Checking History-Determinism is NP-hard for Parity Automata

Aditya Prakash
University of Warwick, UK

0. Parity conditions

1. History-determinism and simulation

2. NP-hardness: a. deciding simulation
   b. deciding history-determinism

0. Parity Condition

Parity Condition

3, 1, 2, 1, 2, ...
4, 2, 3, 2, 3, ... → Sequence of natural numbers

Parity Condition

$3, 1, 2, 1, 2, \ldots$
$4, 2, 3, 2, 3, \ldots$

$\rightarrow$ Sequence of natural numbers

"Highest number occurring ∞ly often is even."

Parity Condition

$3, 1, 2, 1, 2, \ldots$
$4, 2, 3, 2, 3, \ldots$ $\Bigg]\rightarrow$ Sequence of natural numbers

"Highest number occurring ∞ly often is even."

Parity Condition

$3, 1, \underline{\underline{2}}, 1, 2, \ldots$ ✓

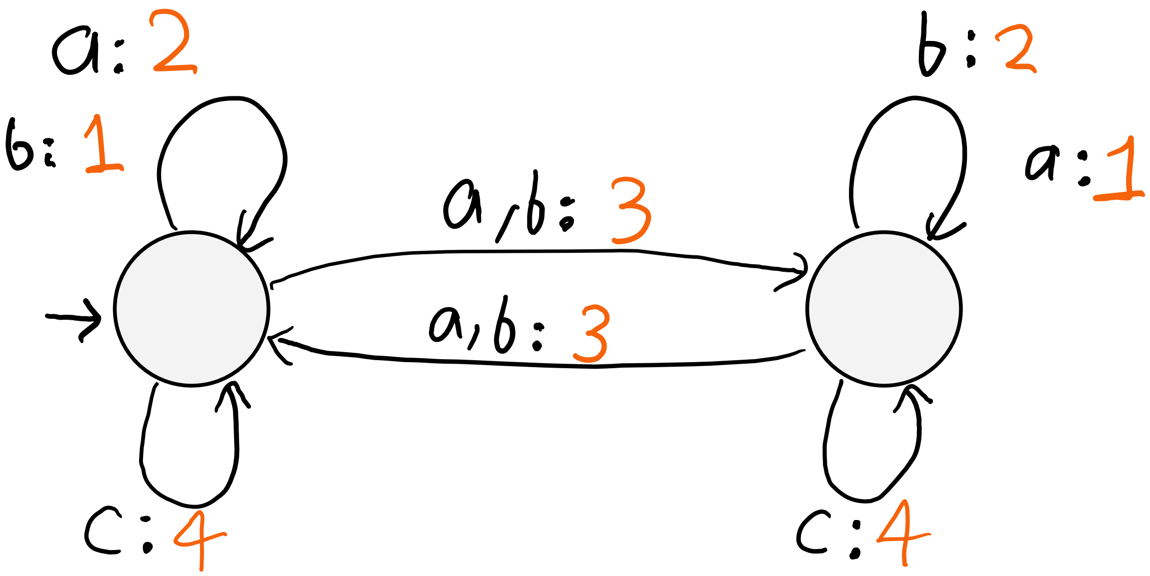
$4, 2, \underline{\underline{3}}, 2, 3, \ldots$ ✗

→ Sequence of natural numbers

"Highest number occurring ∞ly often is even."

Parity Automata

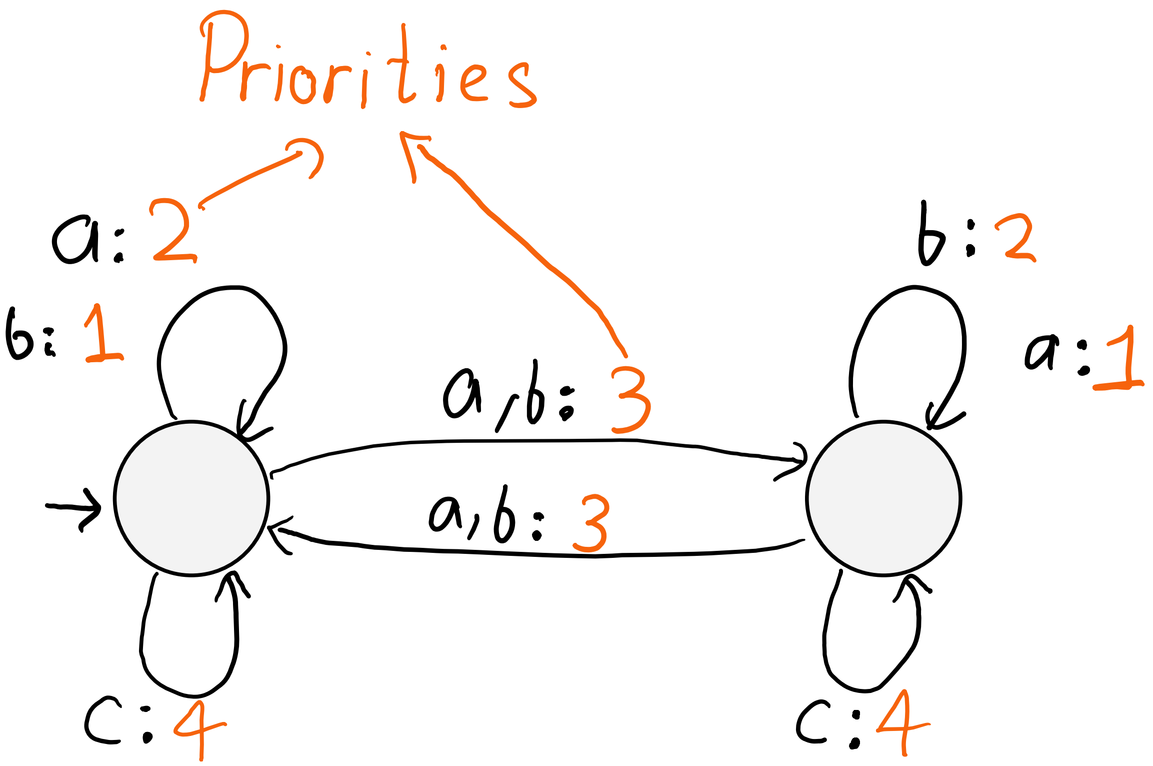
Input: $w \in \{a,b,c\}^{\mathbb{N}}$



Parity Automata

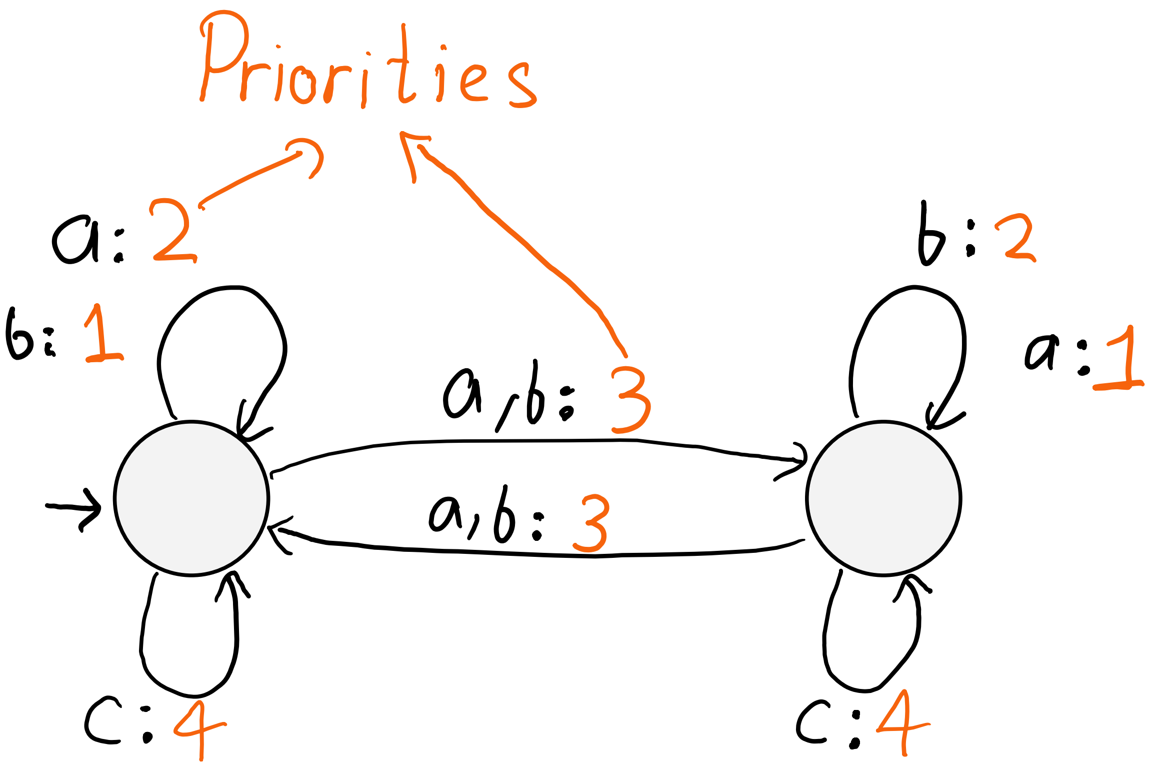
Input: $w \in \{a,b,c\}^{\mathbb{N}}$

Accepting run:
Sequence of priorities satisfies the parity condition.



Parity Automata

Input: $w \in \{a,b,c\}^{\mathbb{N}}$

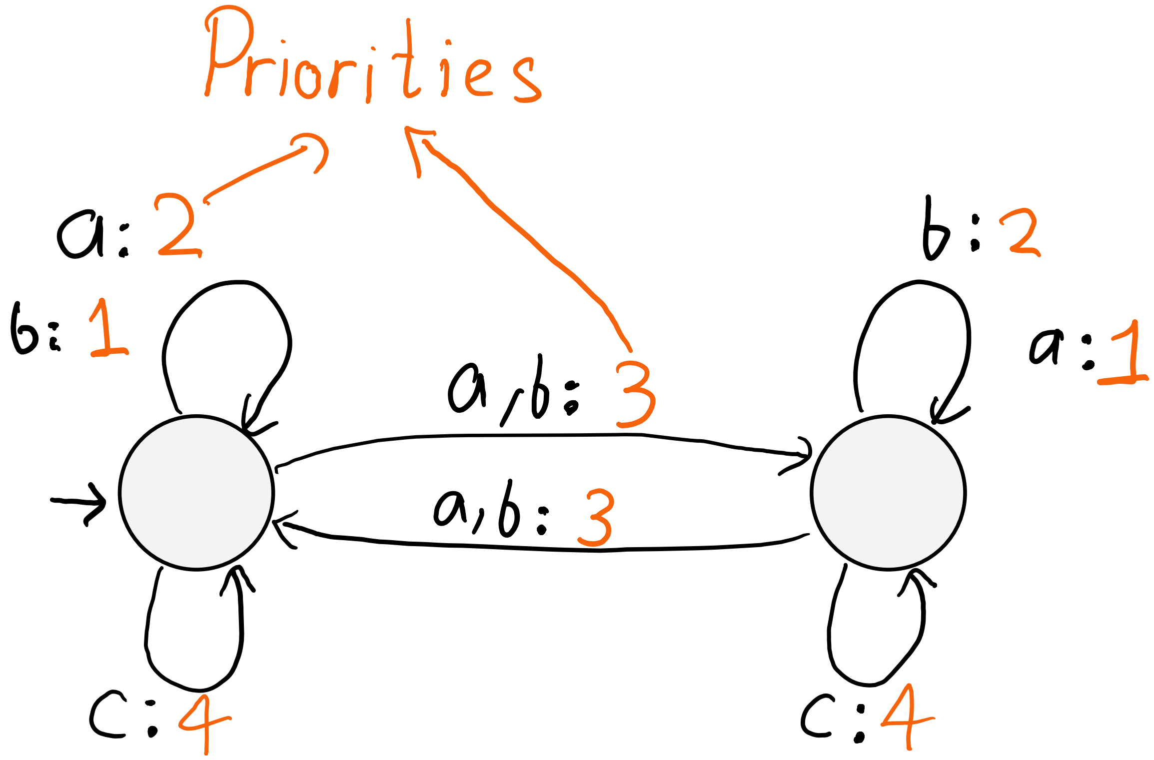


Accepting run:
Sequence of priorities satisfies the parity condition.

Accepting word: If the automaton has an accepting run on it.

