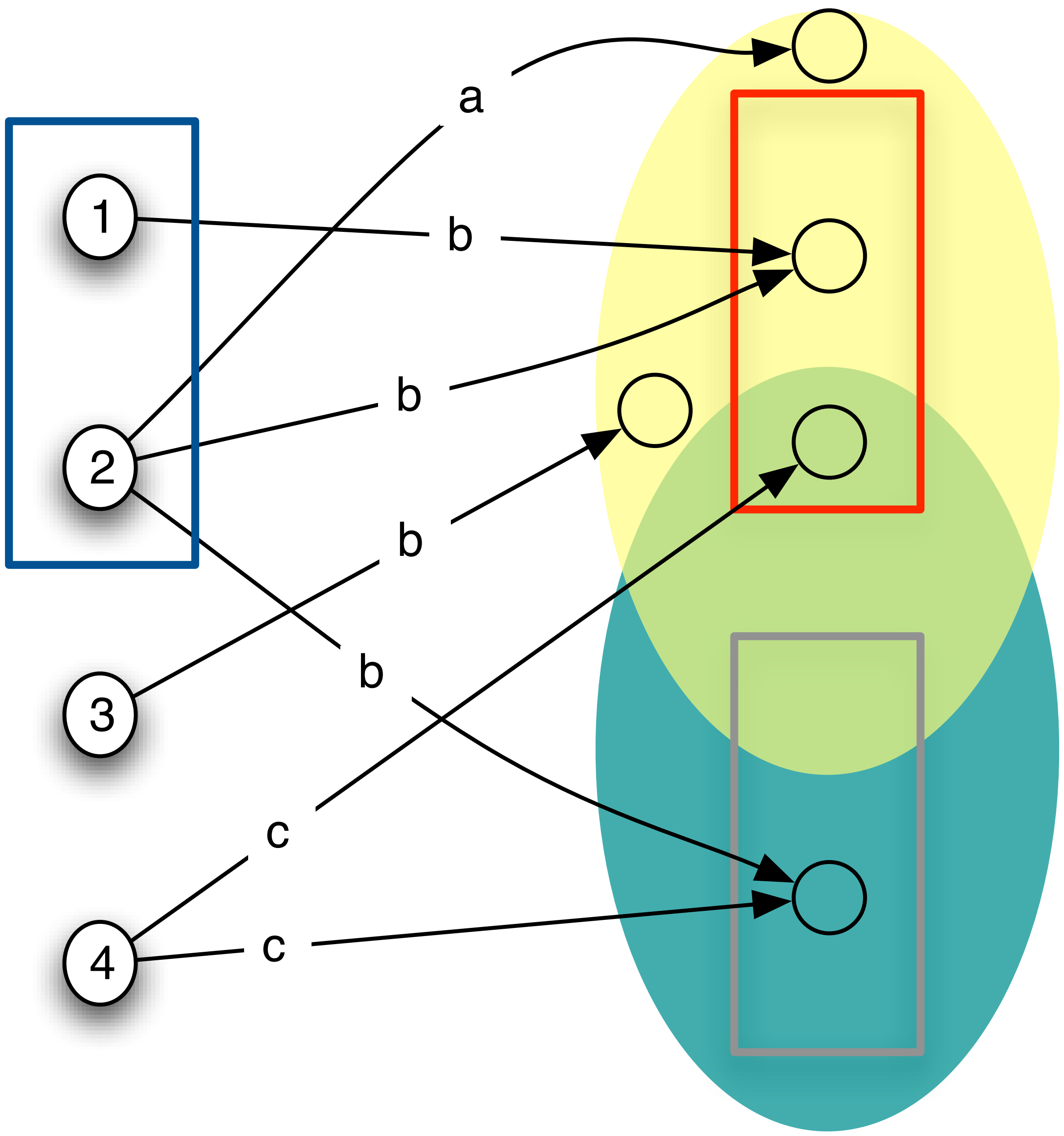
Obs 1

Obs 2

Cpre({A,B})= Blue sets

Example

If there is a strategy for set A,
there is a strategy for any B included in A



It is enough to keep only the **maximal sets**

$$\mathsf{CPre}(q) = [\{s \subseteq \overline{\mathsf{Bad}} \mid \exists \sigma \in \Sigma \cdot \forall \mathsf{obs} \in \mathsf{Obs} \cdot \exists s' \in q : \mathsf{Post}_\sigma(s) \cap \gamma(\mathsf{obs}) \subseteq s'\}]$$

Antichains

Definition 4 [Antichain of sets of states] An *antichain* on the partially ordered set $\langle 2^S, \subseteq \rangle$ is a set $q \subseteq 2^S$ such that for any $A, B \in q$ we have $A \not\subset B$.

Let us call L the set of antichains on S.

Definition 5 [$\sqsubseteq$] Let $q, q' \in 2^{2^S}$ and define $q \sqsubseteq q'$ if and only if

$$\forall A \in q : \exists A' \in q' : A \subseteq A'$$

$$\textbf{lub}: \quad q_1 \sqcup q_2 = \lceil \{ s \mid s \in q_1 \vee s \in q_2 \} \rceil$$

$$\textbf{glb}: \quad q_1 \sqcap q_2 = \lceil \{ s_1 \cap s_2 \mid s_1 \in q_1 \wedge s_2 \in q_2 \} \rceil$$

The minimal element is $\emptyset$, the maximal element $\{S\}$.

$\langle L, \sqsubseteq \rangle$ is a complete lattice.

CPre over antichains

$$\mathsf{CPre}(q) = [\{s \subseteq \overline{\mathsf{Bad}} \mid \exists \sigma \in \Sigma \cdot \forall \mathsf{obs} \in \mathsf{Obs} \cdot \exists s' \in q : \mathsf{Post}_\sigma(s) \cap \gamma(\mathsf{obs}) \subseteq s'\}]$$

- CPre is a **monotone** function over the lattice of antichains
- CPre has a *least* and a *greatest* fixed point

Advantage : we only keep the needed information to find a strategy

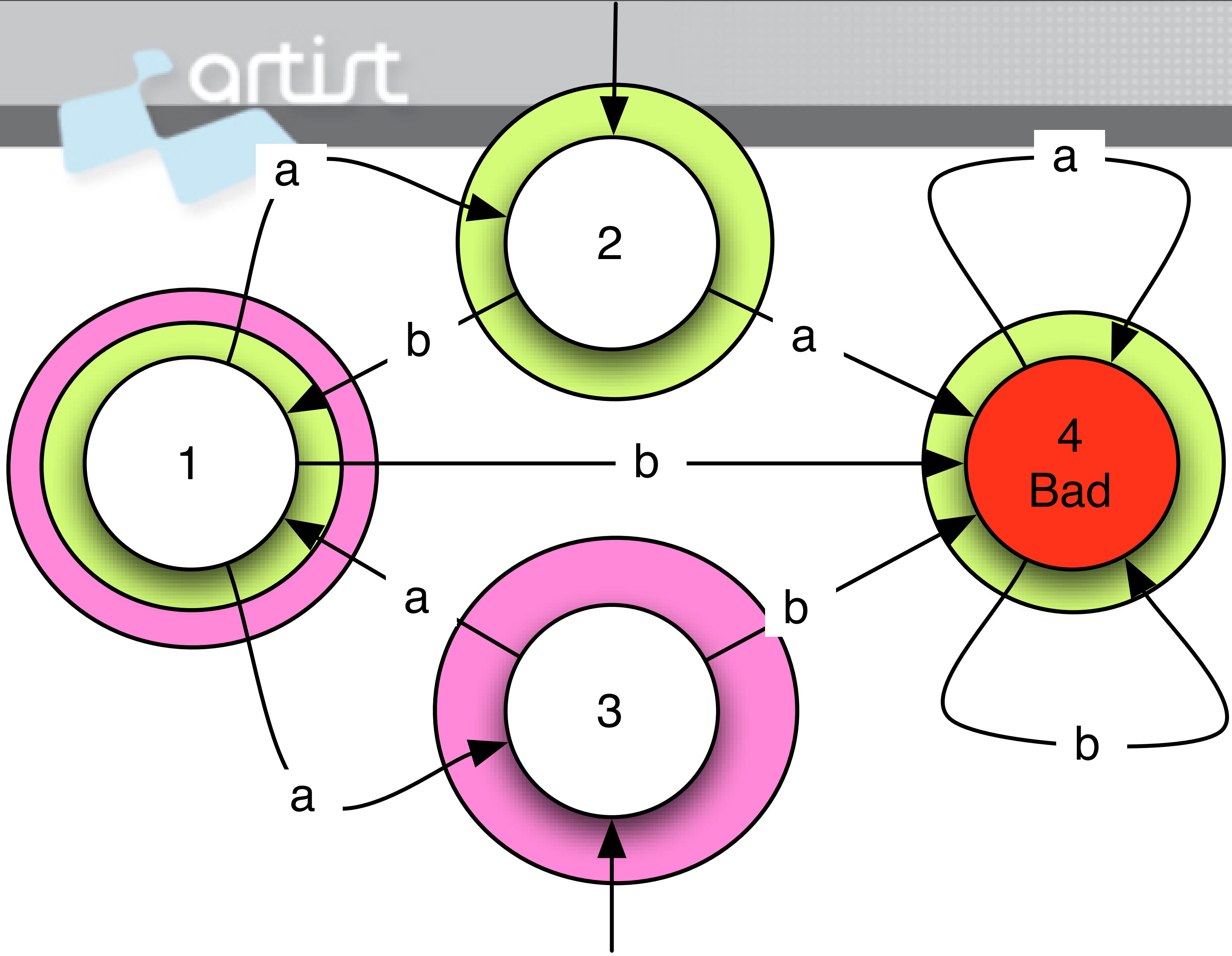
Main theorem

Let $G = \langle S, S_0, \Sigma, \rightarrow, \mathsf{Obs}, \gamma \rangle$

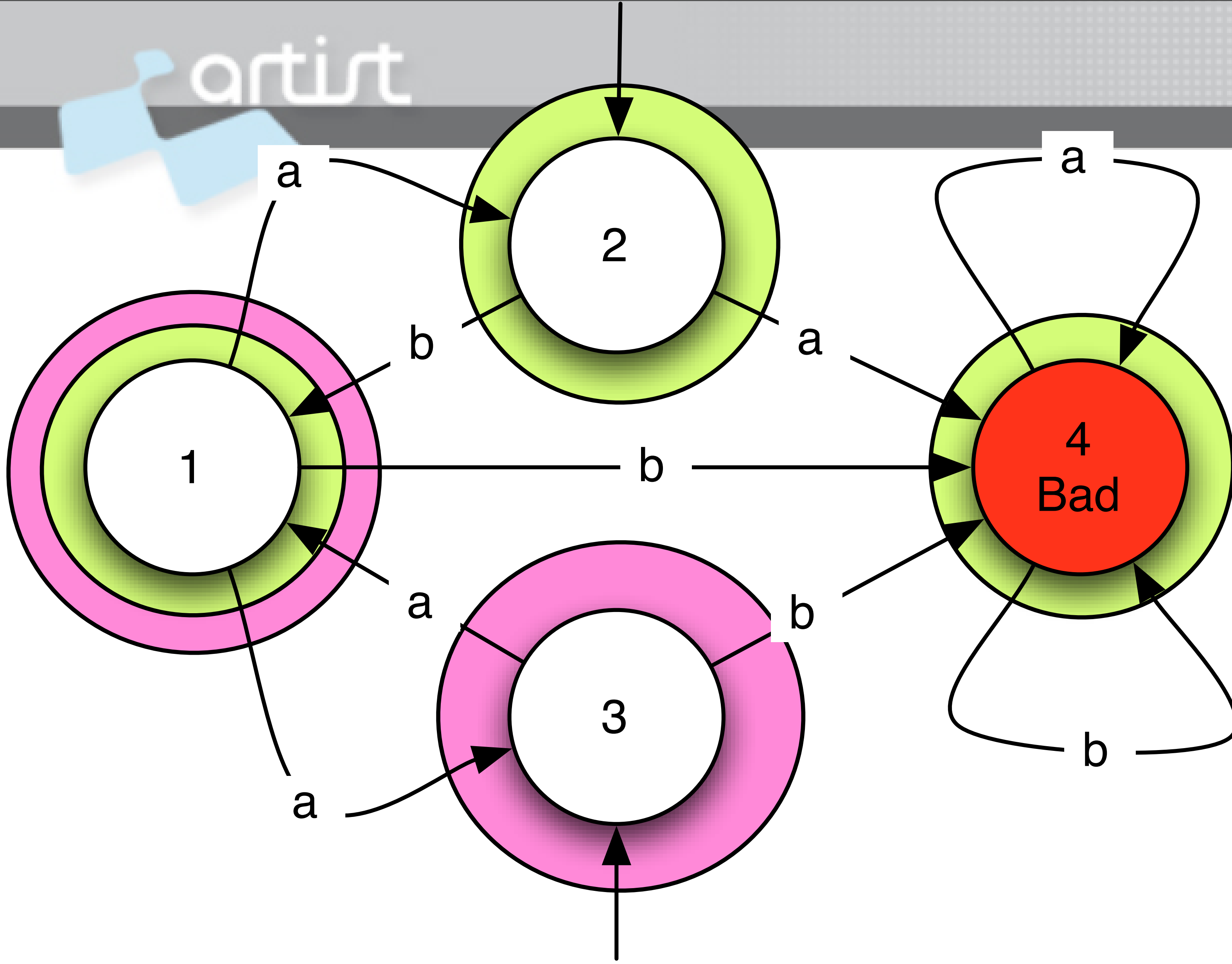
be a two-player game of imperfect information. Player 1 has a winning observation based strategy to avoid Bad, **iff**

$$\{S_0 \cap \gamma(\mathsf{obs}) \mid \mathsf{obs} \in \mathsf{Obs}\} \sqsubseteq \bigsqcup \{q \mid q = \mathsf{CPre}(q)\}.$$