Parity Automata

Input: $w \in \{a,b,c\}^{\mathbb{N}}$

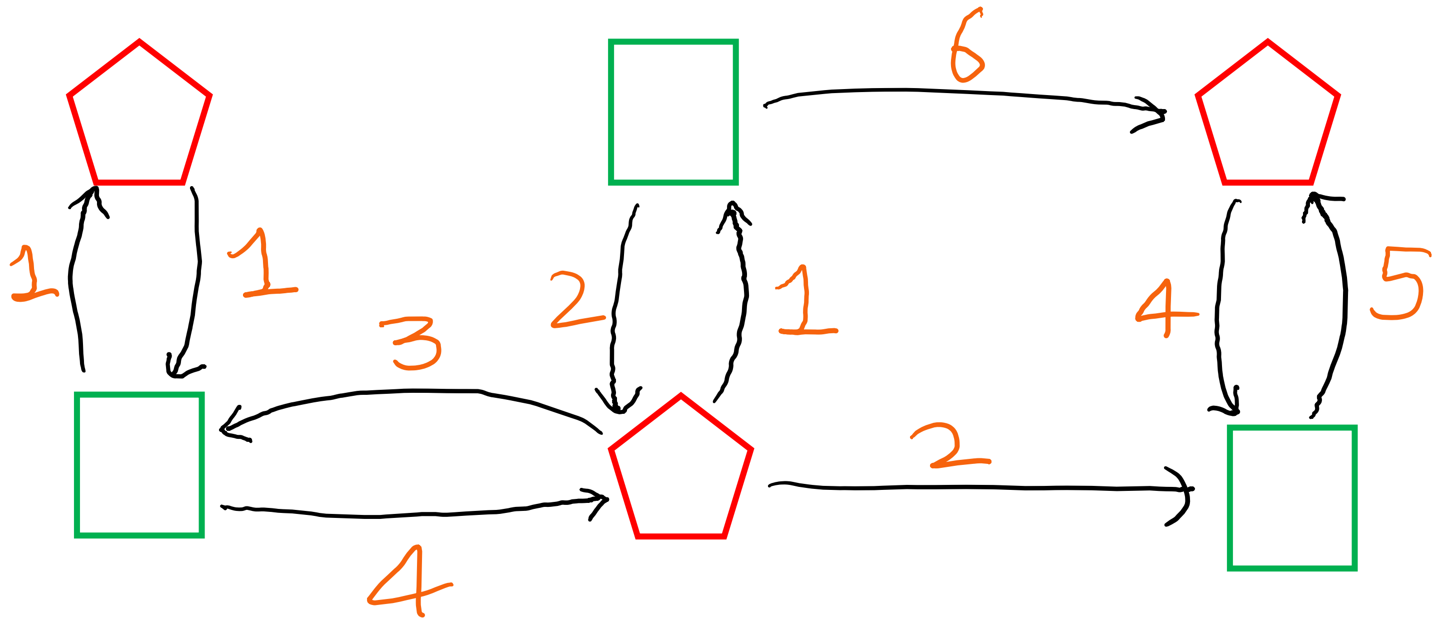


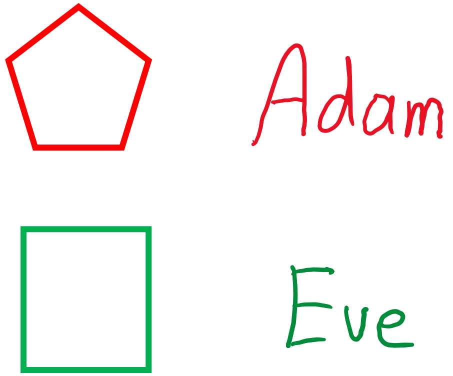
Accepting run:
Sequence of priorities satisfies the parity condition.

Accepting word: If the automaton has an accepting run on it.

Language: Set of accepting words.

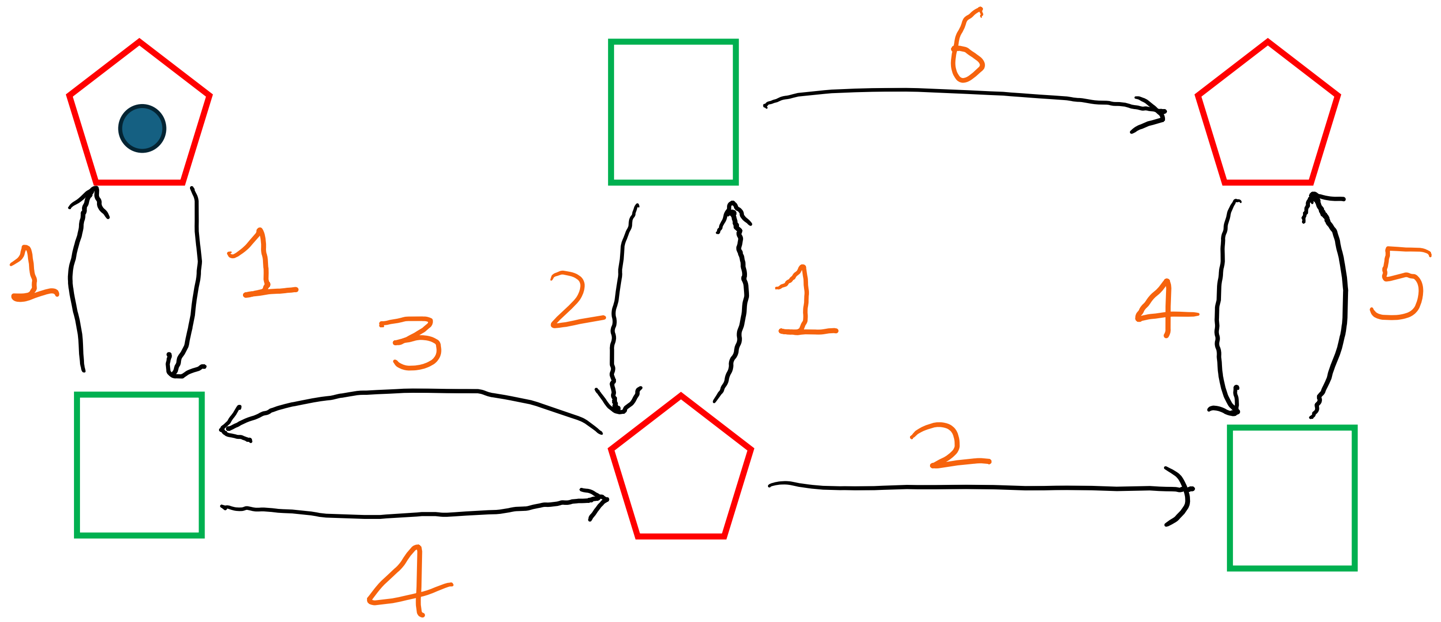
Parity Games

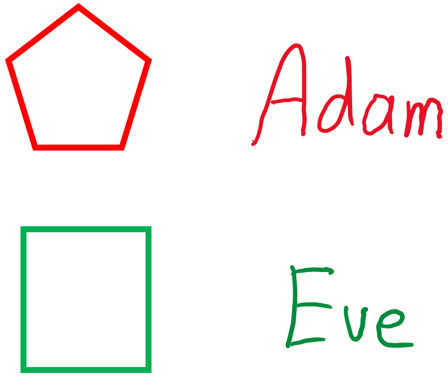




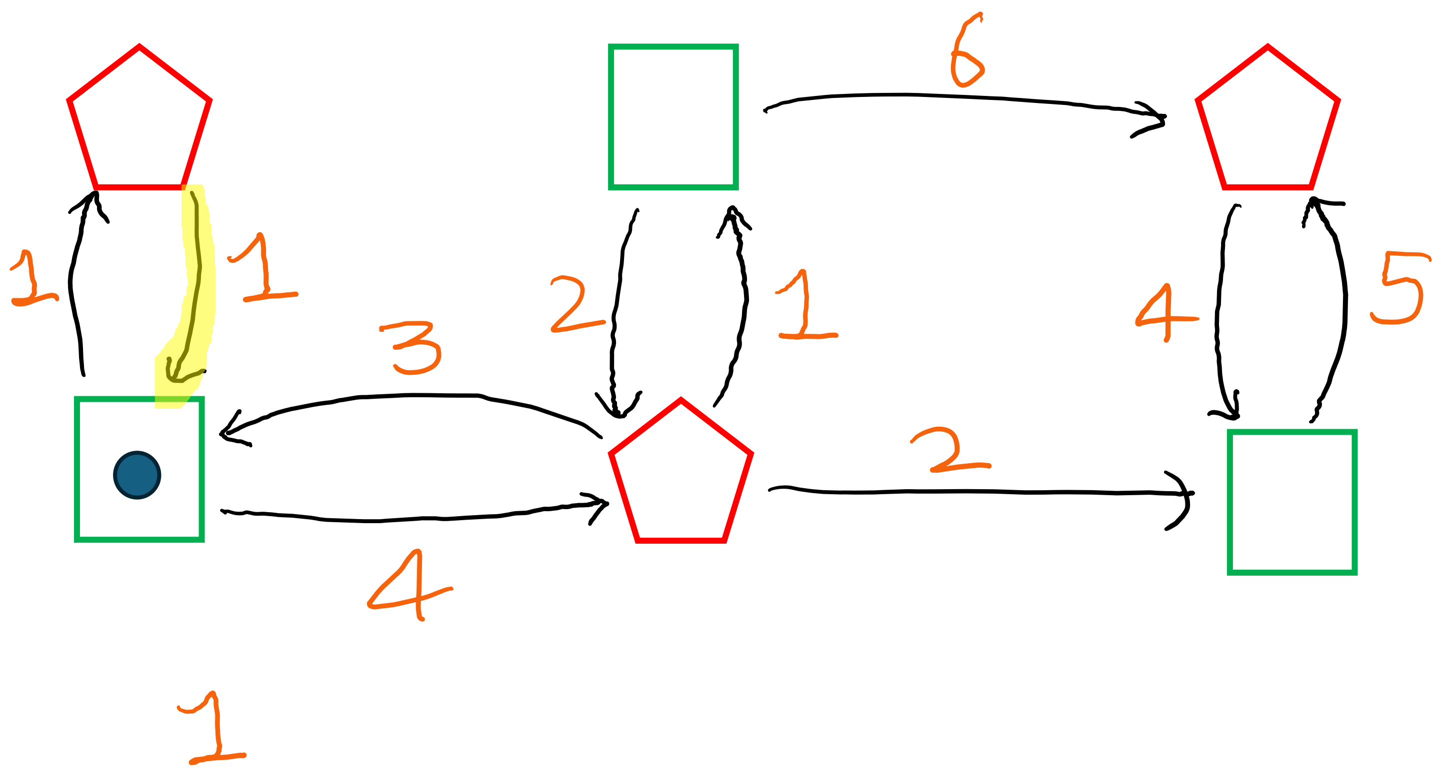
Assume bipartite

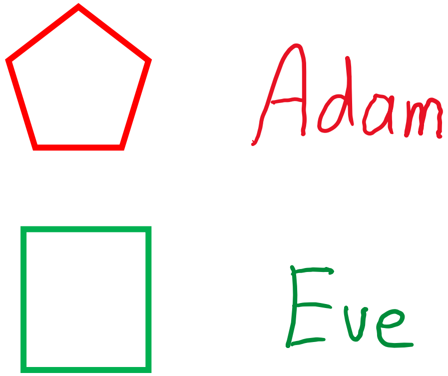
Parity Games



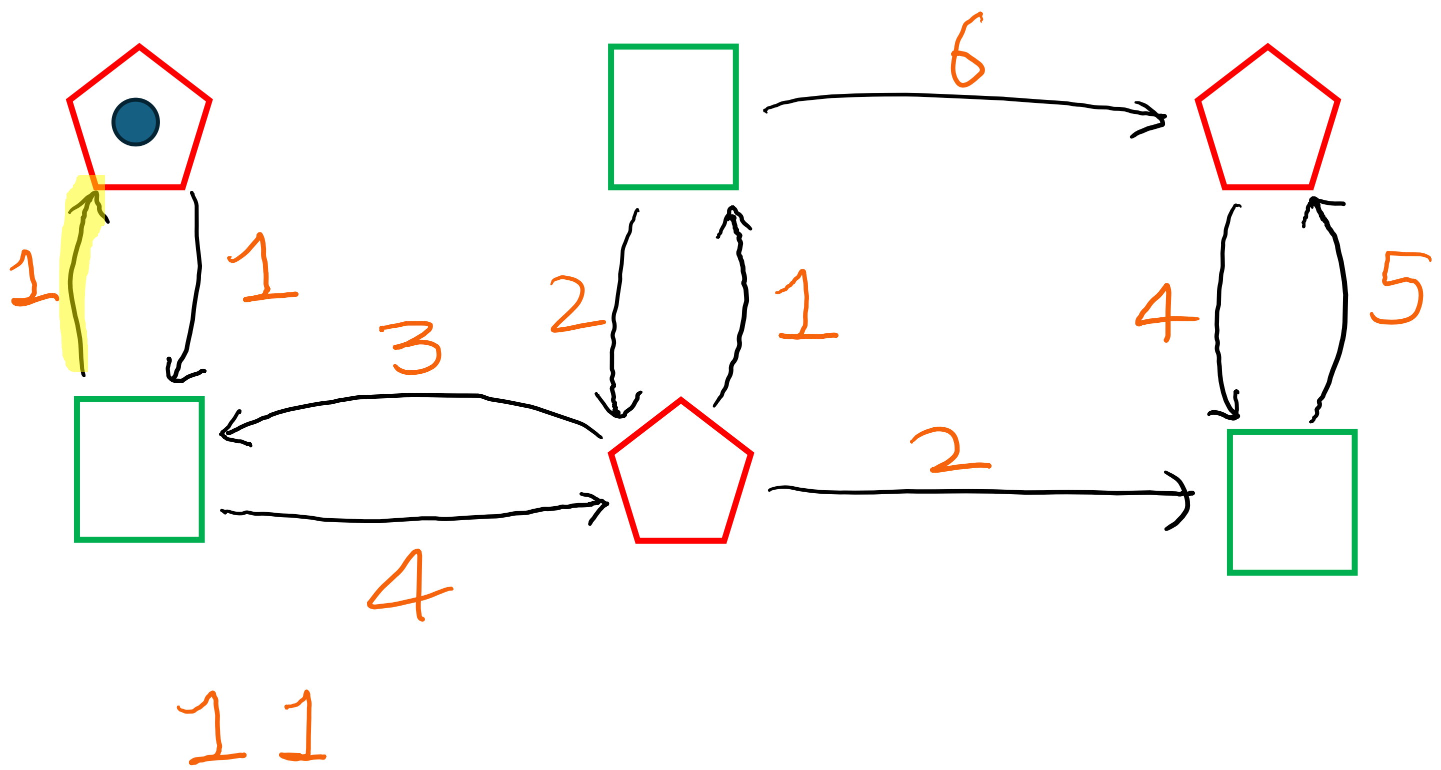


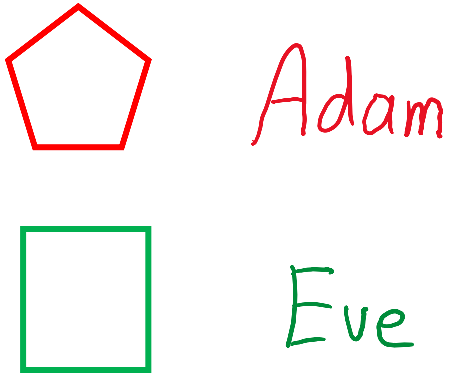
Parity Games



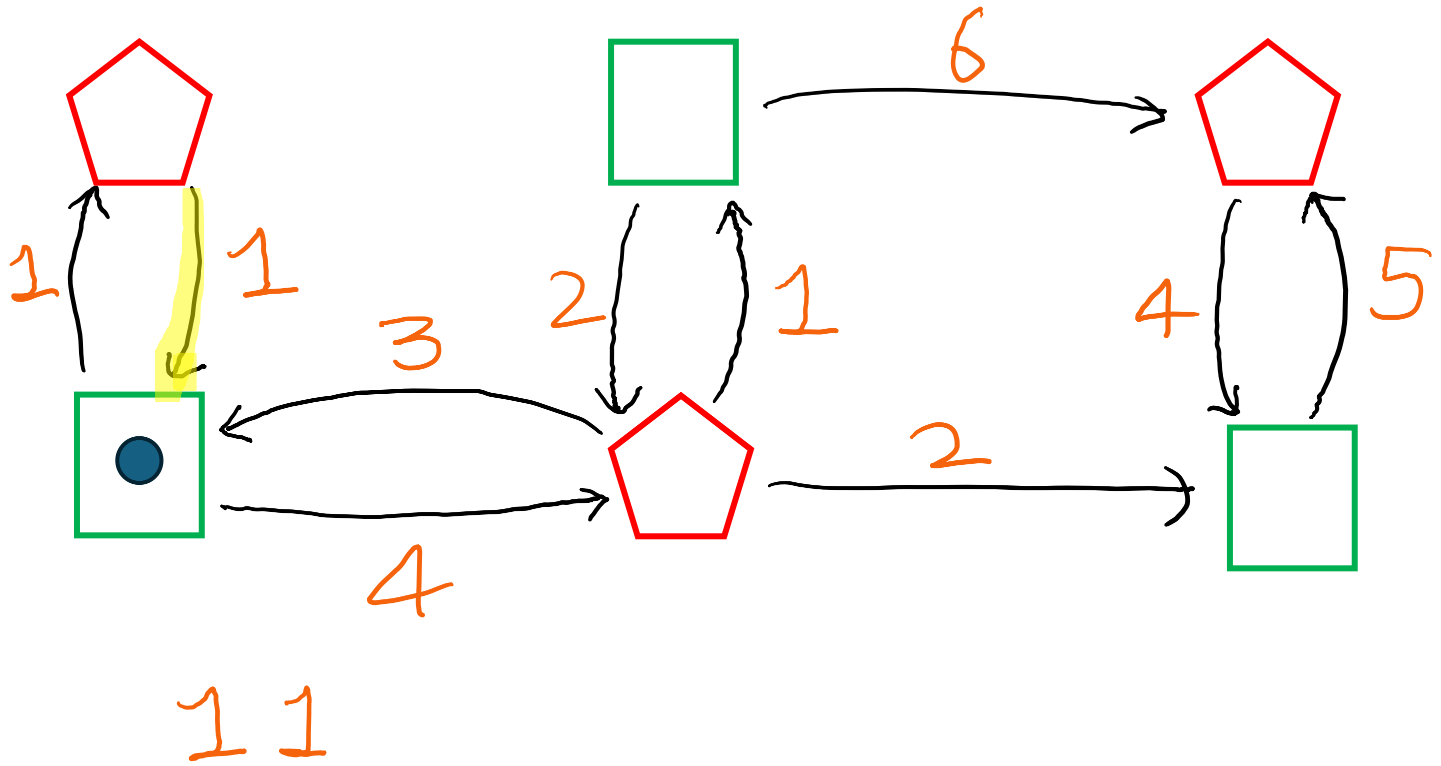


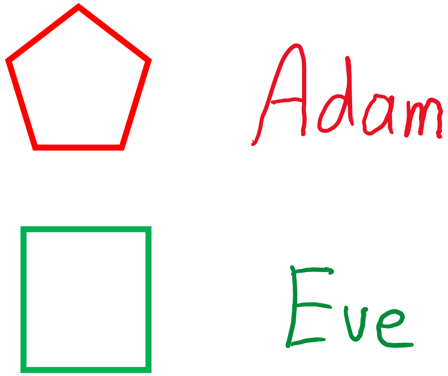
Parity Games



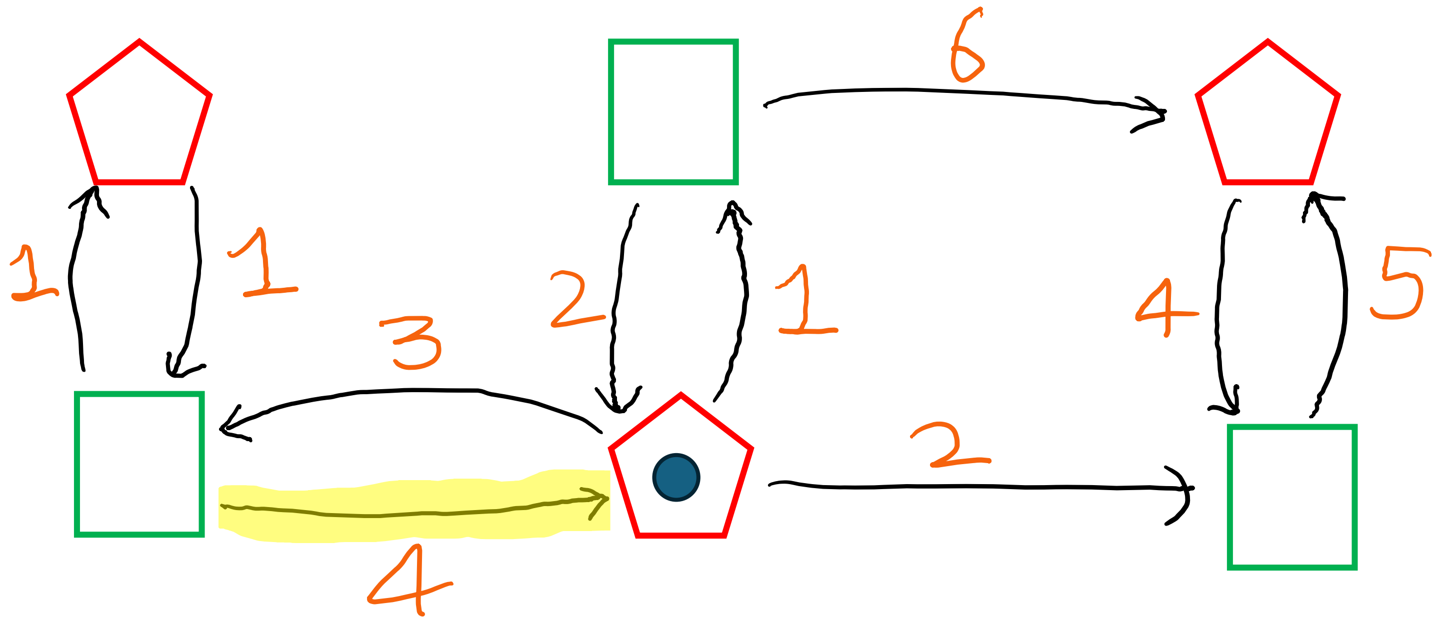


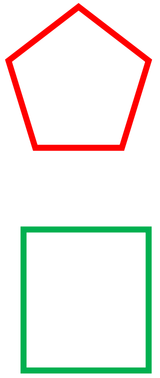
Parity Games





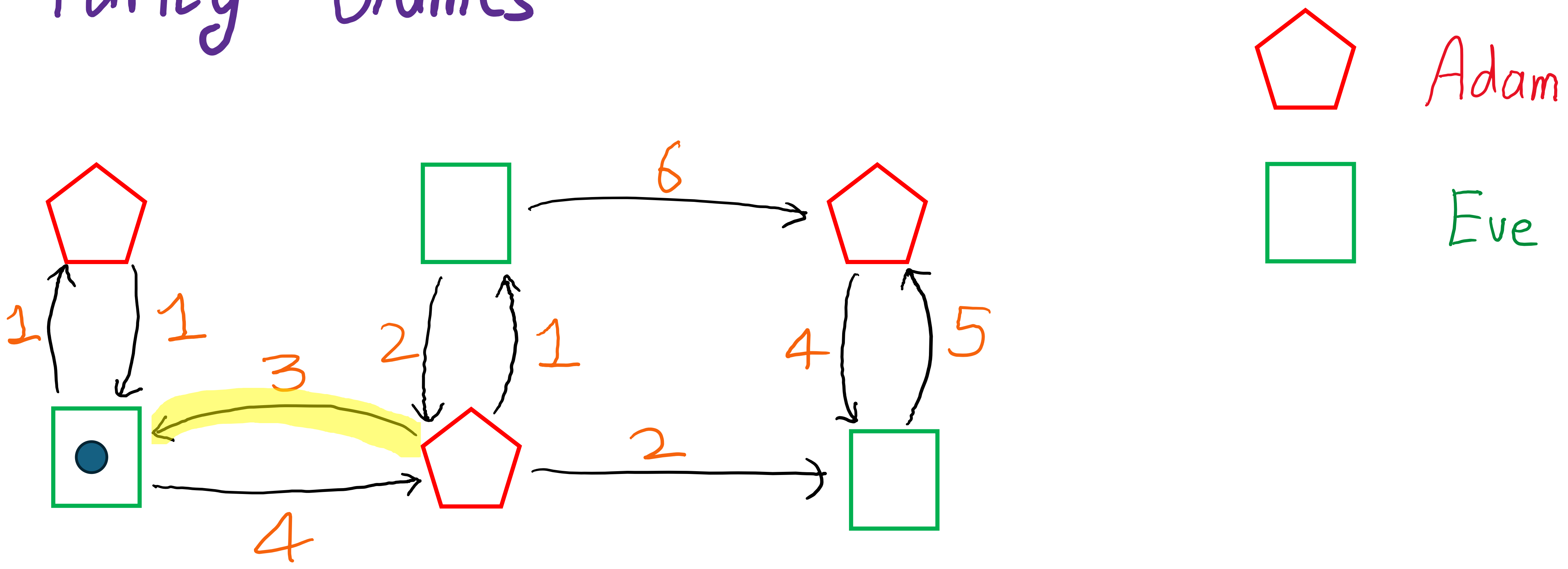
Parity Games





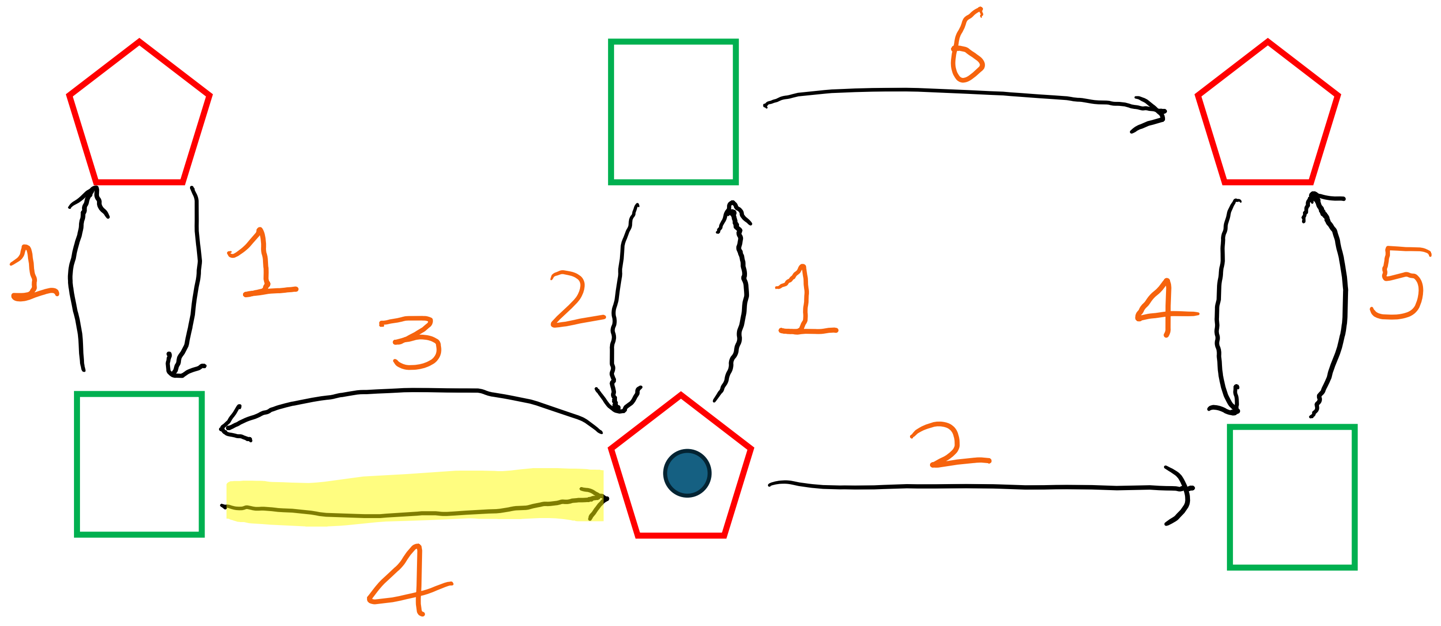
1 1 4

Parity Games

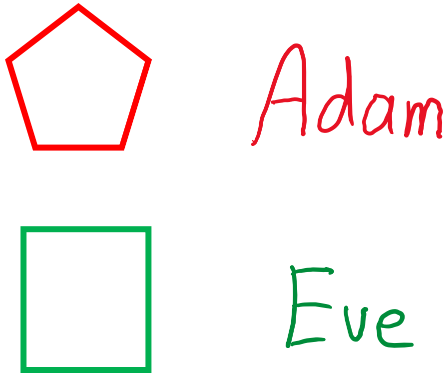


1 1 4 3

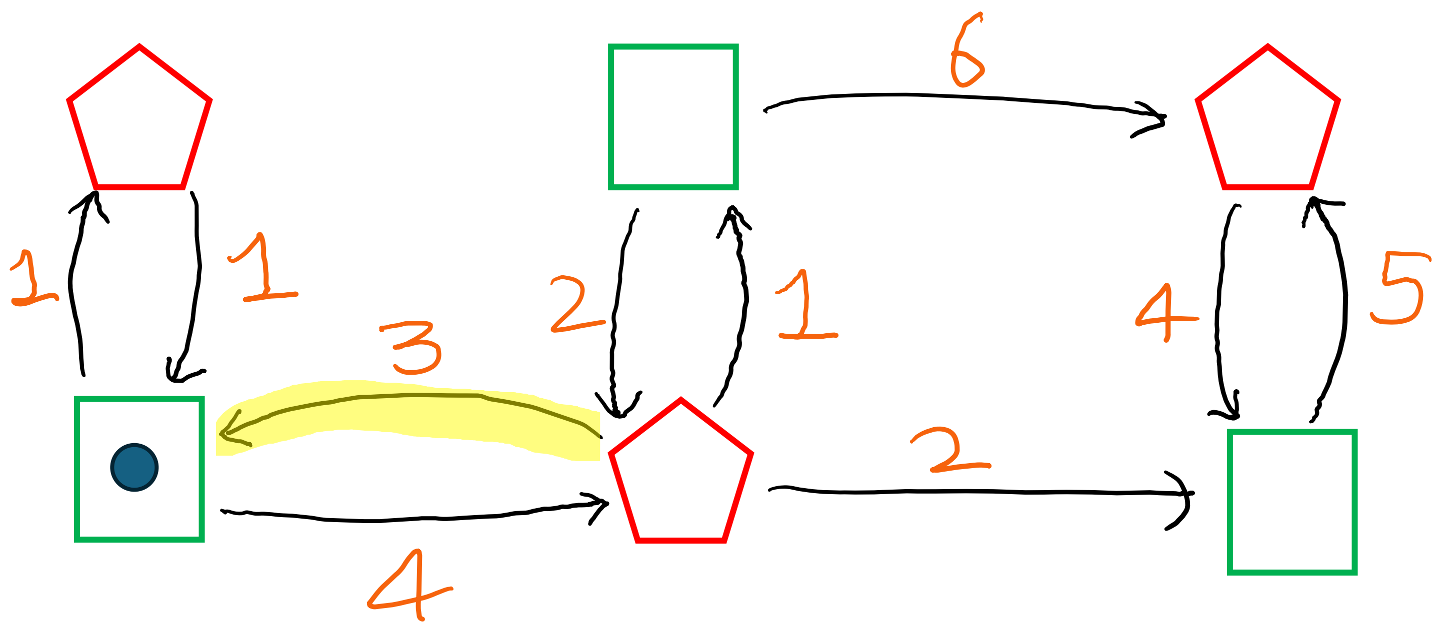
Parity Games

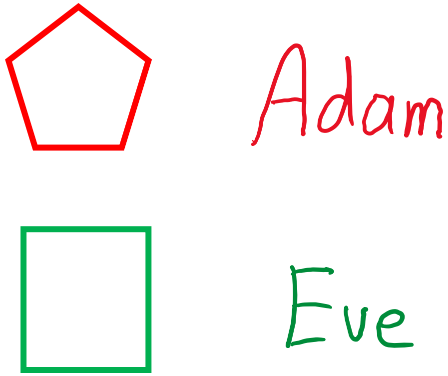


11434



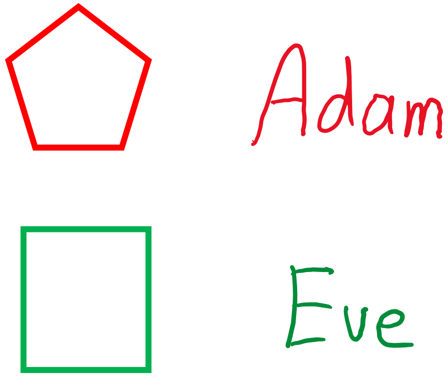
Parity Games