We can extract a strategy from the fixed point

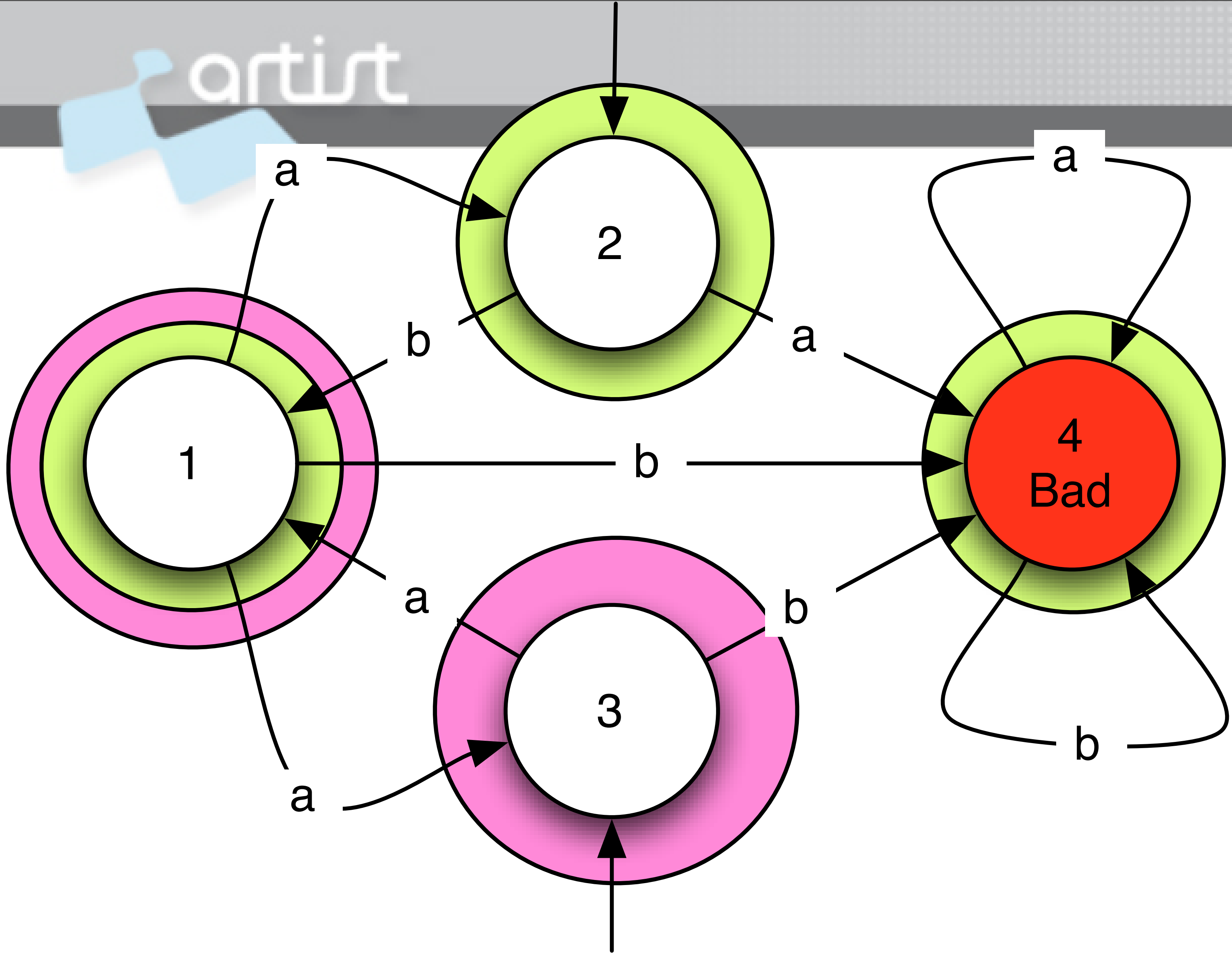


Does Player 0 have an observation based strategy to avoid Bad ?



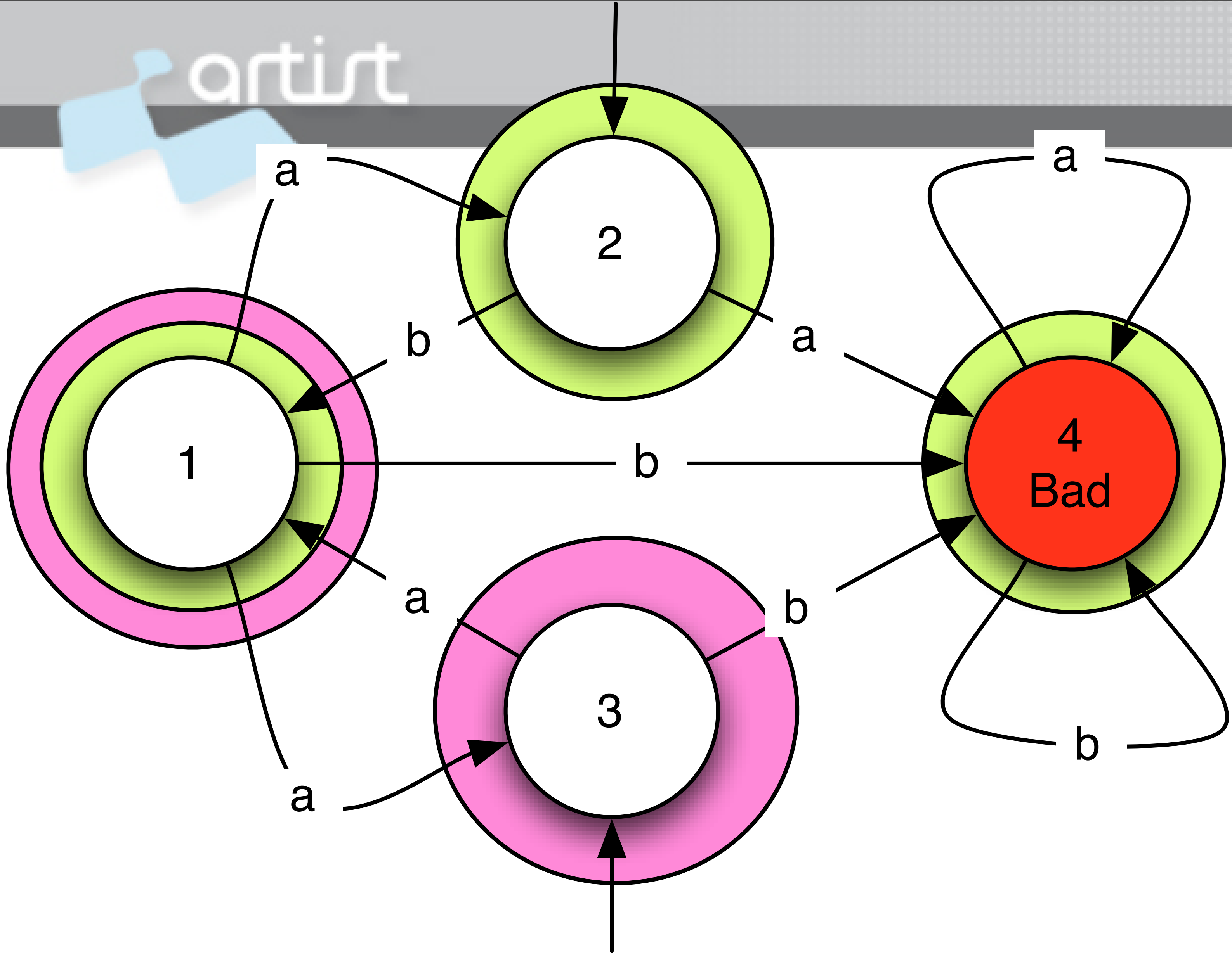
Does Player 0 have an observation based strategy to avoid Bad ?

Let us compute the *gfp* of CPre over L.