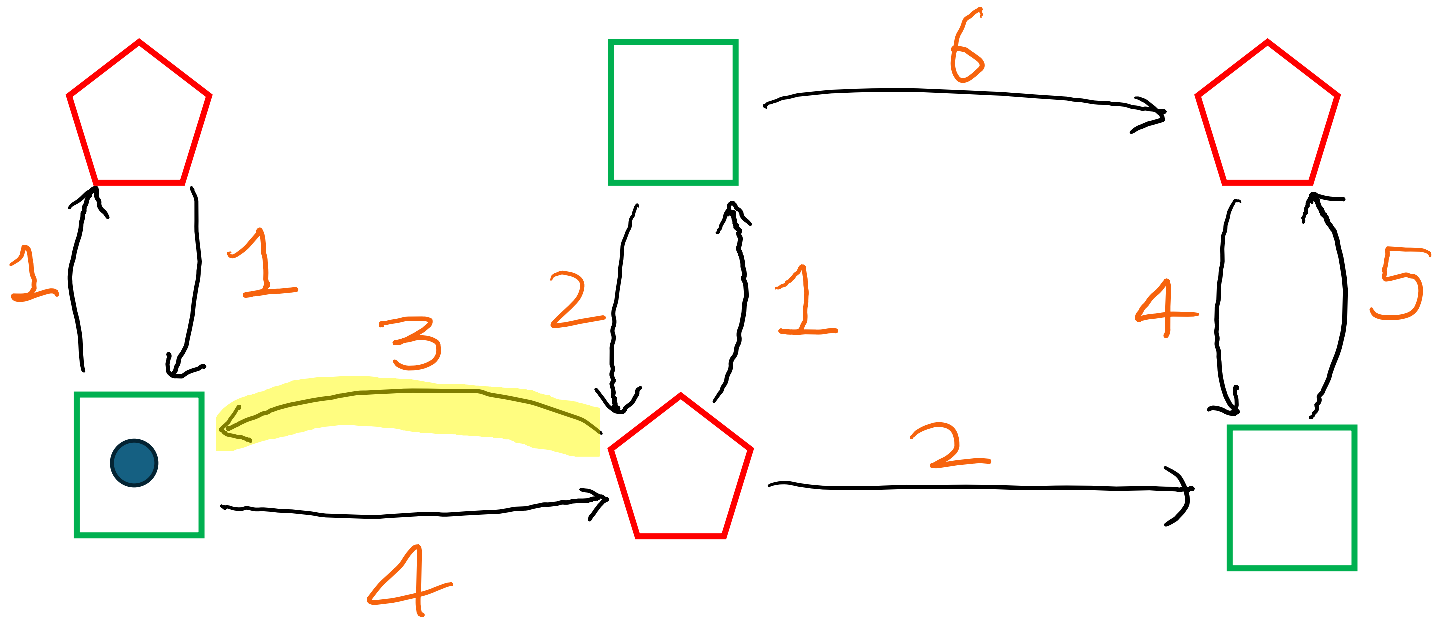




1 1 4 3 4 3 ...

Parity Games

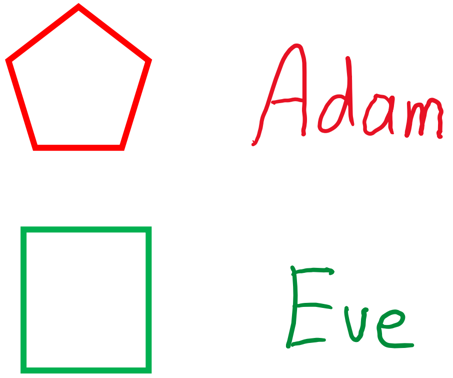


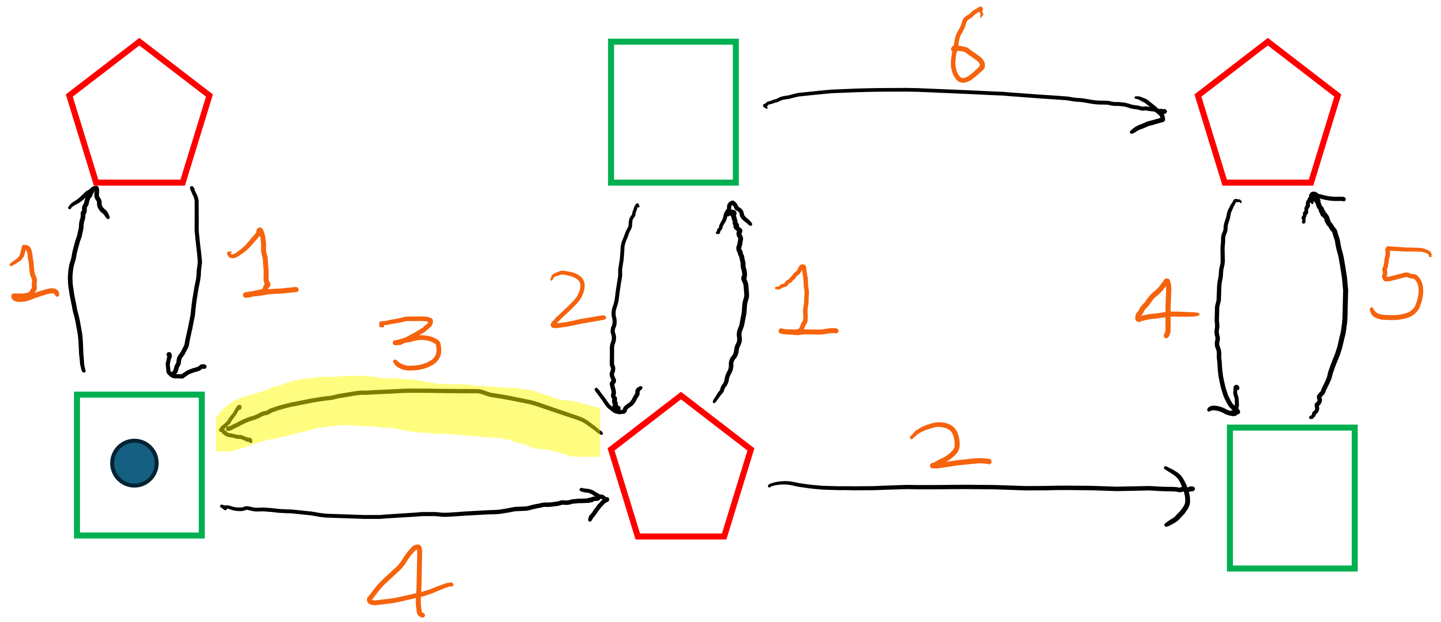


$$\underbrace{1\,1\,4\,3\,4\,3\ldots}_{\text{Satisfies the parity cond}^{n.}}$$

Eve wins the play.

Parity Games





$$\underbrace{1\,1\,4\,3\,4\,3\ldots}_{\text{Satisfies the parity cond}^{n.}}$$

Eve wins the play.

* Eve wins a game if she has a winning strategy.

Parity Games

Finding the winner in a parity game - $NP \cap coNP$, $UP \cap coUP$

n vertices and d priorities - $n^{\Theta(\log d)}$

Calude, Jain, Khoussainov,
Lei, Stephan 2017

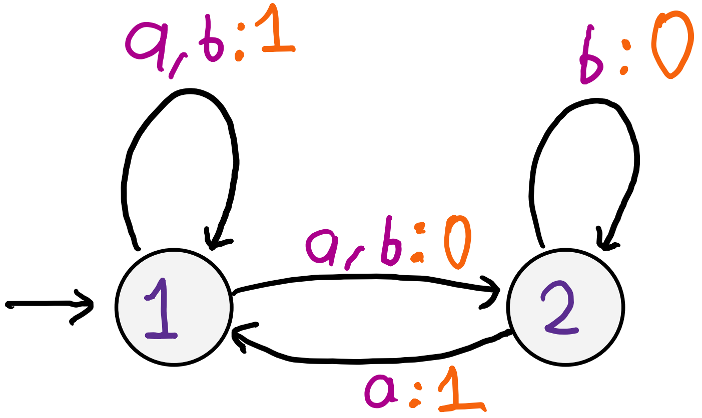
1. History - Determinism

History-Deterministic Automata

Nondeterministic automata in which the nondeterminism that arises while reading a word can be resolved based only on the prefix read so far.

Henzinger, Piterman 2006

History-Determinism Game



Starts at $\rightarrow$ 1

Adam selects letter a_i

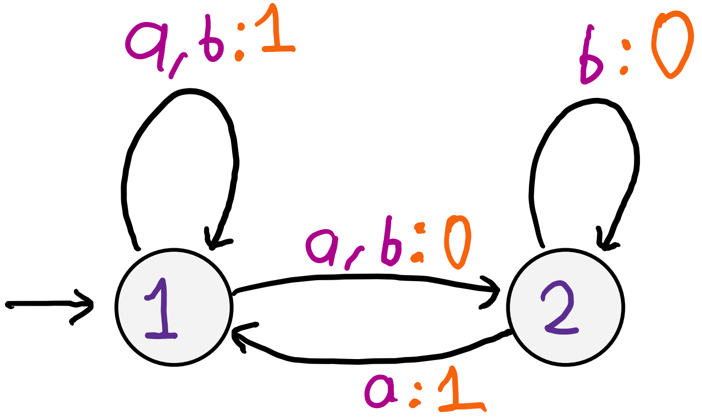
Eve selects transition $q_i \xrightarrow{a_i} q_{i+1}$

H.D. Game

Adam

Eve 1

History-Determinism Game



Starts at $\rightarrow$ 1

Adam selects letter a_i

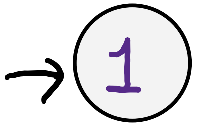
Eve selects transition $q_i \xrightarrow{a_i} q_{i+1}$

H.D. Game

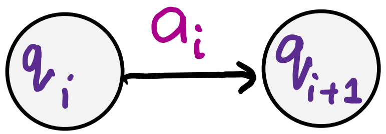
Adam a

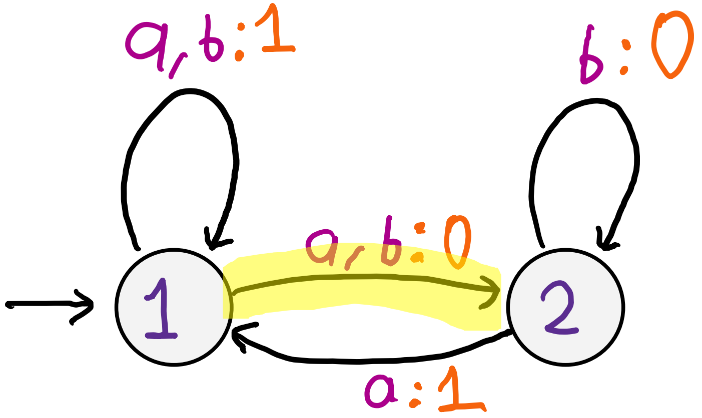
Eve 1

History-Determinism Game

Starts at 

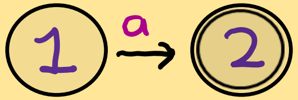
Adam selects letter a_i

Eve selects transition 

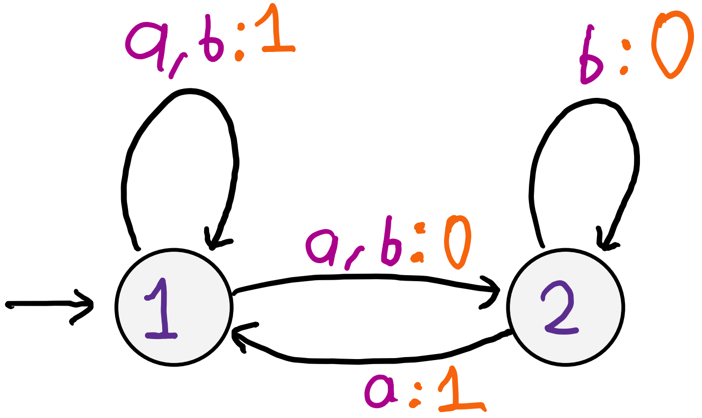


H.D. Game

Adam a

Eve 

History-Determinism Game



Starts at $\rightarrow (1)$

Adam selects letter a_i

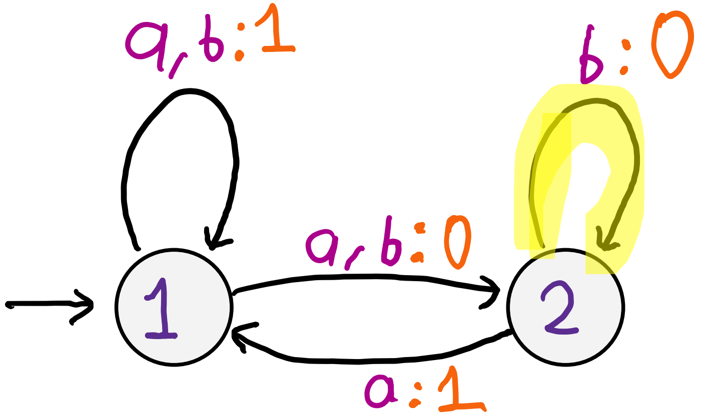
Eve selects transition $q_i \xrightarrow{a_i} q_{i+1}$

H.D. Game

Adam a b

Eve $1 \xrightarrow{a} 2$

History-Determinism Game



Starts at $\rightarrow (1)$

Adam selects letter a_i

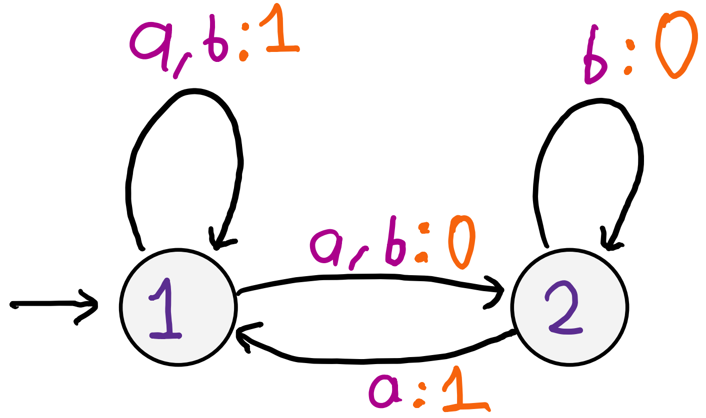
Eve selects transition $q_i \xrightarrow{a_i} q_{i+1}$

H.D. Game

Adam $\quad a \quad b$

Eve $\quad 1 \xrightarrow{a} 2 \xrightarrow{b} 2$

History-Determinism Game



Starts at $\rightarrow (1)$

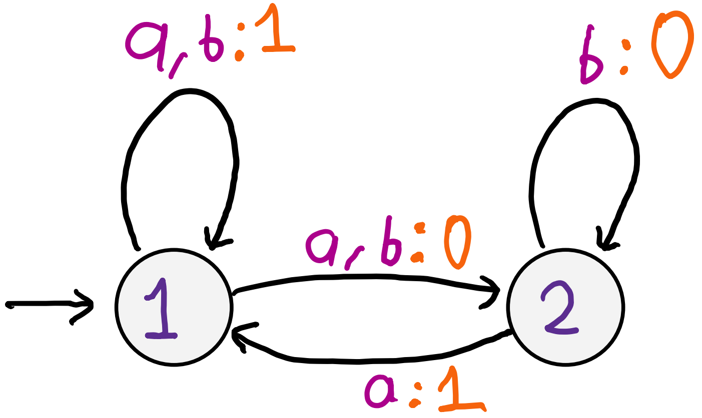
Adam selects letter a_i

Eve selects transition $q_i \xrightarrow{a_i} q_{i+1}$

H.D. Game

Adam	a	b	a	...
Eve	$1 \xrightarrow{a}$	$2 \xrightarrow{b}$	$2 \xrightarrow{a}$	$1 \rightarrow \dots$

History-Determinism Game



Starts at $\rightarrow$ 1

Adam selects letter a_i

Eve selects transition $q_i \xrightarrow{a_i} q_{i+1}$

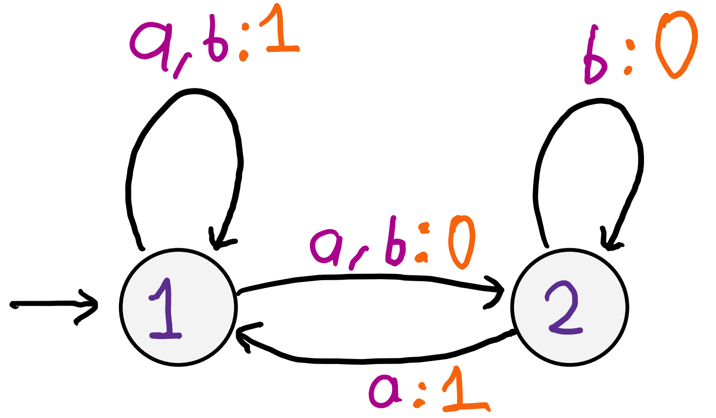
Winning cond$^{n.}$ for Eve:

Construct an accepting run if Adam's word is accepting.

H.D. Game

Adam	a	b	a	...	
Eve	1 $\xrightarrow{a}$	2 $\xrightarrow{b}$	2 $\xrightarrow{a}$	1 $\rightarrow$	...

History-Determinism Game



Starts at $\rightarrow (1)$

Adam selects letter a_i

Eve selects transition $(q_i) \xrightarrow{a_i} (q_{i+1})$

H.D. Game

Adam: a b a ...

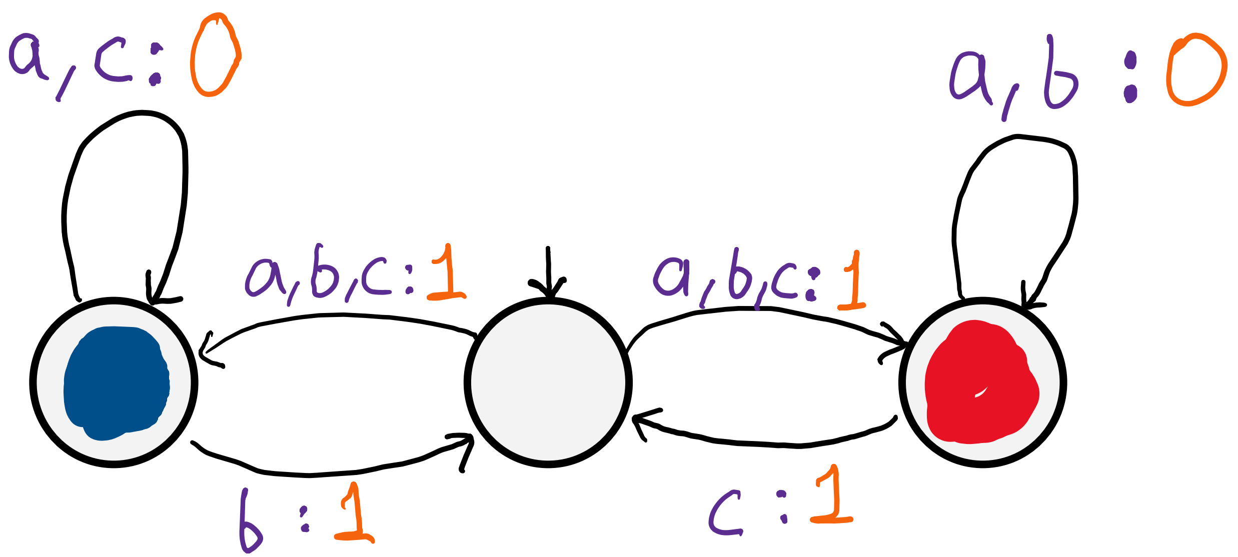
Eve: $(1) \xrightarrow{a} (2) \xrightarrow{b} (2) \xrightarrow{a} (1) \rightarrow \ldots$

Winning cond$^{n.}$ for Eve:

Construct an accepting run if Adam's word is accepting.