$$q_0 = \top$$

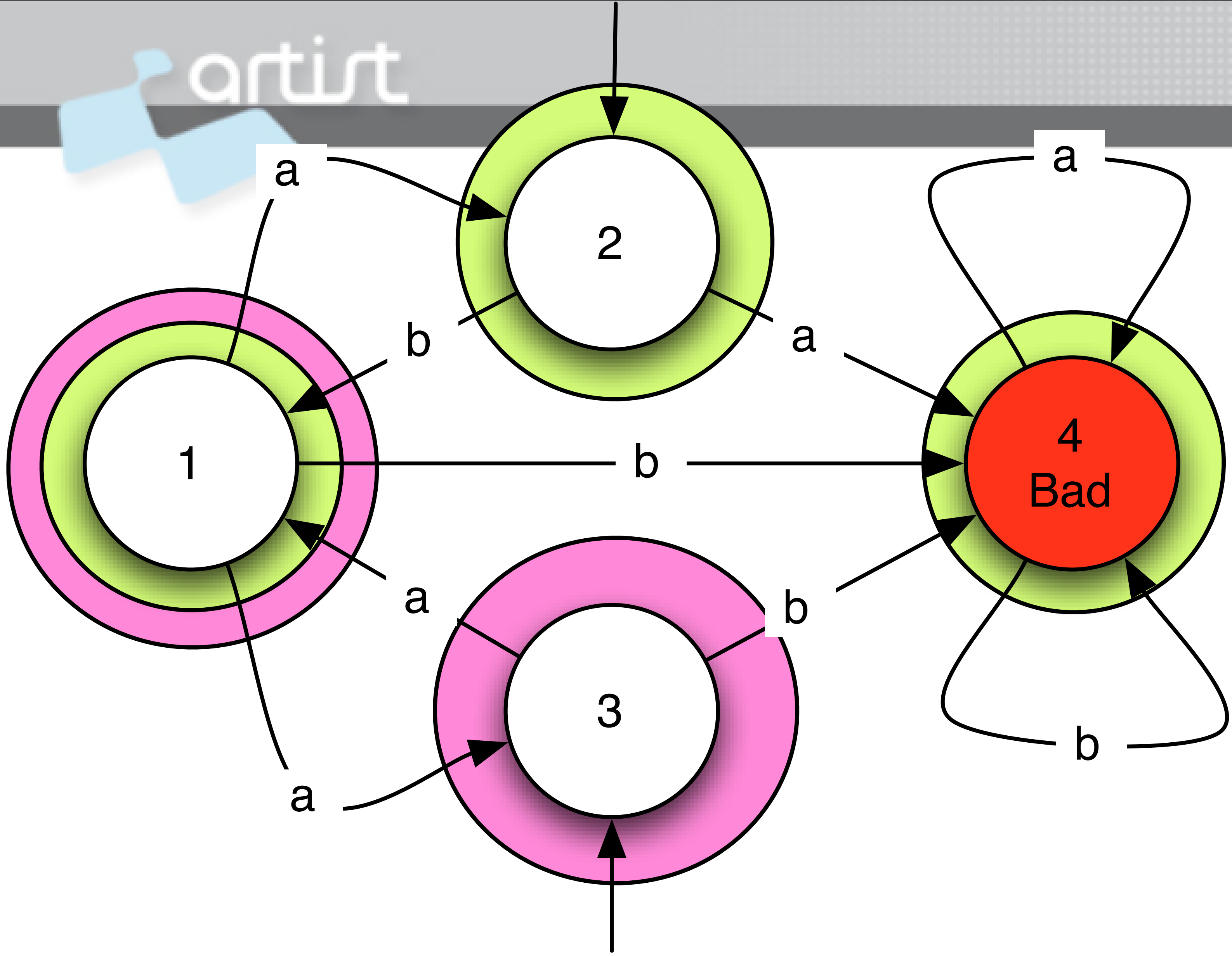
$$q_1 = \{\{1, 2, 3\}_{a,b}\}$$



$$q_0 = \top$$

$$q_1 = \{\{1,2,3\}_{a,b}\}$$

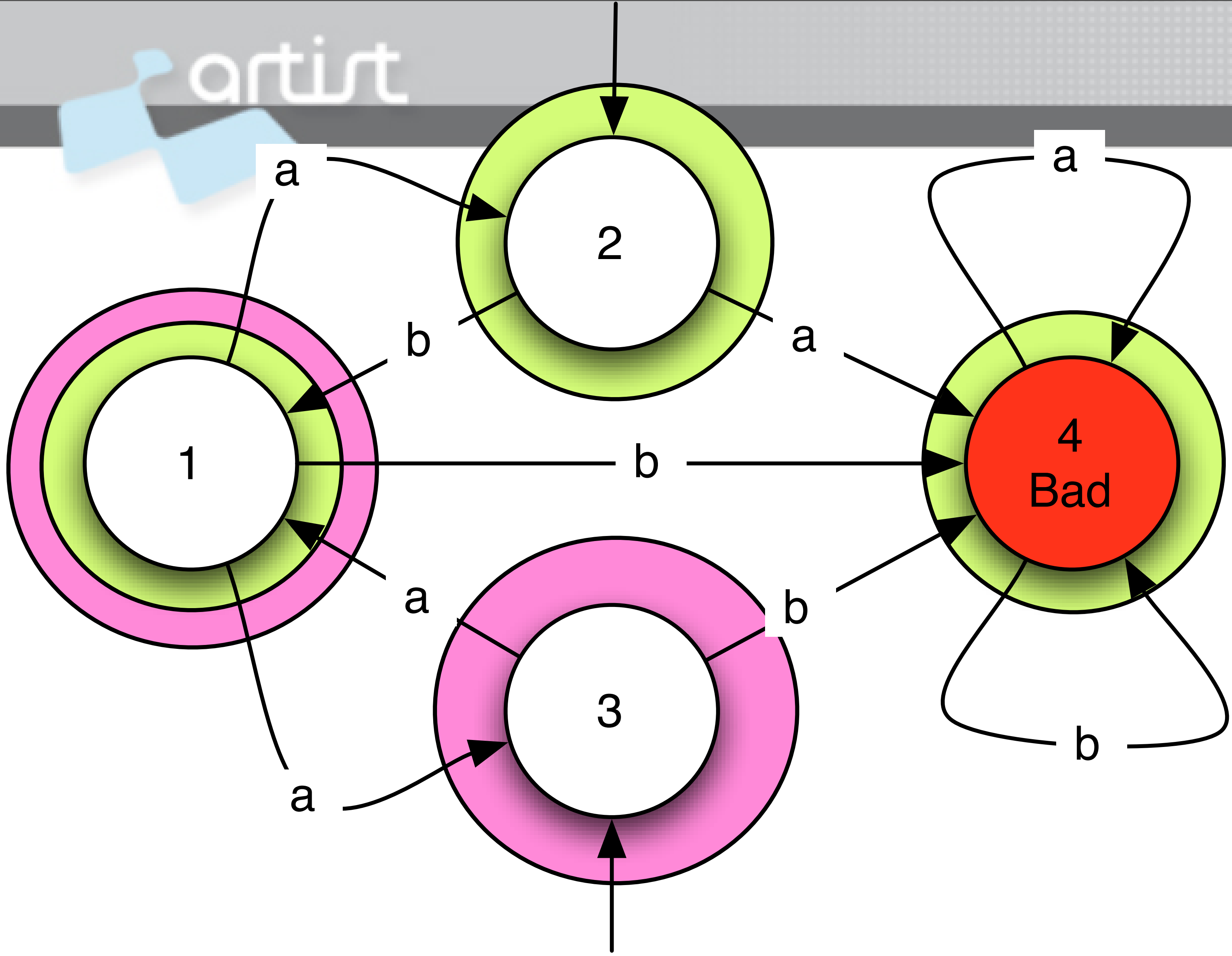
$$q_2 = \mathsf{CPre}(\{\{1,2,3\}\})$$



$q_0 = \top$

$q_1 = \{\{1,2,3\}_{a,b}\}$

$$q_2 = \mathsf{CPre}(\{\{1,2,3\}\})$$
$$= \{\{2\}_b, \{1,3\}_a\}$$



$$q_0 = \top$$

$$q_1 = \{\{1,2,3\}_{a,b}\}$$

$$q_2 = \mathsf{CPre}(\{\{1,2,3\}\})$$

$$= \{\{2\}_b, \{1,3\}_a\}$$

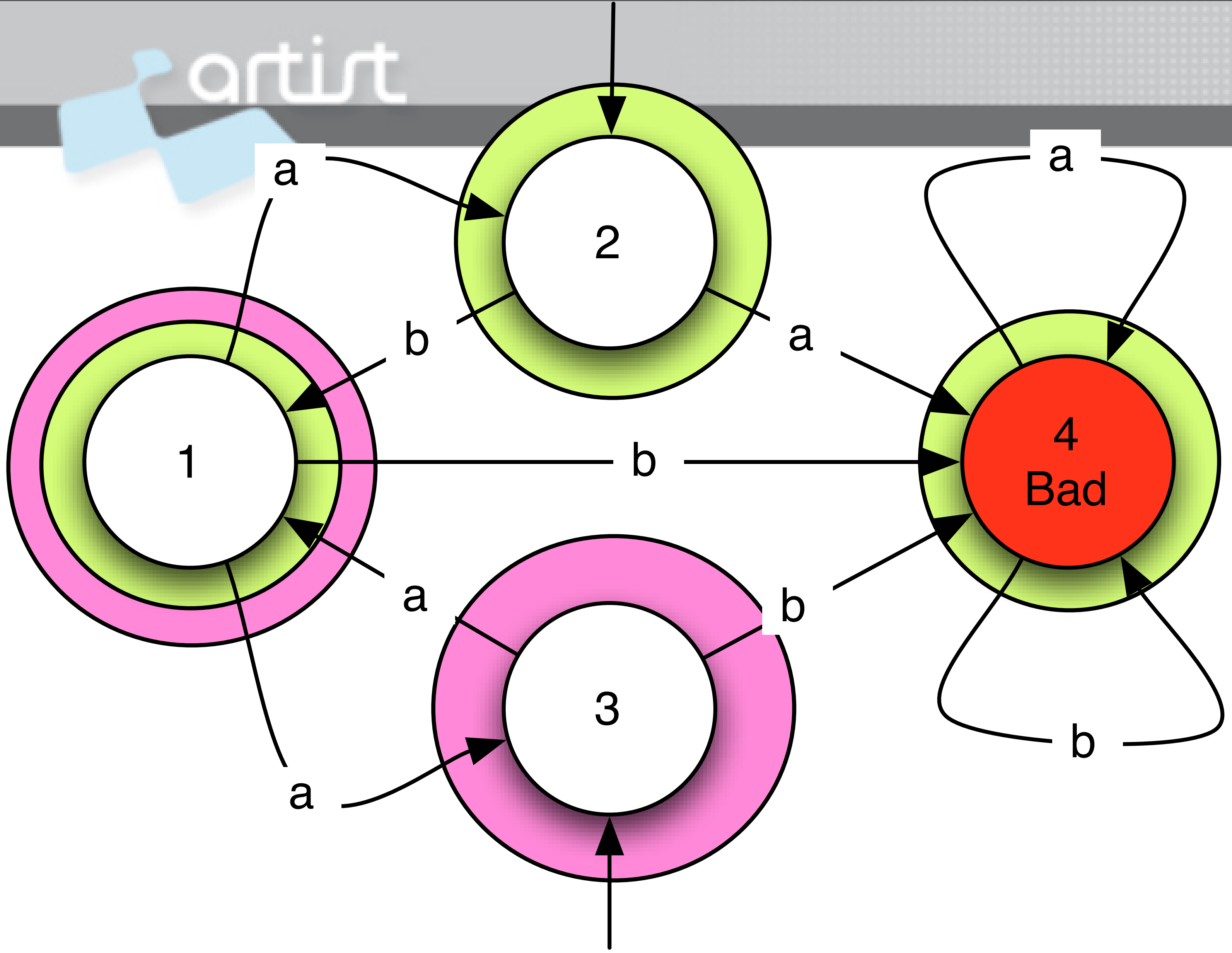
Indeed,

$$\mathsf{Post}_a(\{1,3\}) \cap \{1,2,4\} \subseteq \{1,2,3\}$$

$$\mathsf{Post}_a(\{1,3\}) \cap \{1,3\} \subseteq \{1,2,3\}$$

$$\mathsf{Post}_b(\{2\}) \cap \{1,3\} \subseteq \{1,2,3\}$$

$$\mathsf{Post}_b(\{2\}) \cap \{1,2,4\} \subseteq \{1,2,3\}$$

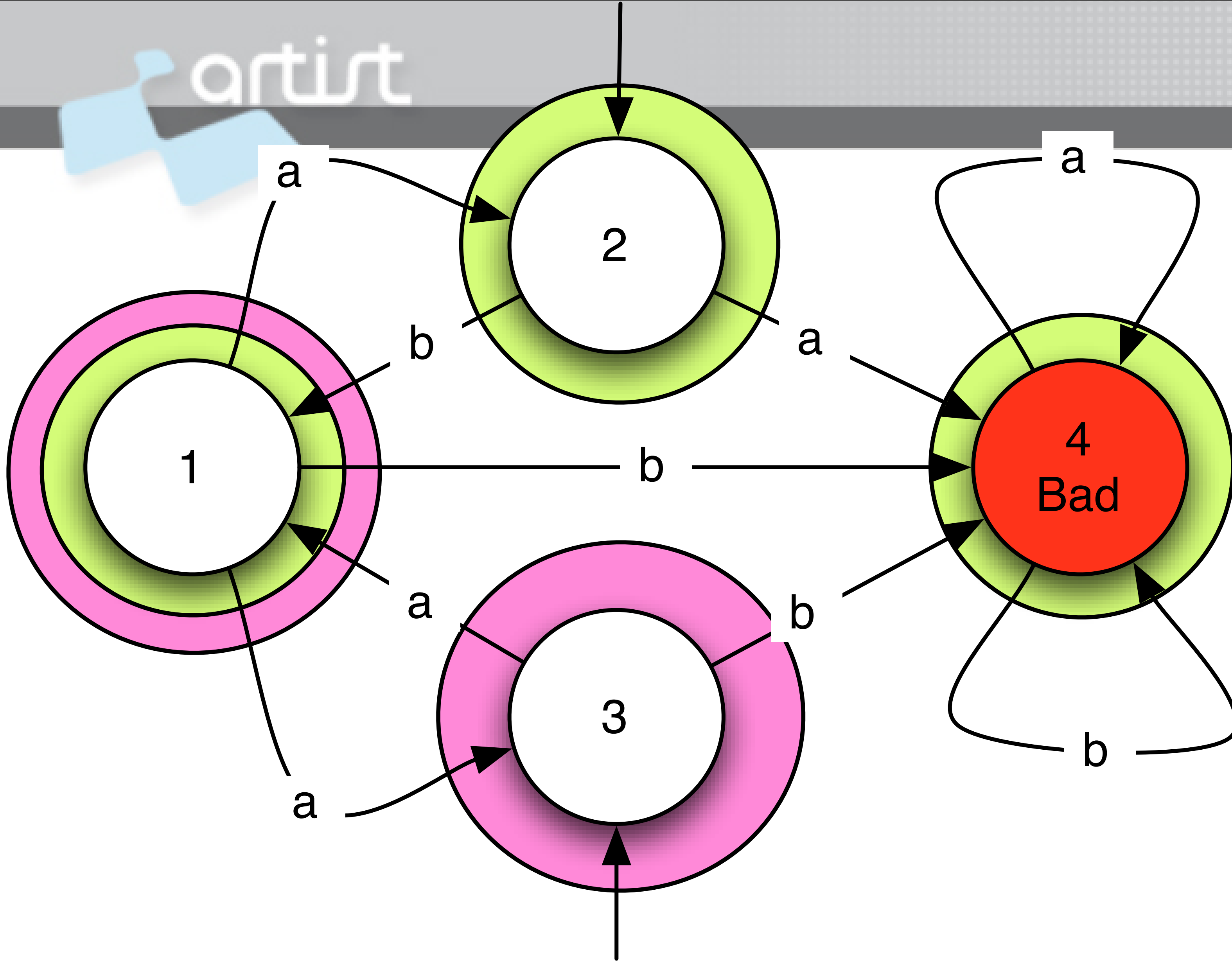


$$q_0 = \top$$

$$q_1 = \{\{1,2,3\}_{a,b}\}$$

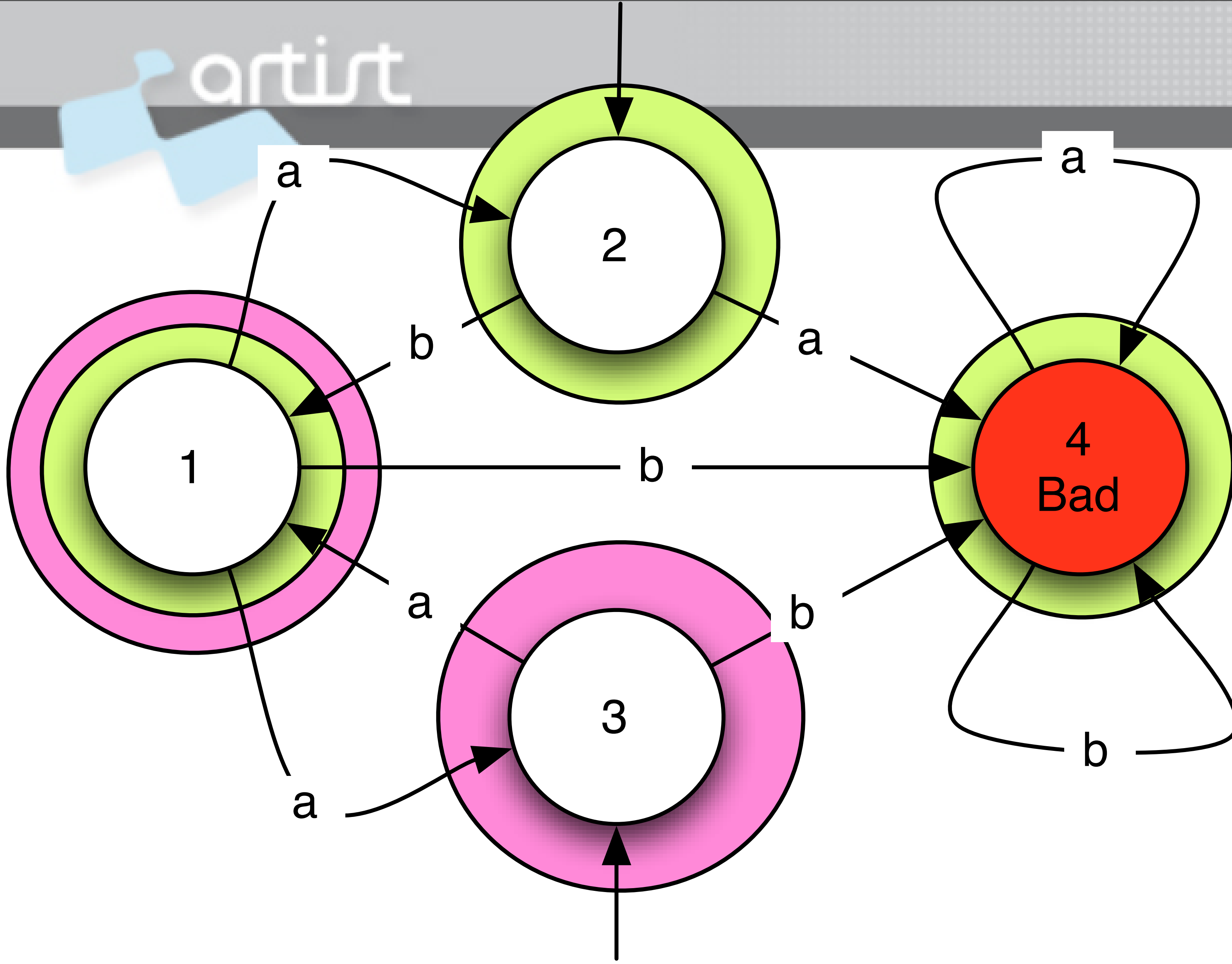
$$q_2 = \{\{2\}_b, \{1,3\}_a\}$$

$$q_3 = \mathsf{CPre}(\{\{2\},\{1,3\}\})$$