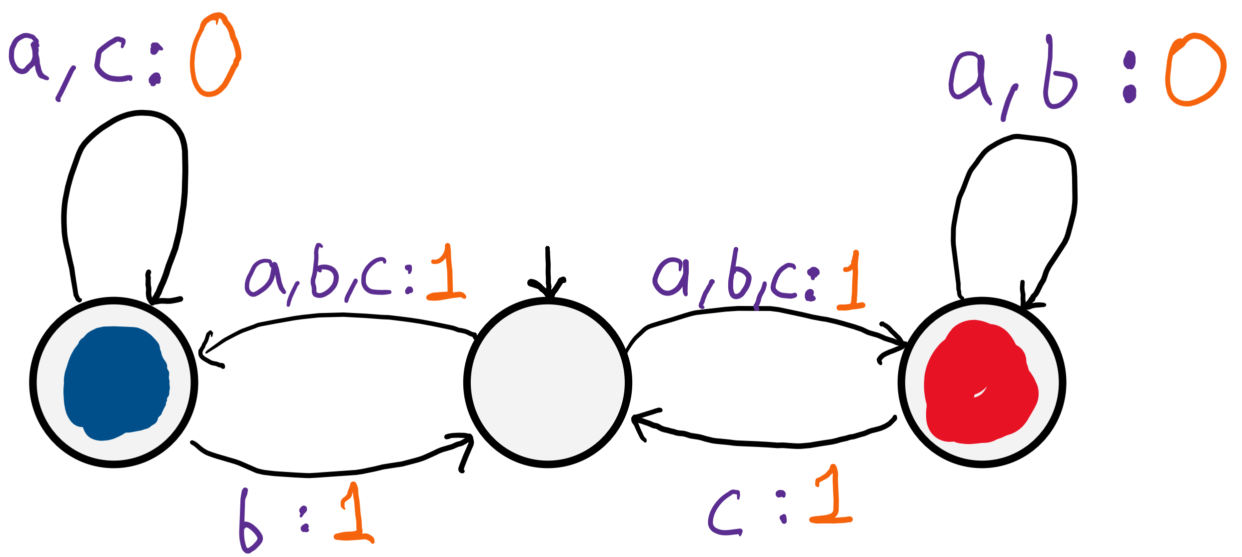
HD Automata: Eve has a winning strategy

Example: coBüchi automata



Accepting Condition: Finitely many 1's

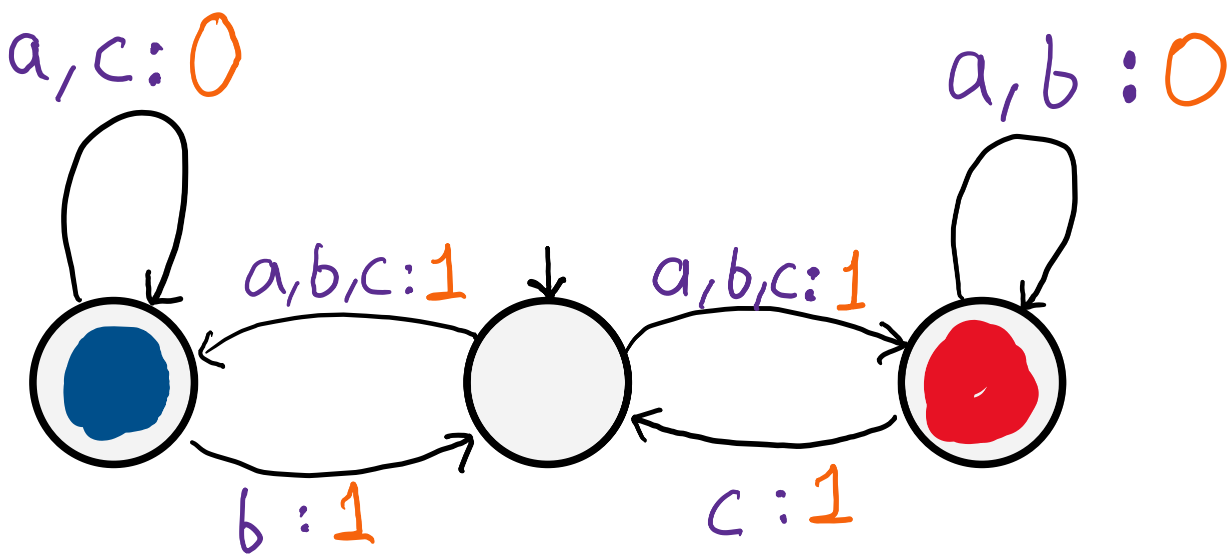
Example: coBüchi automata



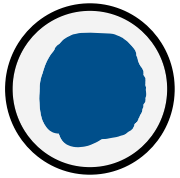
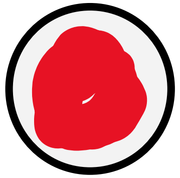
Accepting Condition: Finitely many 1's

$L = \{ w \mid w$ contains finitely many b's or finitely many c's $\}$

Example: coBüchi automata



HD game strategy:

Alternate between  and 

Accepting Condition: Finitely many 1's

$L = \{ w \mid w$ contains finitely many b's or finitely many c's $\}$

Why History-Determinism?

Language Inclusion

I $\qquad\qquad$ S

Implementation $\qquad$ Specification

$$L(I) \subseteq L(S)?$$

Why History-Determinism?

Language Inclusion

I Implementation

S Specification

$$L(I) \subseteq L(S)?$$

Parity Automata: PSPACE

Why History-Determinism?

Language Inclusion

I $S \xrightarrow{\text{(Folklore)}}$ If S is HD:

Implementation   Specification

$L(I) \subseteq L(S)$?

Equivalent to asking:
Does S simulate I?

Parity Automata: PSPACE $\rightsquigarrow$ NP

(Fair) Simulation Game

Automata I, S

Starts at $\to p_0$, $\to s_0$

In round i:

1. Adam selects a_i
2. Adam selects $p_i \xrightarrow{a_i} p_{i+1}$ in I
3. Eve selects $s_i \xrightarrow{a_i} s_{i+1}$ in S

Henzinger, Kupferman, Rajamani '97

(Fair) Simulation Game

$w = \quad a_0 \quad a_1 \quad a_2 \quad a_3 \ldots$

$\rho_A = \to p_0 \to p_1 \to p_2 \to p_3 \to \ldots$

$\rho_E = \to s_0 \to s_1 \to s_2 \to s_3 \to \ldots$

Automata I, S

Starts at $\to p_0$, $\to s_0$

In round i:

1. Adam selects a_i
2. Adam selects $p_i \xrightarrow{a_i} p_{i+1}$ in I
3. Eve selects $s_i \xrightarrow{a_i} s_{i+1}$ in S

Henzinger, Kupferman, Rajamani '97

(Fair) Simulation Game

Automata I , S

Starts at $\to p_0$, $\to s_0$

In round i :

1. Adam selects a_i
2. Adam selects $p_i \xrightarrow{a_i} p_{i+1}$ in I
3. Eve selects $s_i \xrightarrow{a_i} s_{i+1}$ in S

$$w = \quad a_0 \quad a_1 \quad a_2 \quad a_3 \ldots$$

$$\rho_A = \to p_0 \to p_1 \to p_2 \to p_3 \to \ldots$$

$$\rho_E = \to s_0 \to s_1 \to s_2 \to s_3 \to \ldots$$

Winning condition for Eve:

ρ_A is accepting $\Rightarrow$ ρ_E is accepting

Henzinger, Kupferman, Rajamani '97

(Fair) Simulation Game

Automata I, S

Starts at $\to p_0$, $\to s_0$

In round i:

1. Adam selects a_i
2. Adam selects $p_i \xrightarrow{a_i} p_{i+1}$ in I
3. Eve selects $s_i \xrightarrow{a_i} s_{i+1}$ in S

$$w = \quad a_0 \quad a_1 \quad a_2 \quad a_3 \dots$$

$$\rho_A = \to p_0 \to p_1 \to p_2 \to p_3 \to \dots$$

$$\rho_E = \to s_0 \to s_1 \to s_2 \to s_3 \to \dots$$

Winning condition for Eve:

ρ_A is accepting $\Rightarrow$ ρ_E is accepting

S simulates I if Eve has a winning strategy.

$I \hookrightarrow S$

Henzinger, Kupferman, Rajamani '97

Why History-Determinism?

* Model-checking: Inclusion reduces to simulation

Lemma : If S is HD, then for all I,
$$L(I) \subseteq L(S) \iff S \text{ simulates } I$$

Why History-Determinism?

* Model-checking: Inclusion reduces to simulation

Lemma : If S is HD, then for all I,

$$L(I) \subseteq L(S) \iff S \text{ simulates } I$$

Proof: $\Leftarrow$: Clear.

Why History-Determinism?

* Model-checking: Inclusion reduces to simulation

Lemma : If S is HD, then for all I,

$$L(I) \subseteq L(S) \Leftrightarrow S \text{ simulates } I$$

Proof: $\Leftarrow$: Clear.

$\Rightarrow$: Winning strategy for Eve in simulation game:

Why History-Determinism?

* Model-checking: Inclusion reduces to simulation

Lemma: If S is HD, then for all I,

$$L(I) \subseteq L(S) \Leftrightarrow S \text{ simulates } I$$

Proof: $\Leftarrow$: Clear.

$\Rightarrow$: Winning strategy for Eve in simulation game: select transitions using Eve's winning strategy in HD game on S.

Why History-Determinism?

* Model-checking: Inclusion reduces to simulation

Lemma : If S is HD, then for all I,
$L(I) \subseteq L(S) \iff S$ simulates I

Corollary: Deciding $L(I) \subseteq L(S)$ is in NP
if S is HD.

Why History-Determinism?

* Model-checking: Inclusion reduces to simulation

Lemma : If S is HD, then for all I,
$L(I) \subseteq L(S) \Leftrightarrow S$ simulates I

Theorem1: Deciding $L(I) \subseteq L(S)$ is in ~~NP~~ quasi-polynomial time if S is HD.

Why History-Determinism?

* Model-checking: Inclusion reduces to simulation
* Good-for-games

  [Henzinger, Piterman'06]

Why History-Determinism?

* Model-checking: Inclusion reduces to simulation
* Good-for-games

  [Henzinger, Piterman'06]
* Exponentially more succinct than deterministic parity automata

  [Kuperberg, Skrzypczak'15]

Why History-Determinism?

* Model-checking: Inclusion reduces to simulation
* Good-for-games [Henzinger, Piterman'06]
* Exponentially more succinct than deterministic parity automata [Kuperberg, Skrzypczak'15]

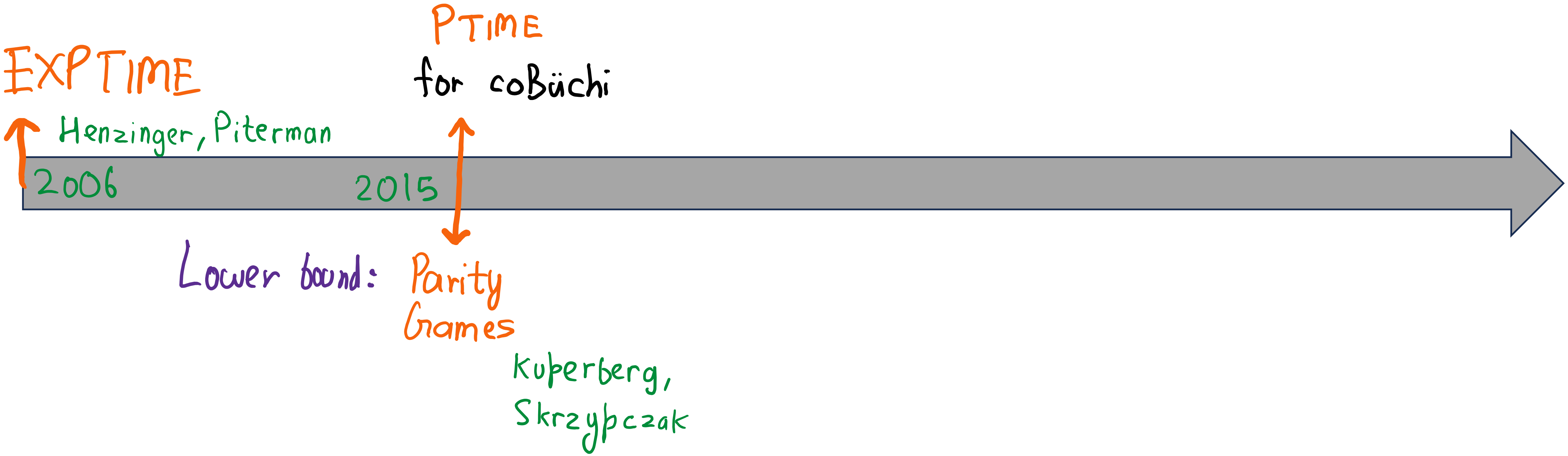
But : No known tractable algorithm to construct HD automata

Recognising HD Parity Automata

Given a parity automaton S, is S HD?

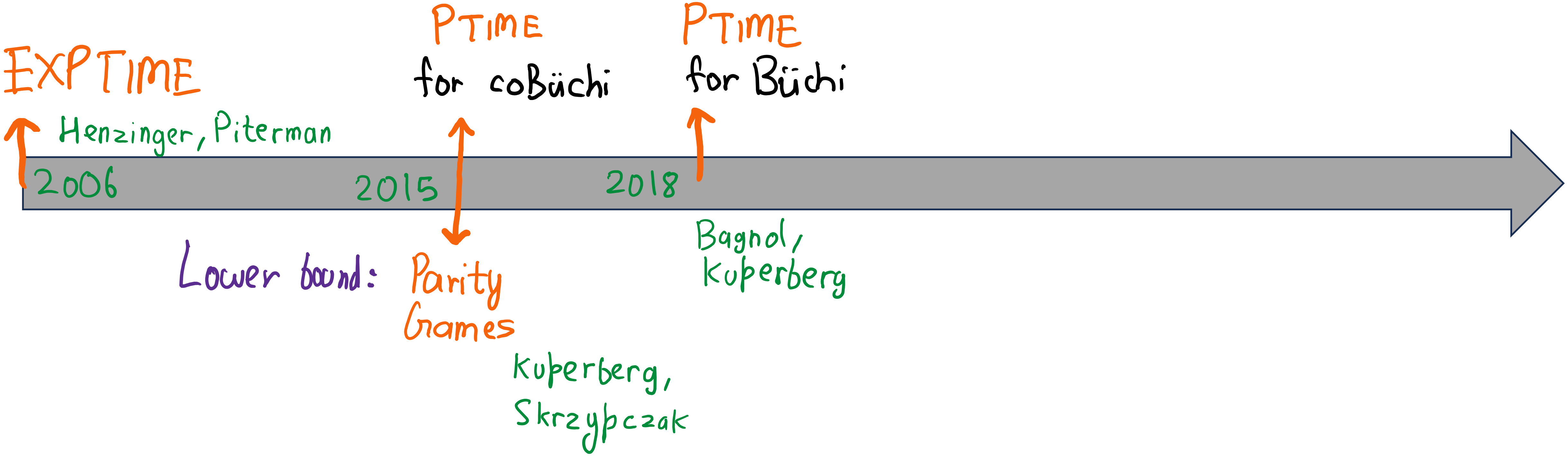
Recognising HD Parity Automata

Given a parity automaton $\mathcal{S}$, is $\mathcal{S}$ HD?



Recognising HD Parity Automata

Given a parity automaton S, is S HD?



Recognising HD Parity Automata

Given a parity automaton $\mathcal{S}$, is $\mathcal{S}$ HD?

EXPTIME

Henzinger, Piterman

2006

PTIME for coBüchi

2015

Lower bound: Parity Games

Kuperberg, Skrzypczak

2-token game

PTIME for Büchi

2018

Bagnol, Kuperberg

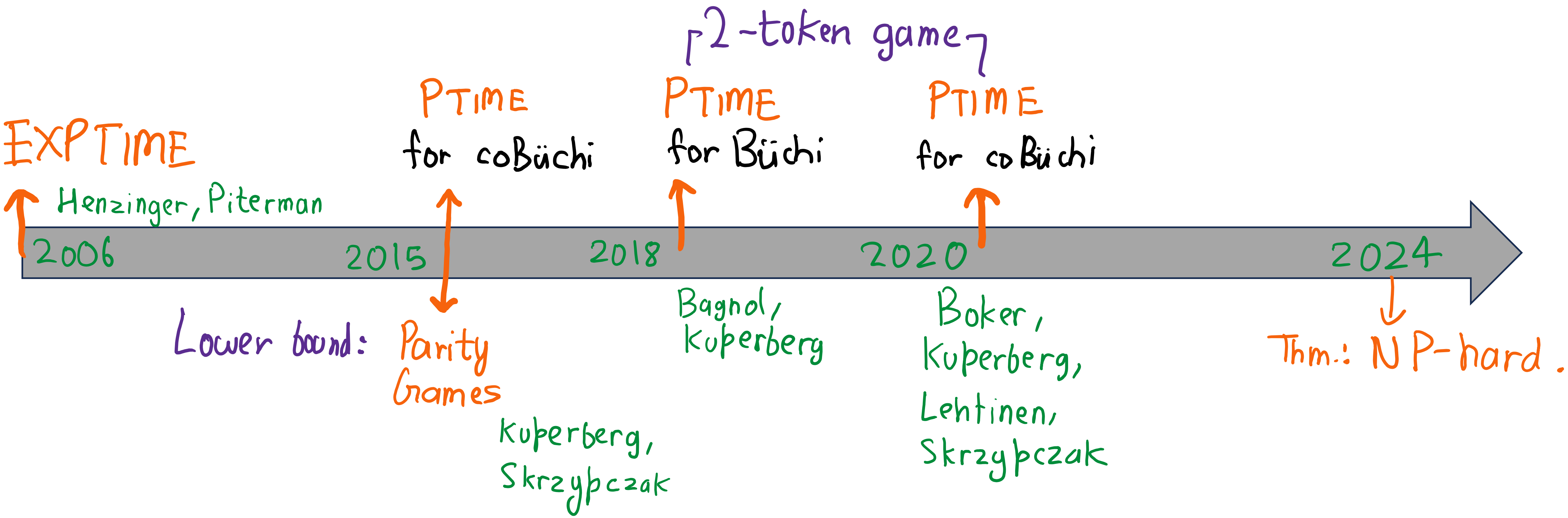
PTIME for coBüchi

2020

Boker, Kuperberg, Lehtinen, Skrzypczak

Recognising HD Parity Automata

Given a parity automaton $\mathcal{S}$, is $\mathcal{S}$ HD?



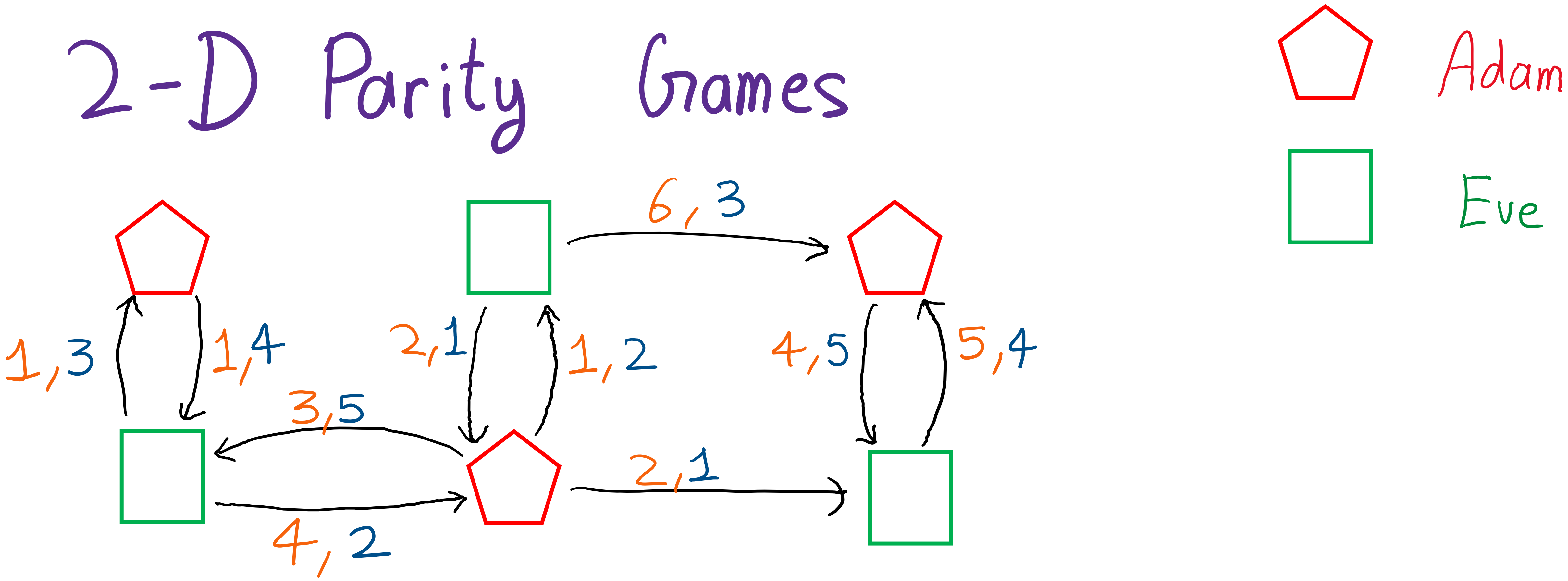
2. NP-hardness

2. NP-hardness

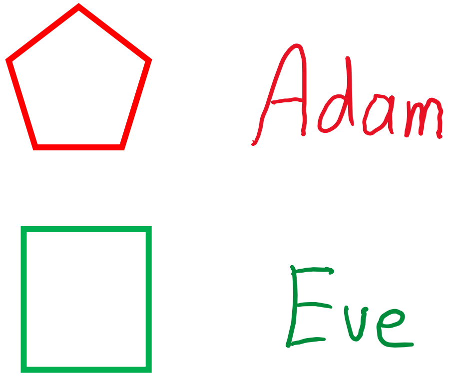
a. Simulation

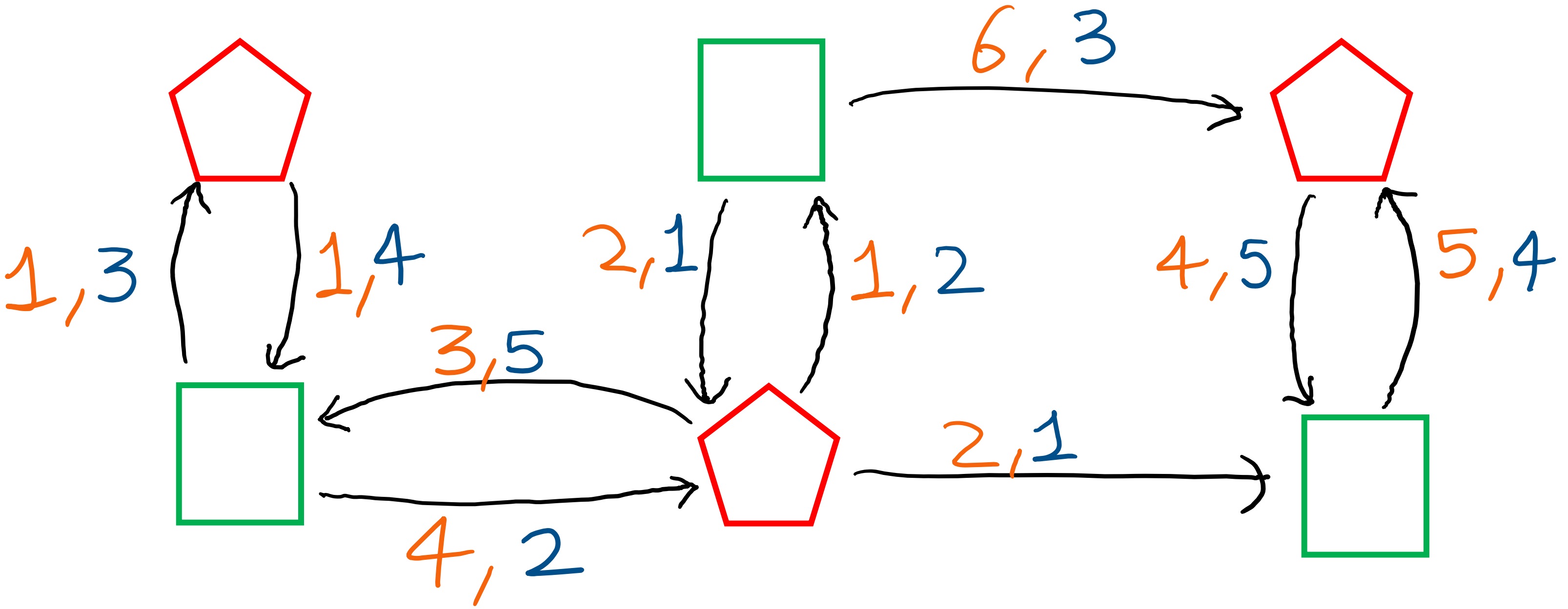
b. Checking History-Determinism

2-D Parity Games



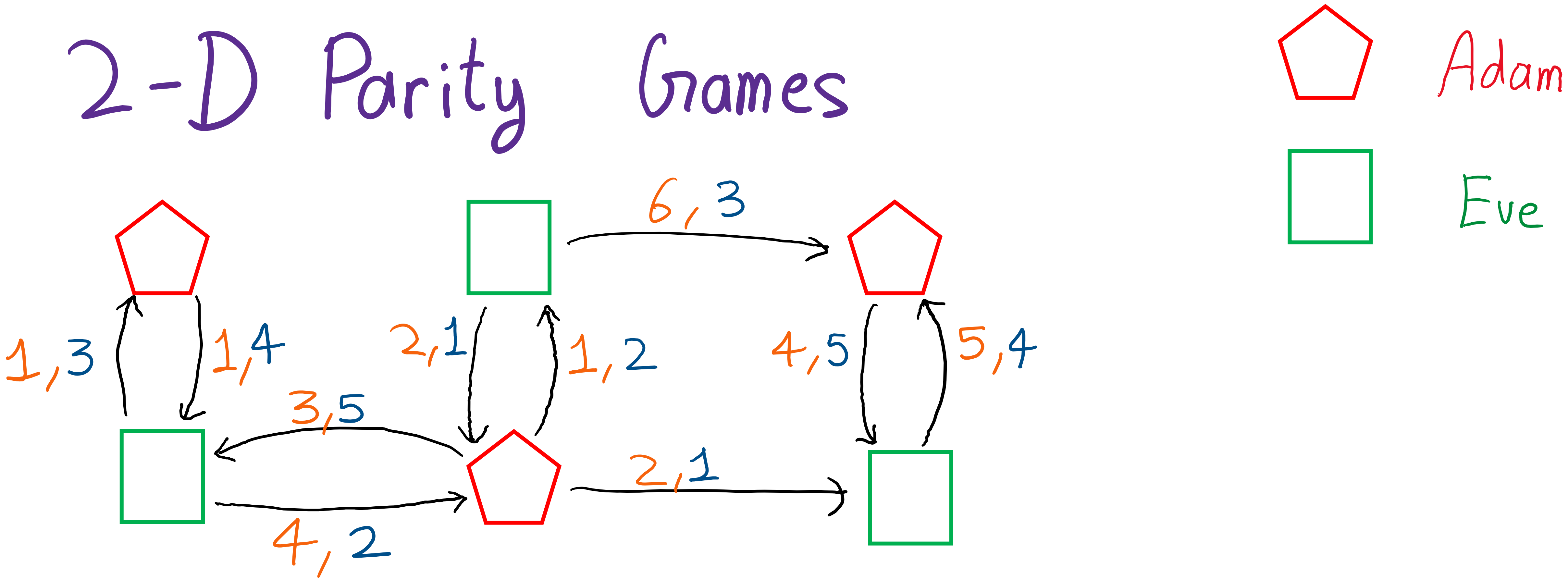
2-D Parity Games





(1, 3) (1, 4) (4, 2) (2, 1) (6, 3) ...