$$q_0 = \top$$
$$q_1 = \{\{1,2,3\}_{a,b}\}$$
$$q_2 = \{\{2\}_b, \{1,3\}_a\}$$

$$q_3 = \mathsf{CPre}(\{\{2\},\{1,3\}\})$$
$$= \{\{1\}_a, \{2\}_b, \{3\}_a\}$$



$$q_0 = \top$$
$$q_1 = \{\{1,2,3\}_{a,b}\}$$
$$q_2 = \{\{2\}_b, \{1,3\}_a\}$$

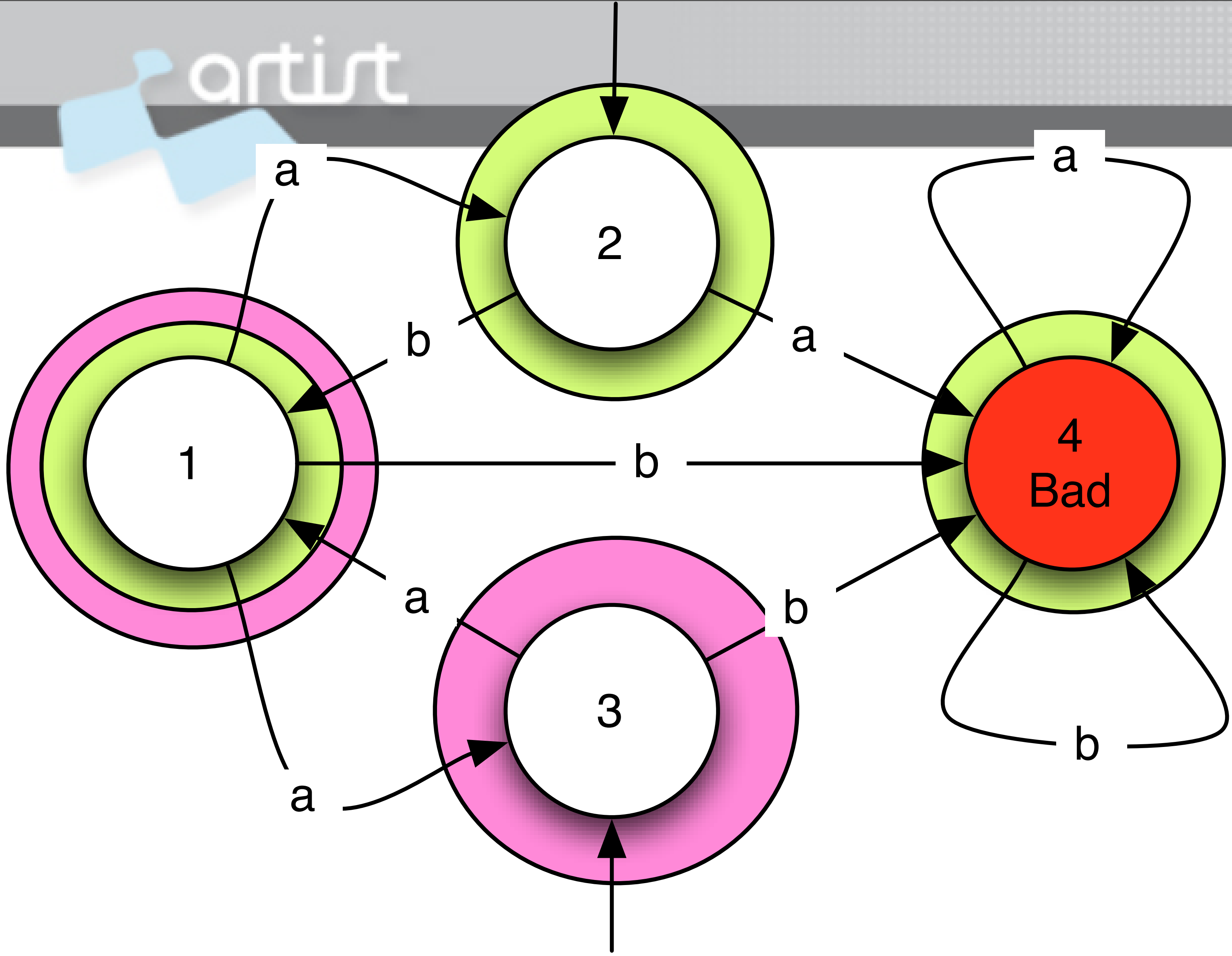
$$q_3 = \mathsf{CPre}(\{\{2\},\{1,3\}\})$$
$$= \{\{1\}_a, \{2\}_b, \{3\}_a\}$$

Indeed,

$\mathsf{Post}_a(\{1\}) \cap \{1,2,4\} \subseteq \{2\}$

$\mathsf{Post}_a(\{1\}) \cap \{1,3\} \subseteq \{3\}$

Adding any state would break this property



$$q_0 = \top$$

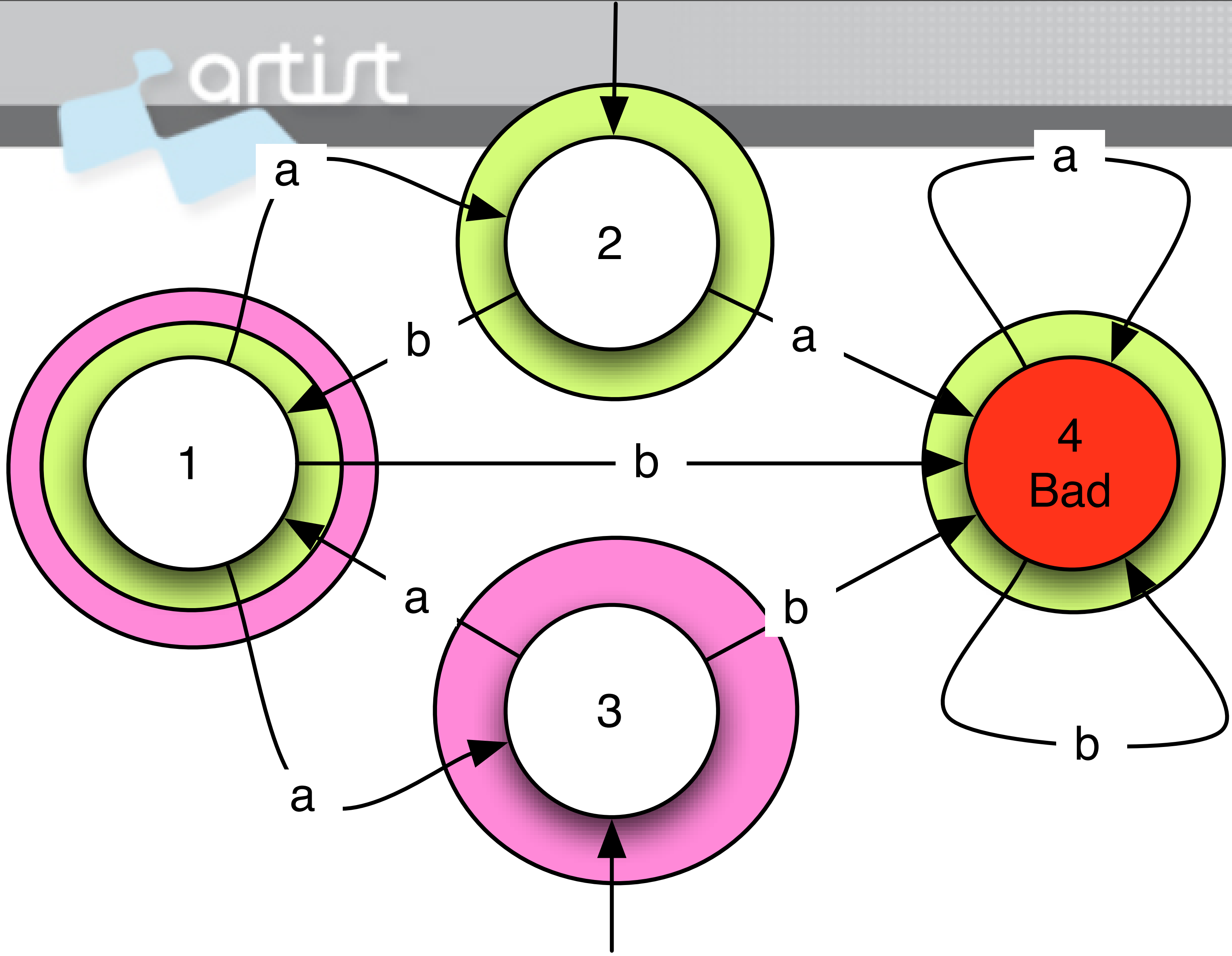
$$q_1 = \{\{1,2,3\}_{a,b}\}$$

$$q_2 = \{\{2\}_b, \{1,3\}_a\}$$

$$q_3 = \{\{1\}_a, \{2\}_b, \{3\}_a\}$$

$$q_4 = \{\{1\}_a, \{2\}_b, \{3\}_a\}$$

Fixed point



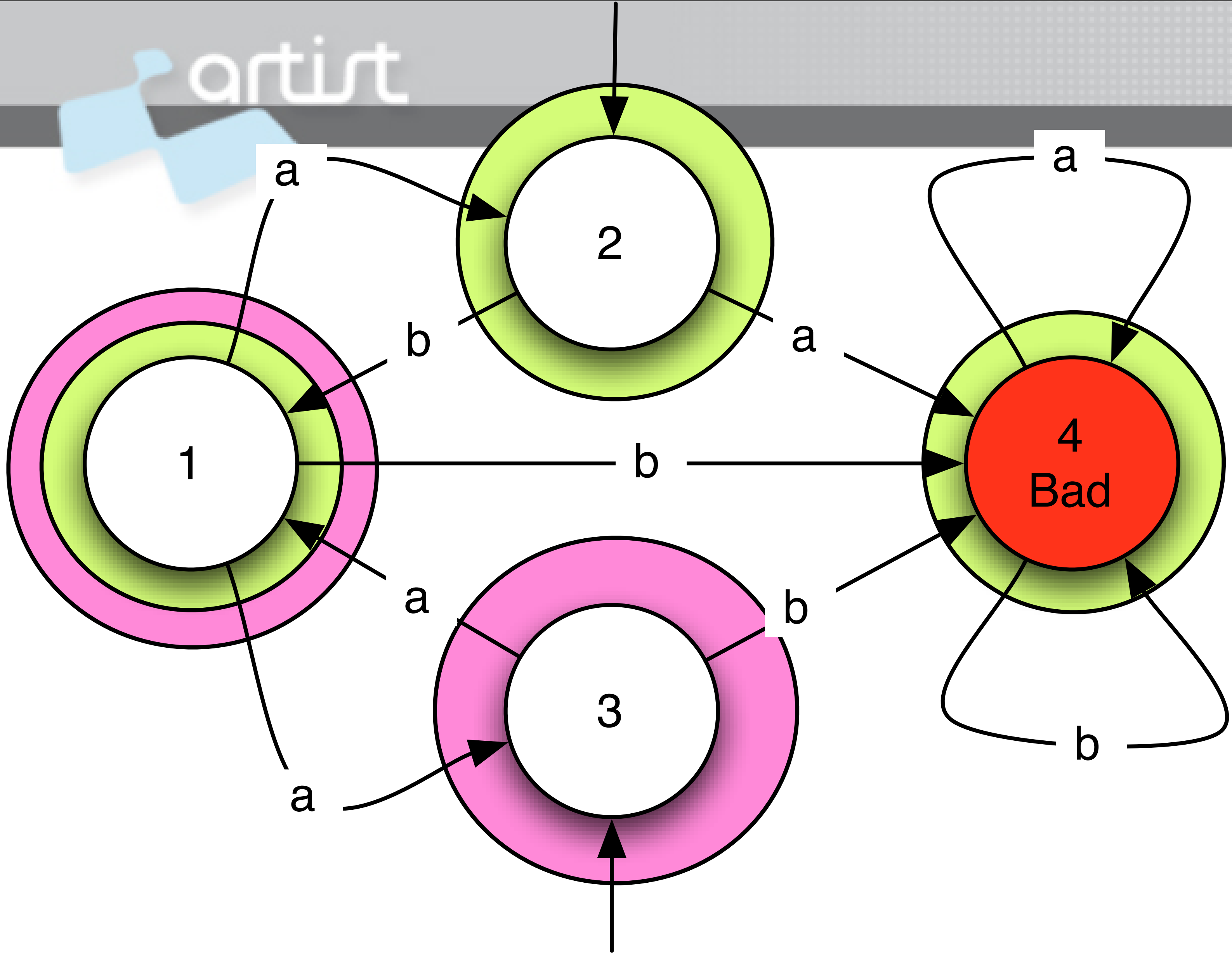
$$q_0 = \top$$
$$q_1 = \{\{1,2,3\}_{a,b}\}$$
$$q_2 = \{\{2\}_b, \{1,3\}_a\}$$
$$q_3 = \{\{1\}_a, \{2\}_b, \{3\}_a\}$$
$$q_4 = \{\{1\}_a, \{2\}_b, \{3\}_a\}$$

Fixed point

We have

$$\{\{2,3\} \cap \mathsf{Obs}_0, \{2,3\} \cap \mathsf{Obs}_1\} \sqsubseteq \sqcup\{q \mid q = \mathsf{CPre}(q)\}$$

and so, Player 0 has an observation based winning strategy to avoid Bad



$$q_0 = \top$$

$$q_1 = \{\{1,2,3\}_{a,b}\}$$

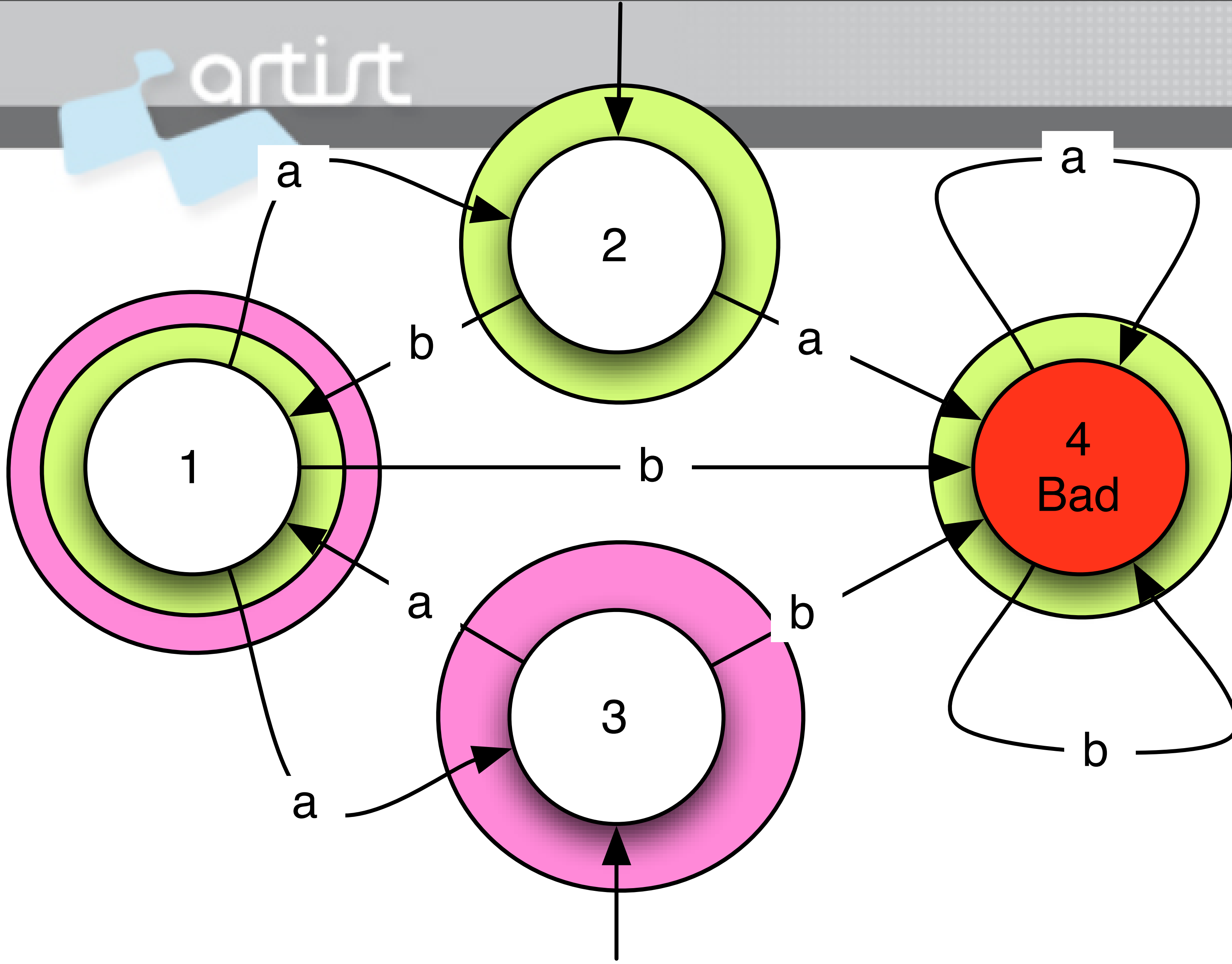
$$q_2 = \{\{2\}_b, \{1,3\}_a\}$$

$$q_3 = \{\{1\}_a, \{2\}_b, \{3\}_a\}$$

$$q_4 = \{\{1\}_a, \{2\}_b, \{3\}_a\}$$

Fixed point

We can extract a strategy from the fixed point



$$q_0 = \top$$

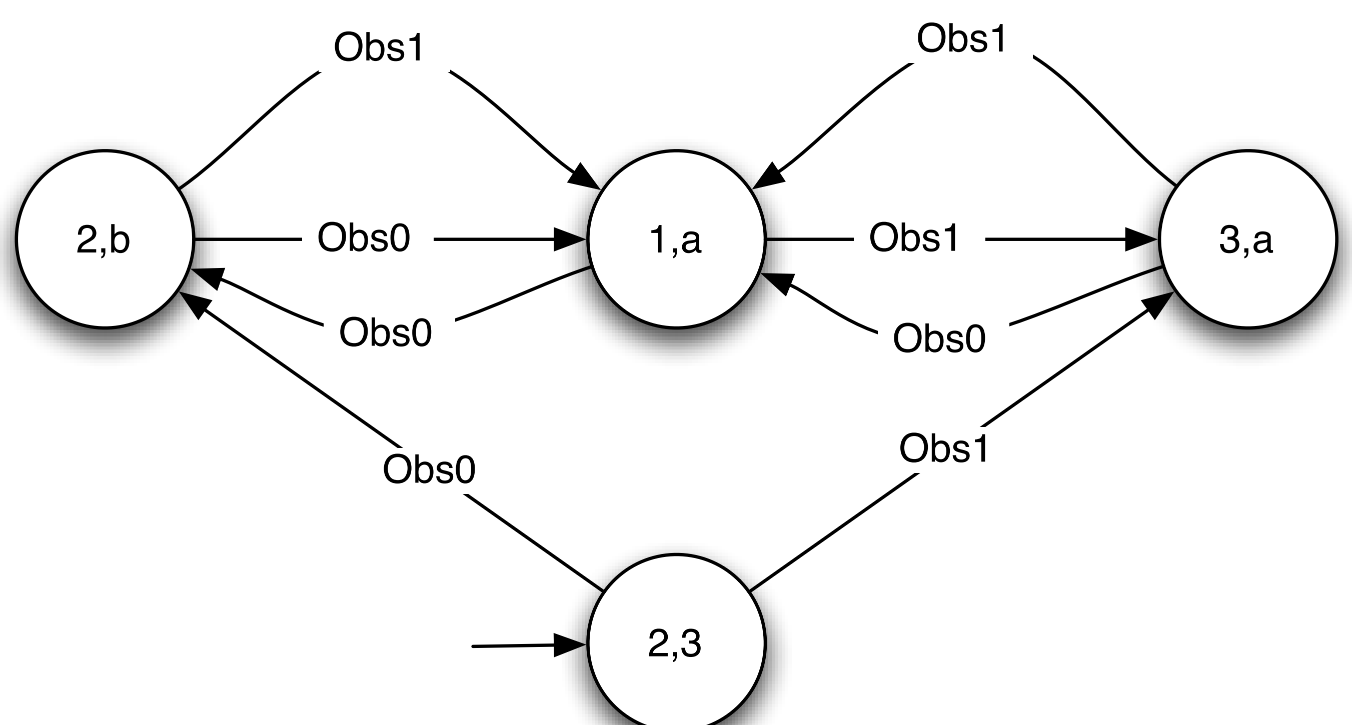
$$q_1 = \{\{1,2,3\}_{a,b}\}$$

$$q_2 = \{\{2\}_b, \{1,3\}_a\}$$

$$q_3 = \{\{1\}_a, \{2\}_b, \{3\}_a\}$$

$$q_4 = \{\{1\}_a, \{2\}_b, \{3\}_a\}$$

Fixed point



Complexity for finite state games

- The imperfect information control problem is *EXPTIME-complete*
- There exist finite state games of incomplete information for which the algorithm of [Rei84] requires an exponential time where our algorithm needs only polynomial time

Complexity for finite state games

- The imperfect information control problem is *EXPTIME-complete*
- There exist finite state games of incomplete information for which the algorithm of [Rei84] requires a
our algorithm nee

We compute exactly what is needed to control the system for a given objective

Infinite state games

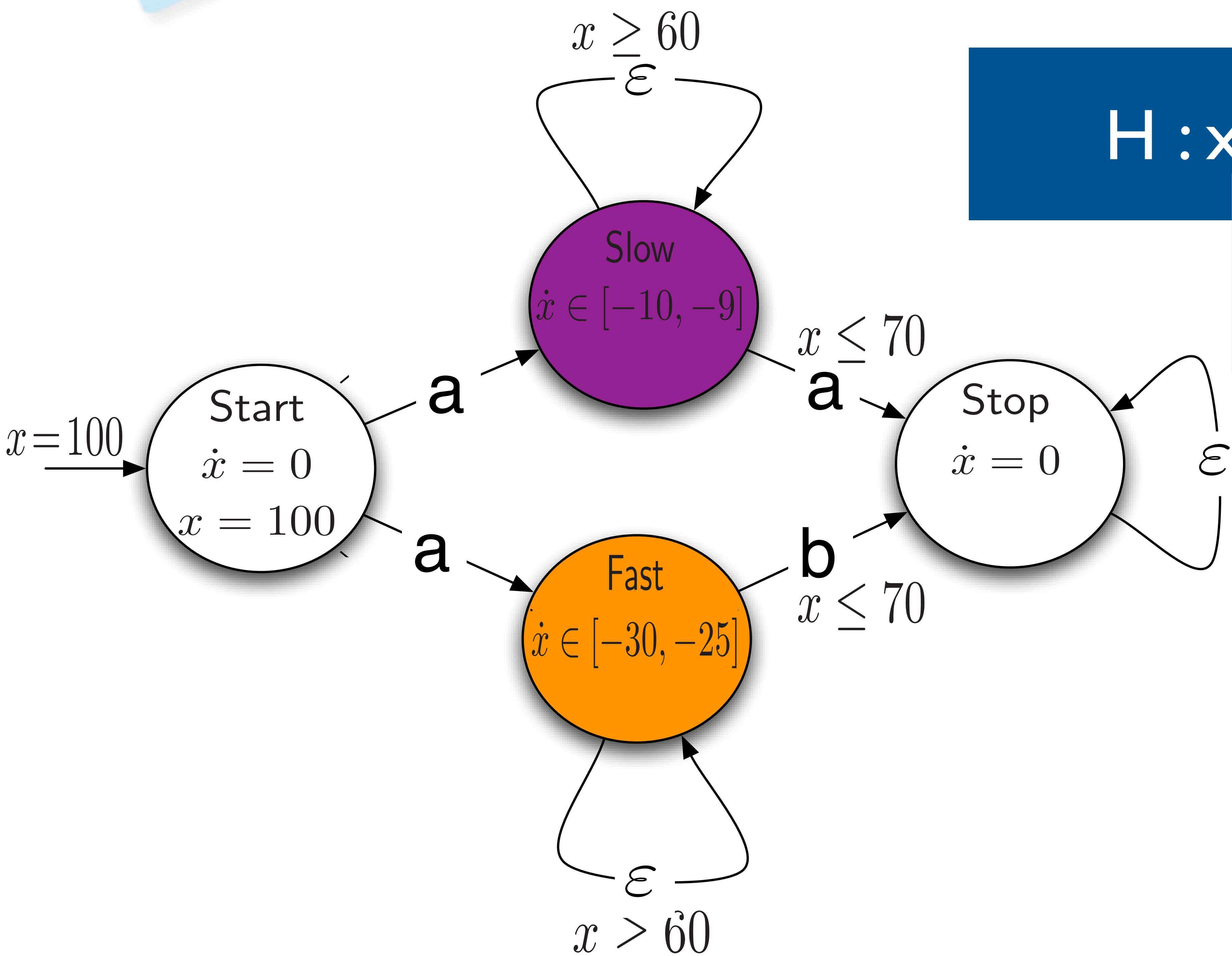
We drop the assumption that S if finite

Our fixed point algorithm will terminate if

There exists a finite quotient of the state space
Post, Enabled, γ are definable using this quotient

Application : Discrete Time Control of RHA

Discrete time control of RHA



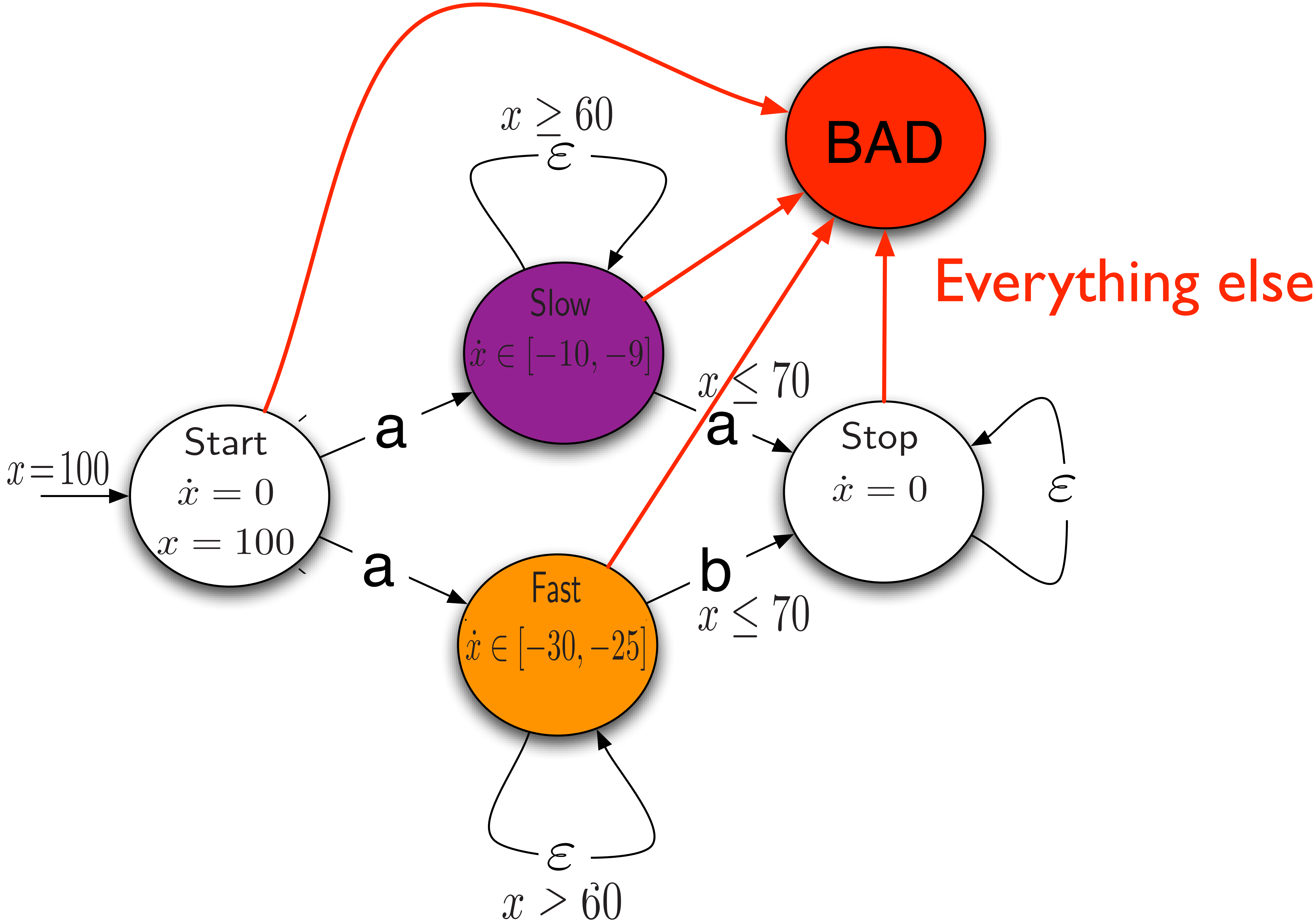
H : x >=80

L : x <=85

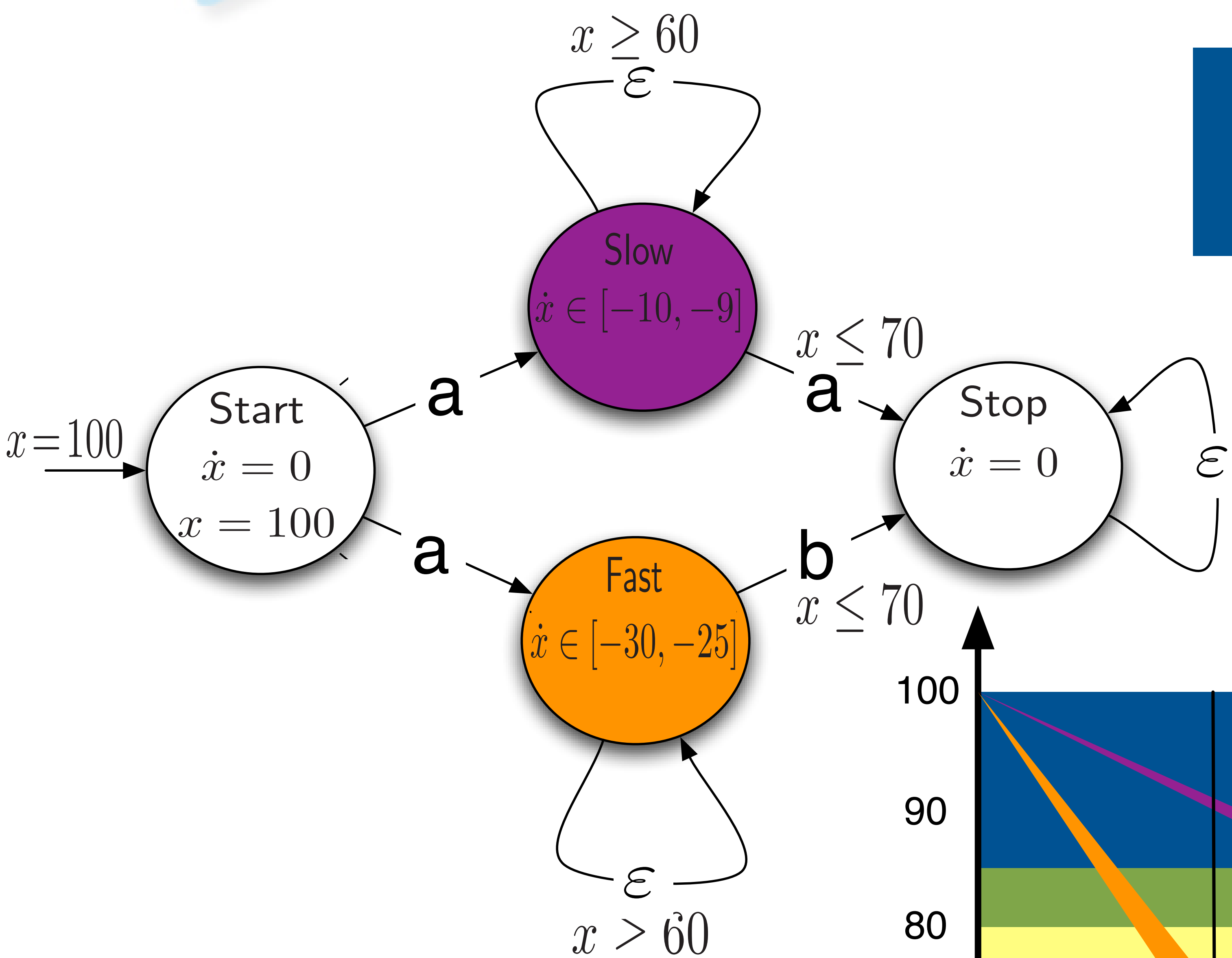
Player 1 (contr.) chooses an action every 1 time unit

Player 2 (env.) resolves nondeterminism
(in discrete and continuous steps).

Discrete time control of RHA

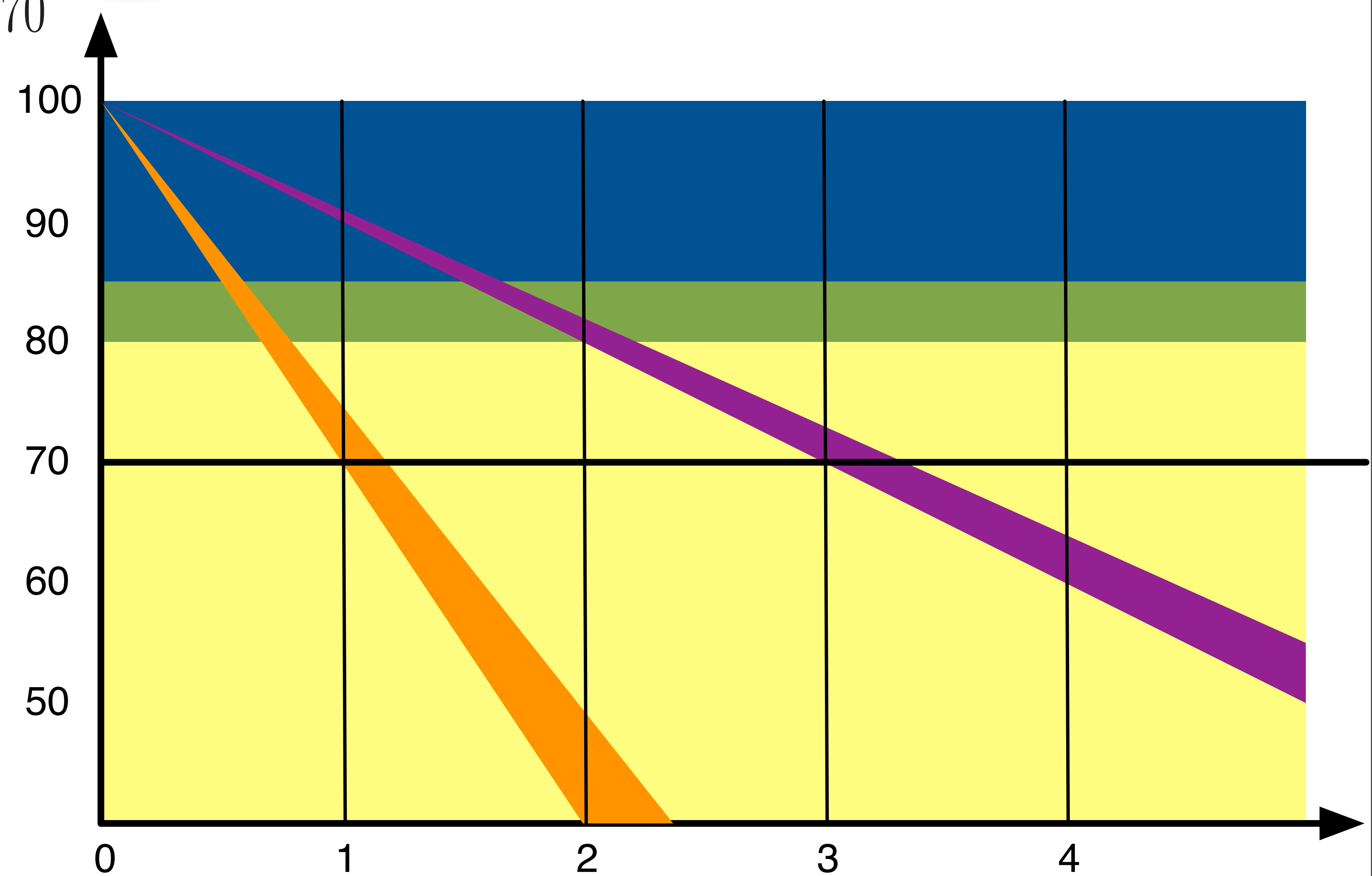


Discrete time control of RHA

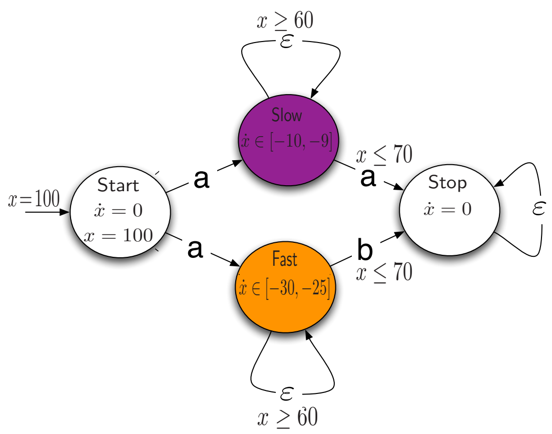


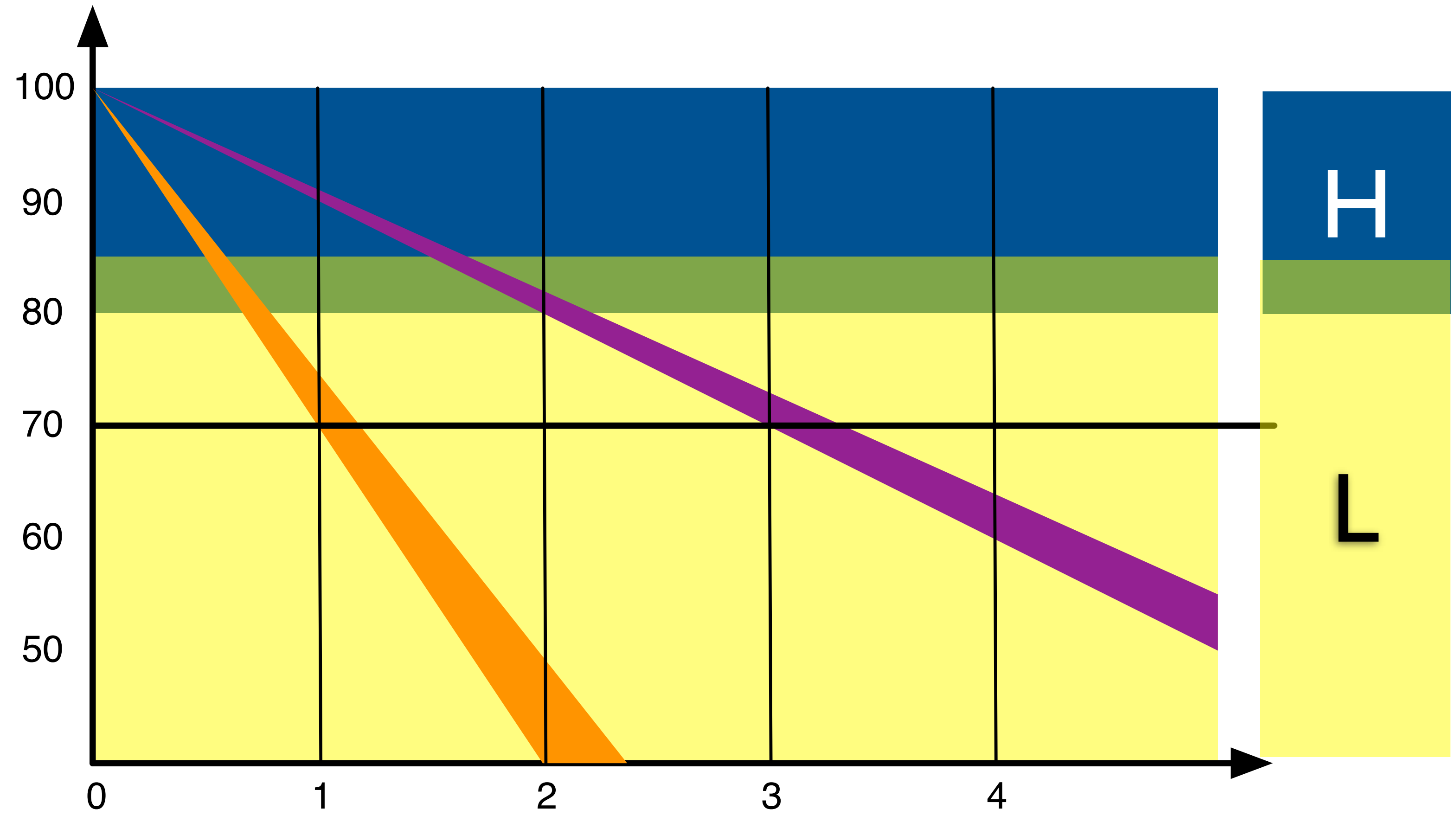
H : x >=80

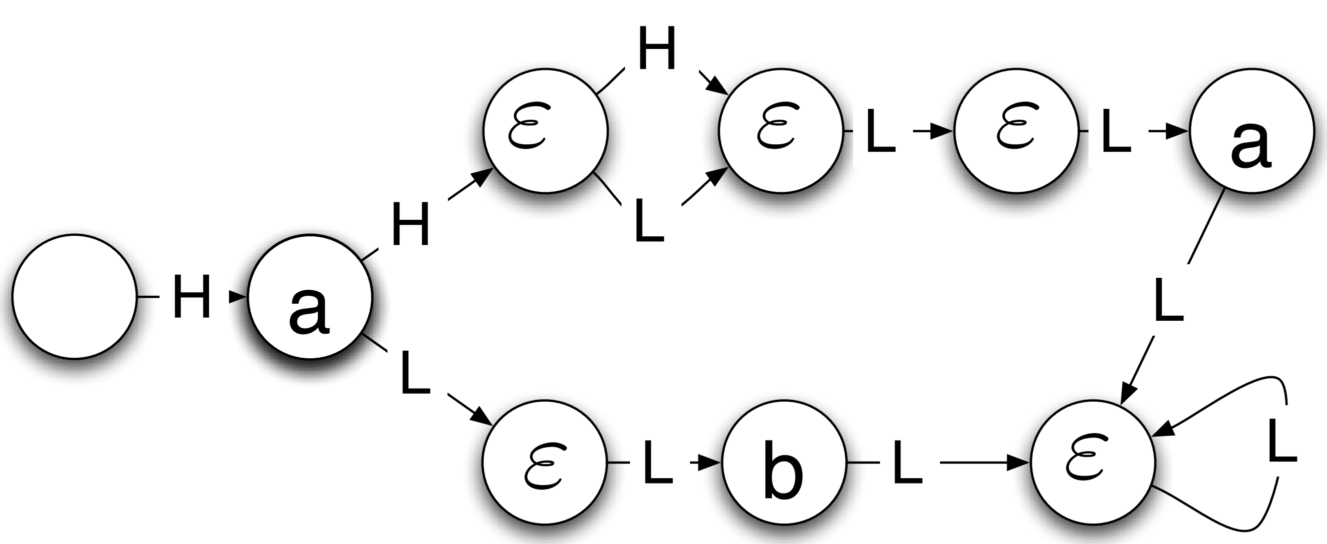
L : x <=85



Discrete time control of RHA

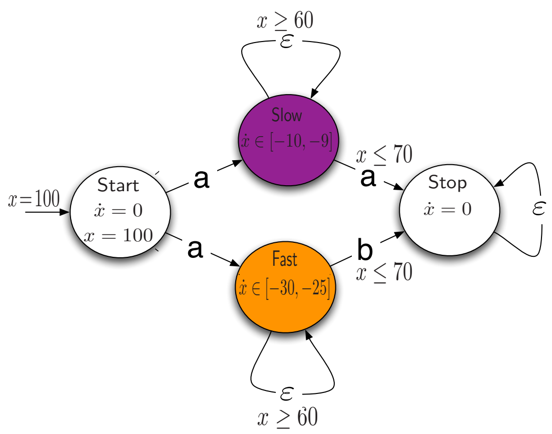


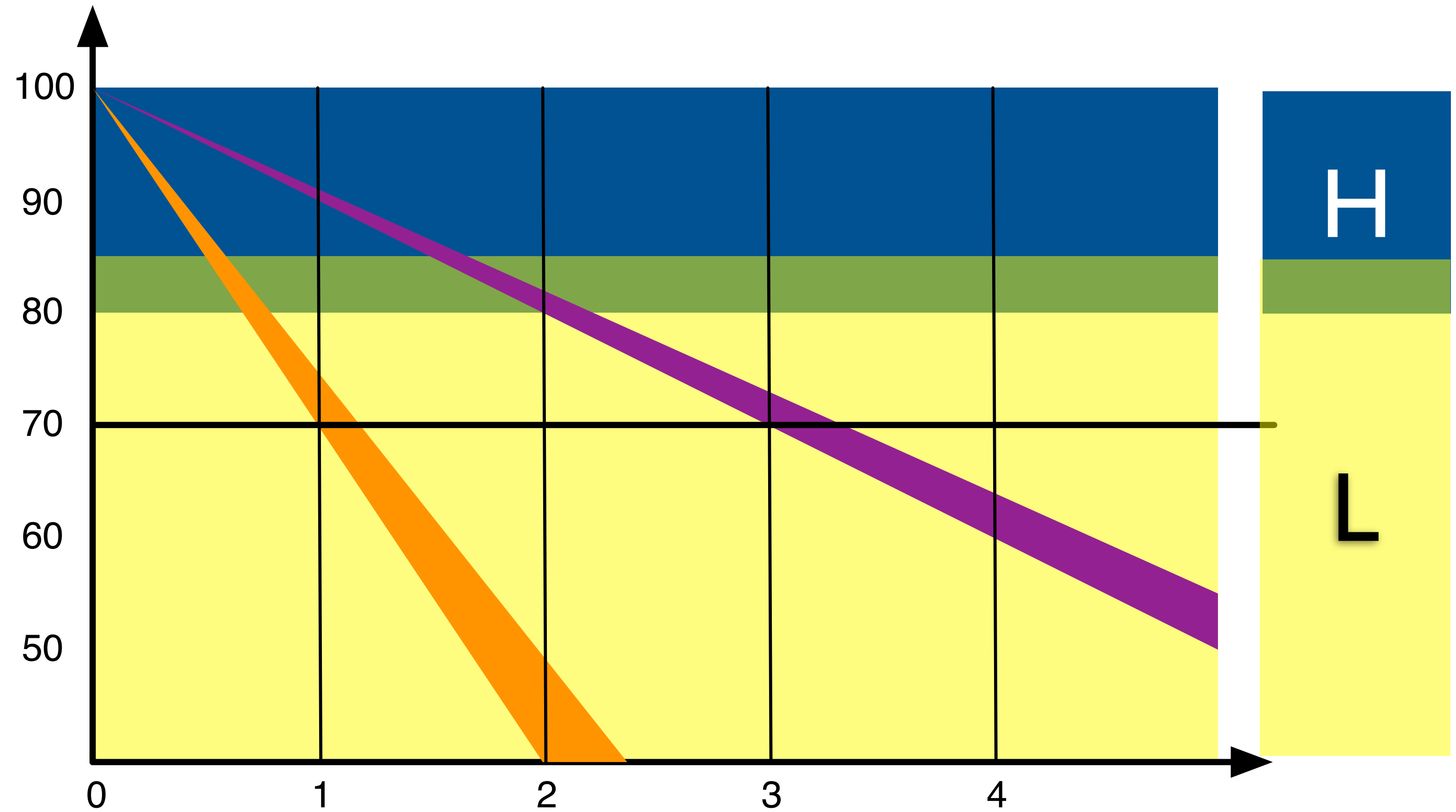


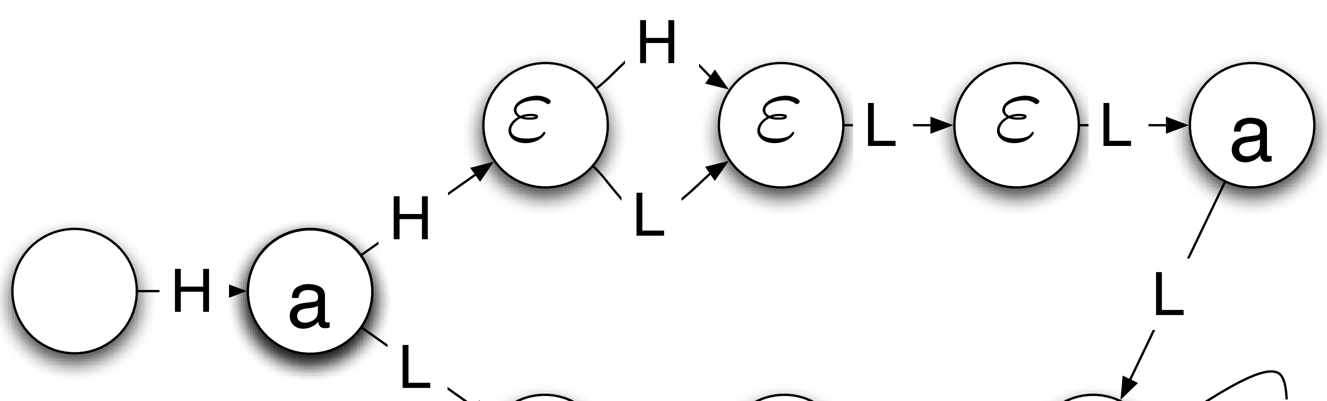


The Strategy

Discrete time control of RHA



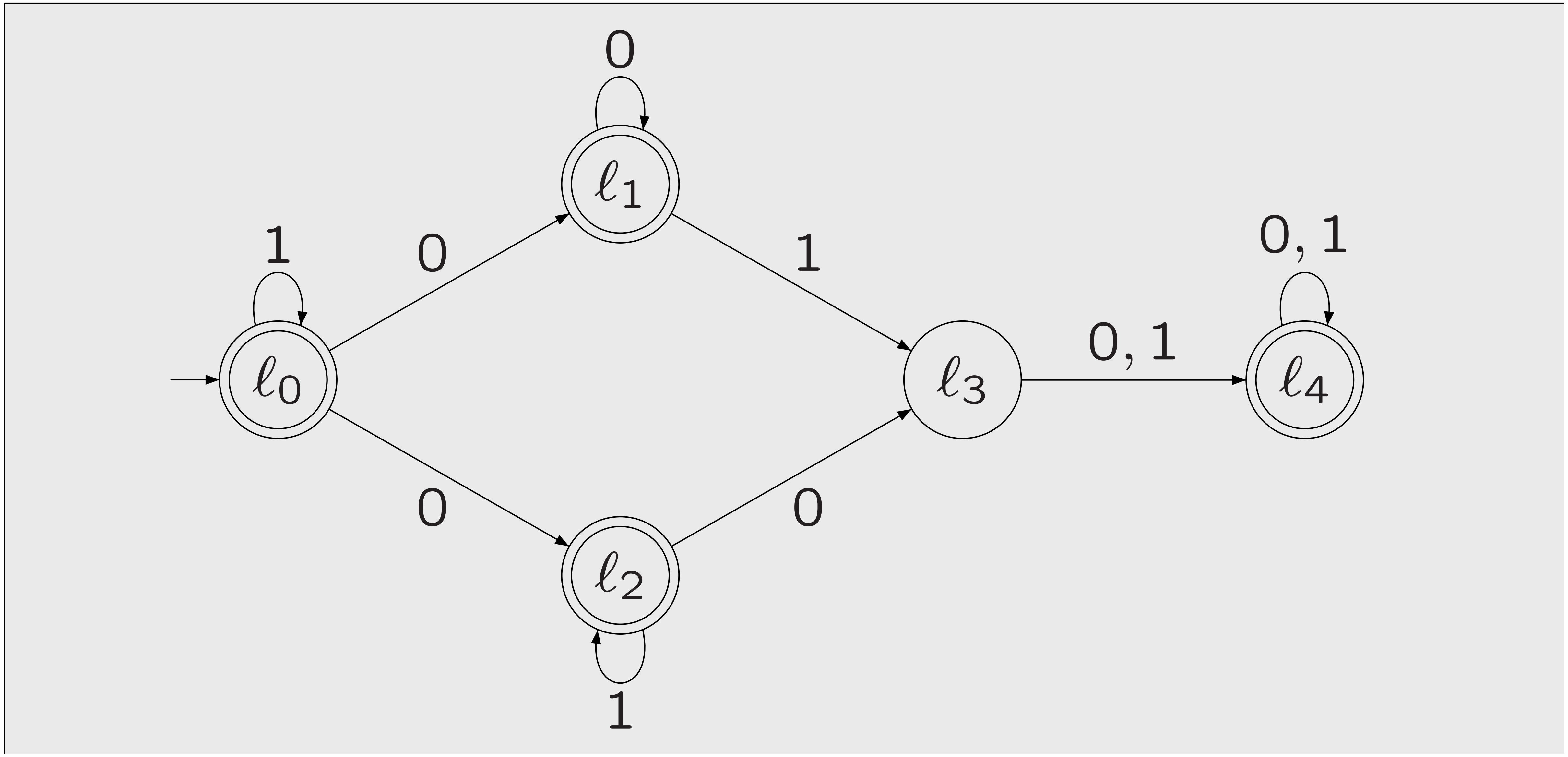




The symbolic CPre can be encoded in the script language of HyTech

Another application:
avoiding determinization
when testing
universability of NFA

Universality of NFA



Universality of NFA

Consider a game played by a protagonist and an antagonist

The protagonist wants to establish that $\mathcal{A}$ is not universal.

The protagonist has to provide a finite word w such that no matter how the antagonist reads it using $\mathcal{A}$, the automaton ends up in a rejecting location.

$\Longrightarrow$ This is a one-shot game.

Universality of NFA

Consider a game played by a protagonist and an antagonist

The protagonist wants to establish that $\mathcal{A}$ is not universal.

The protagonist has to provide a finite word w such that no matter how the antagonist reads it using $\mathcal{A}$, the automaton ends up in a rejecting location.

$\Longrightarrow$ This is a one-shot game.

The game is turn-based: the protagonist provides the word w one letter at a time, and the antagonist updates the state of $\mathcal{A}$. The protagonist cannot observe the state chosen by the antagonist.

$\Longrightarrow$ This is a blind game (or game of null information).

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Consider the following controllable predecessor operator over sets of sets of locations. For $q \subseteq 2^{\mathsf{Loc}}$, let:

$$\mathsf{CPre}(q) = \{s \mid \exists s' \in q \cdot \exists \sigma \in \Sigma \cdot \forall \ell \in s \cdot \forall \ell' \in \mathsf{Loc} : \delta_A(\ell, \sigma, \ell') \rightarrow \ell' \in s'\}$$

So $s \in \mathsf{CPre}(q)$ if there is a set $s' \in q$ that is reached from any location in s, reading input letter σ, that is $\mathsf{Post}_\sigma(s) \subseteq s'$.

$\Longrightarrow$ CPre encodes the blindness of the game.

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Theorem:

$\{\ell_I\} \in \mu x.(\mathsf{CPre}(x) \cup \{T\})$

iff

Protagonist has a strategy to win G_T

iff

$\mathcal{A}$ is not universal

Claim: For $s_1 \subseteq s_2$, if $\mathsf{Post}_\sigma(s_2) \subseteq s'$ then $\mathsf{Post}_\sigma(s_1) \subseteq s'$

and if $s_2 \in \mathsf{CPre}(\cdot)$, then $s_1 \in \mathsf{CPre}(\cdot)$

Idea: Keep in $\mathsf{CPre}(x)$ only the maximal elements.

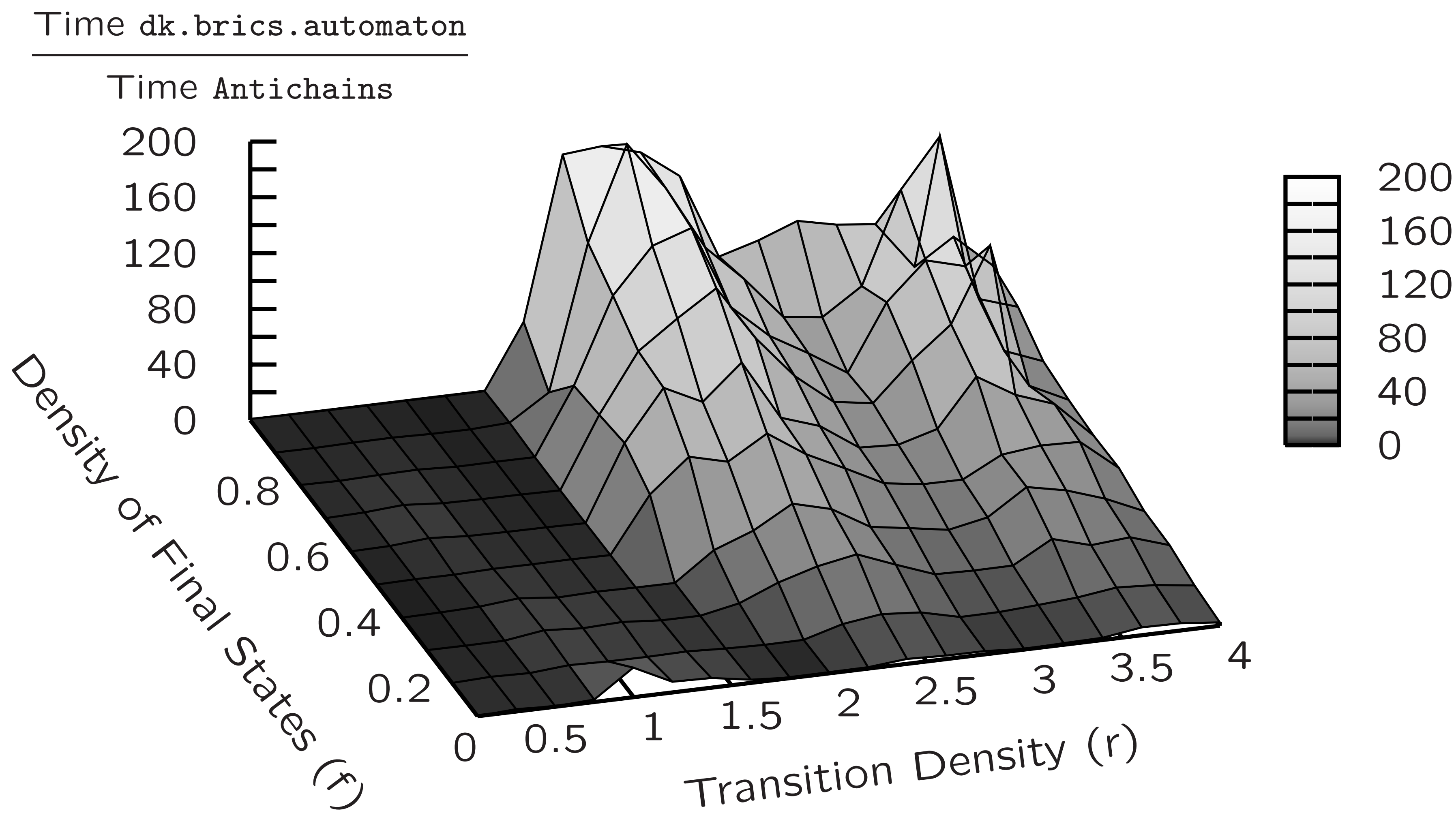
Universality - Experimental results (1)

- We compare our algorithm `Antichains` with the best[(1)] known algorithm `dk.brics.automaton` by Anders Møller.

  [(1)] According to "*D. Tabakov, M. Y. Vardi. Experimental Evaluation of Classical Automata Constructions. LPAR 2005*".

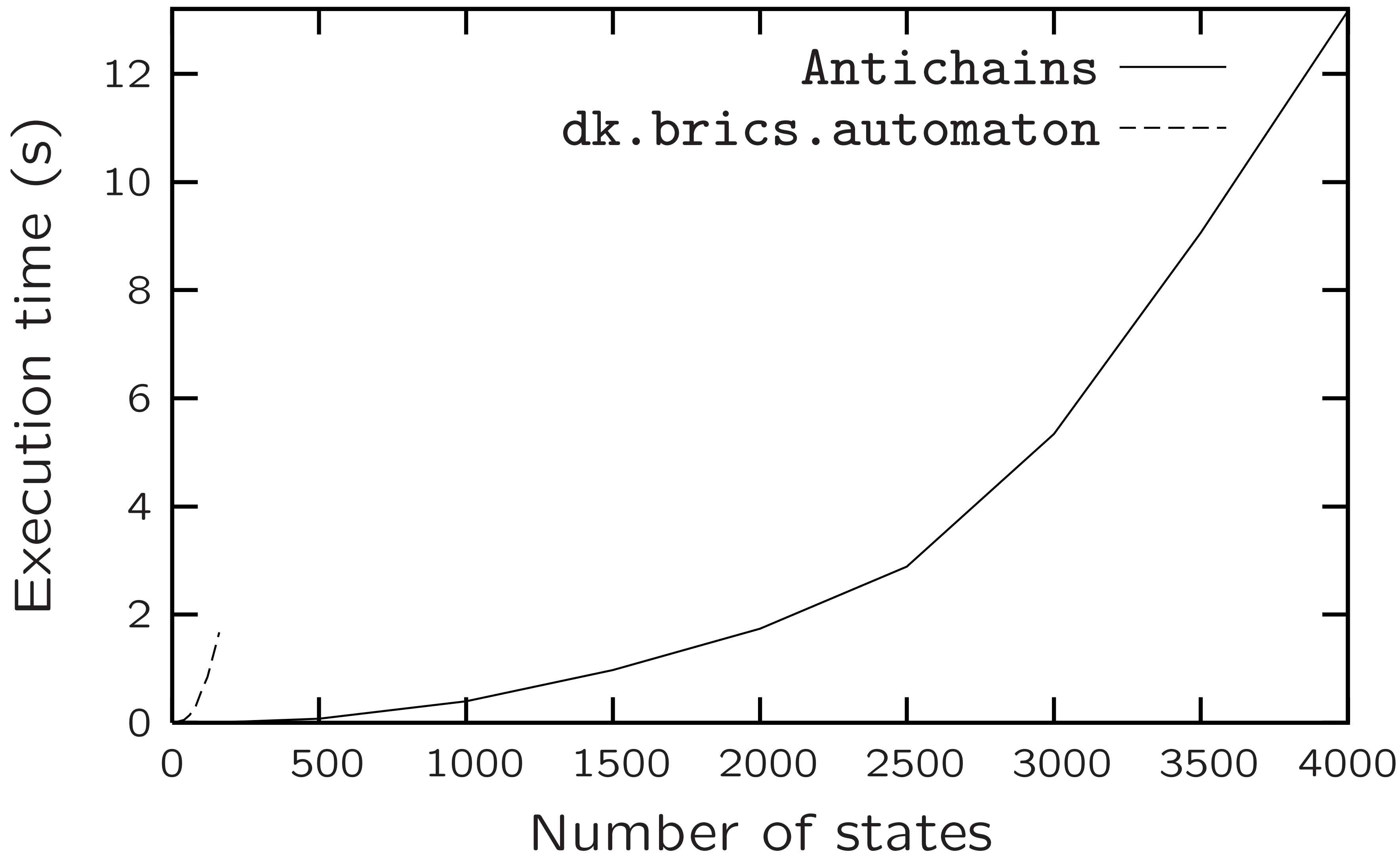
- We use a randomized model to generate the instances (automata of 175 locations). Two parameters:
  - Transition density: $r \geq 0$
  - Density of accepting states: $0 \leq f \leq 1$

Universality - Experimental results (2)



Each sample point: 100 automata with $|\mathsf{Loc}| = 175$, $\Sigma = \{0, 1\}$.

Universality - Experimental results (3)



- Transition density: $r = 2$.
- Density of accepting states: $f = 1$.

Works also for

- *language inclusion* between NFA
- *emptiness* of AFA

- See proceedings of CAV 2006

(joint work with Martin De Wulf, Laurent Doyen and Tom Henzinger)

Conclusion/Perspectives

- Winning strategies are controllers. We review a **lattice theory** to solve games.
- We have extended this theory for games of imperfect information, those games are needed to make the synthesis of **robust controllers** (= finite precision).
- Our technique computes only the information that is needed to find a winning strategy, i.e. we **avoid** the explicit subset construction.
- Applicable to **discrete time control** of RHA and useful to solve efficiently **classical problems** for NFA and AFA.
- Perspectives : continuous time control, finite automata on infinite words, efficient implementation issues, etc.

Bibliography

- Khrishnendu Charterjee, Laurent Doyen, Thomas A. Henzinger and Jean-Francois Raskin. **Algorithms for Omega-regular games of Incomplete Information**. In CSL'06, Lecture Notes in Computer Science, 4207, 287-302, 2006.
- Martin De Wulf, Laurent Doyen, Thomas A. Henzinger and Jean-Francois Raskin. **Antichains: a New Algorithm to Solve Universality of FA**. In CAV'06, Lecture Notes in Computer Science, 4144, Springer-Verlag, pp. 17-30, 2006.
- Martin De Wulf, Laurent Doyen, and Jean-Francois Raskin. **A Lattice Theory for Solving Games of Imperfect Information**. In HSCC'06, Lecture Notes in Computer Science, 3927, pp. 153-168, Springer-Verlag, 2006.
- Laurent Doyen and Jean-François Raskin. **Improved Algorithms for the Automata-based Approach to Model-Checking**. Accepted for publication in TACAS'07, Lecture Notes in Computer Science, Springer Verlag, 2007.
- Luca de Alfaro, Marco Faella, Thomas A. Henzinger, Rupak Majumdar, and Marielle Stoelinga. **The element of surprise in timed games.** Proceedings of the 14th International Conference on Concurrency Theory (CONCUR), Lecture Notes in Computer Science 2761, Springer, 2003, pp. 144-158.
- Franck Cassez, Alexandre David, Emmanuel Fleury, Kim G. Larsen, and Didier Lime. **Efficient on-the-fly algorithms for the analysis of timed games**. In Martín Abadi and Luca de Alfaro, editors, Proceedings of the 16th International Conference on Concurrency Theory (CONCUR'05), volume 3653 of Lecture Notes in Computer Science, pages 66-80, San Francisco, CA, USA, August 2005. Springer.
- Rajeev Alur, Thomas A. Henzinger, and Orna Kupferman. **Alternating-time temporal logic**. Journal of the ACM 49:672-713, 2002.