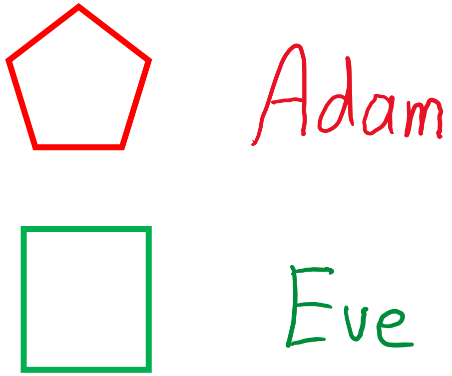
2-D Parity Games

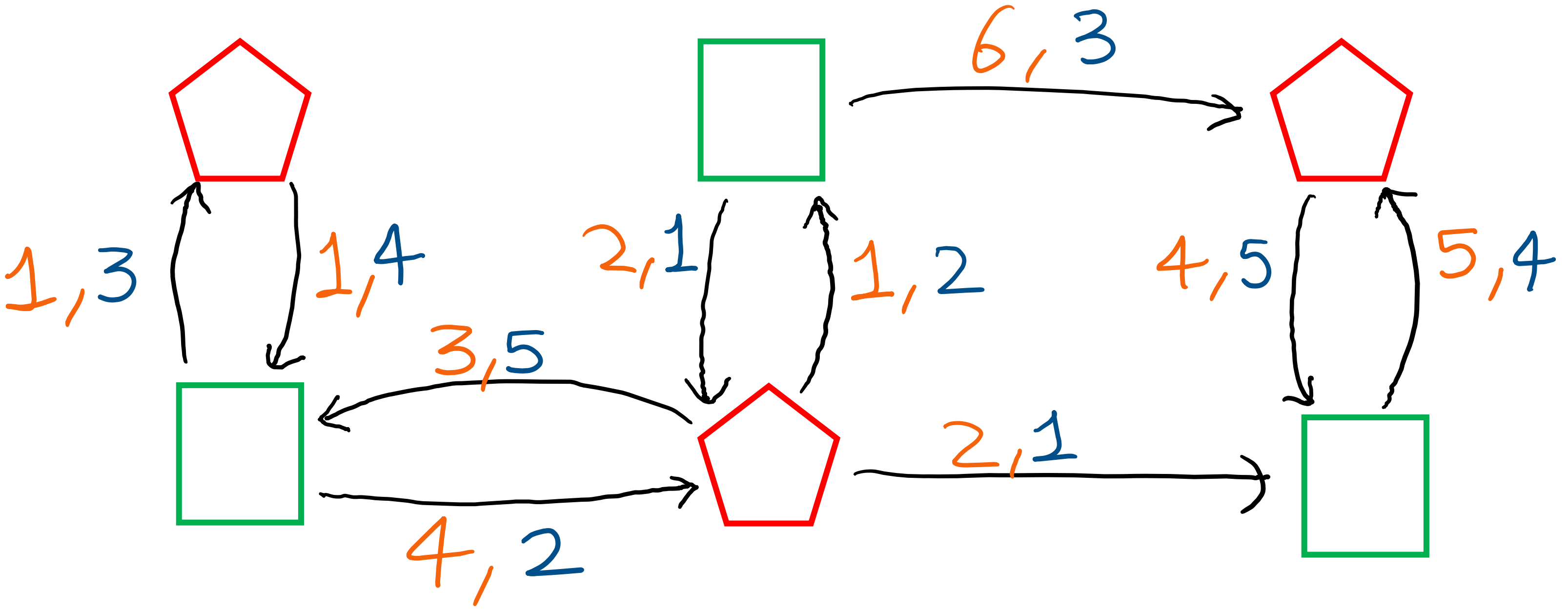


(1, 3) (1, 4) (4, 2) (2, 1) (6, 3) ...

Winning condⁿ for Eve: Play satisfies Orange or Blue parity condⁿ.

2-D Parity Games



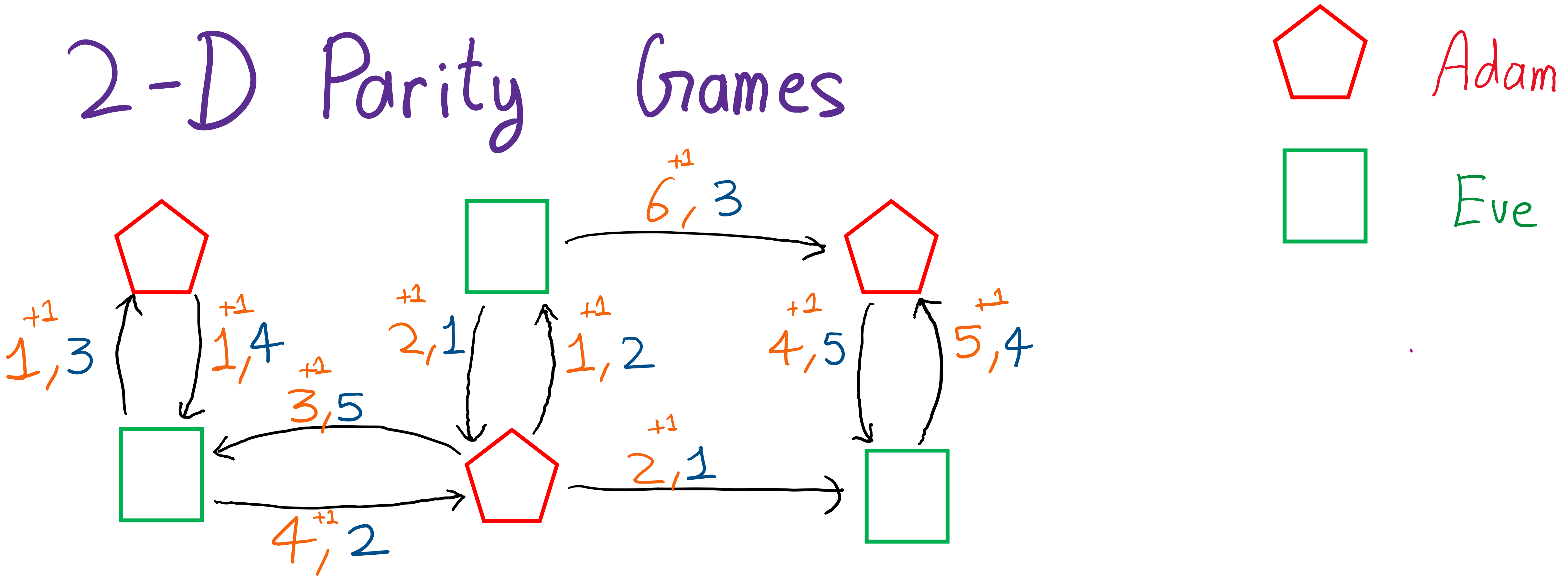


(1,3) (1,4) (4,2) (2,1) (6,3) ...

Winning cond^n for Eve: Play satisfies Orange or Blue parity cond^n.

$$(O \text{ or } B) \iff (\neg O \Rightarrow B)$$

2-D Parity Games

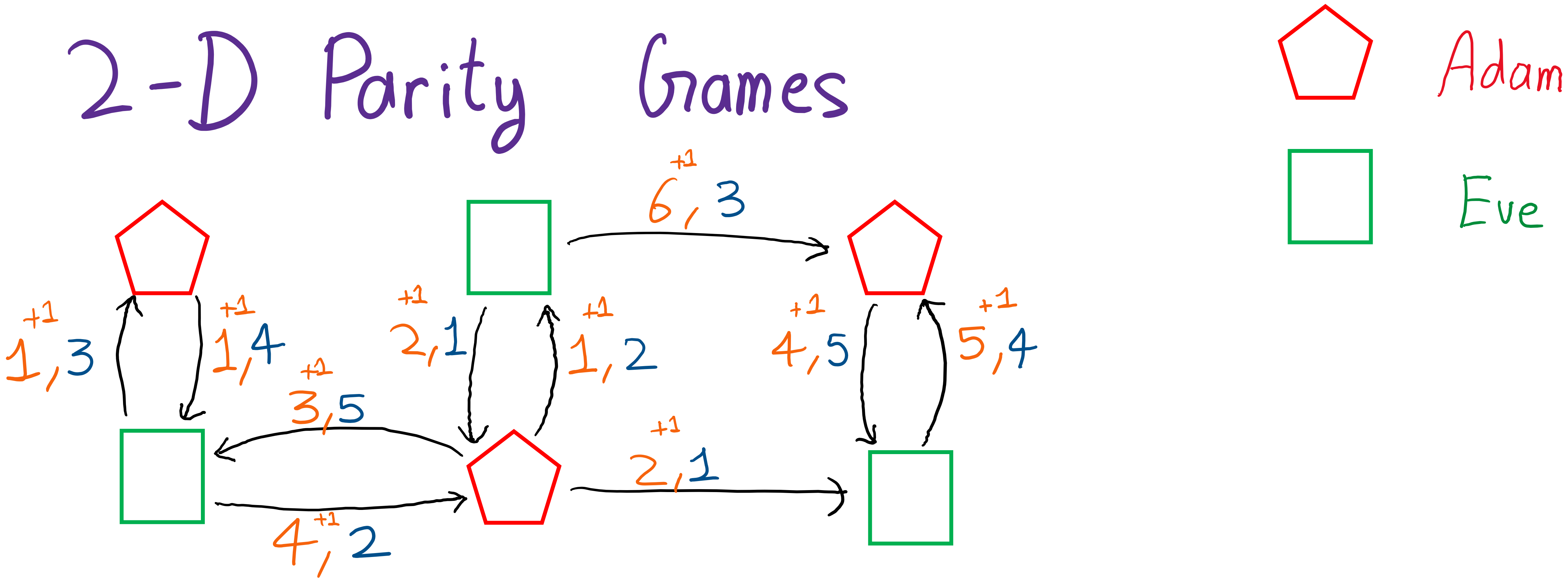


(1,3) (1,4) (4,2) (2,1) (6,3) ...

Winning condⁿ for Eve: Play satisfies Orange or Blue parity condⁿ.

$(O \text{ or } B) \iff (\neg O \Rightarrow B)$

2-D Parity Games



(1, 3) (1, 4) (4, 2) (2, 1) (6, 3) ...

Winning condⁿ for Eve: Play satisfies

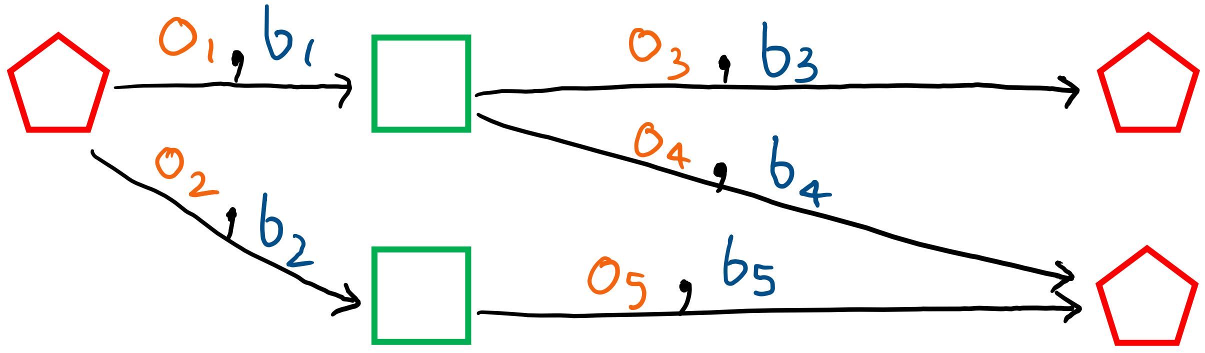
O ⇒ B

2-D Parity Games

Chatterjee, Henzinger, Piterman '05

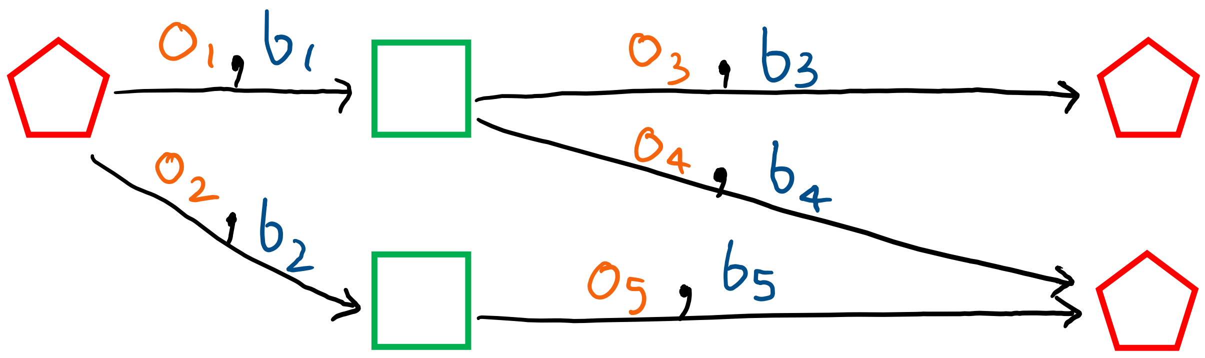
Deciding if Eve wins a 2-D parity game is NP-complete.

Reduction to Simulation



Winning cond^{n}: $O \Rightarrow B$

Reduction to Simulation



Winning cond^n: $O \Rightarrow B$

Simulation game

In round i:

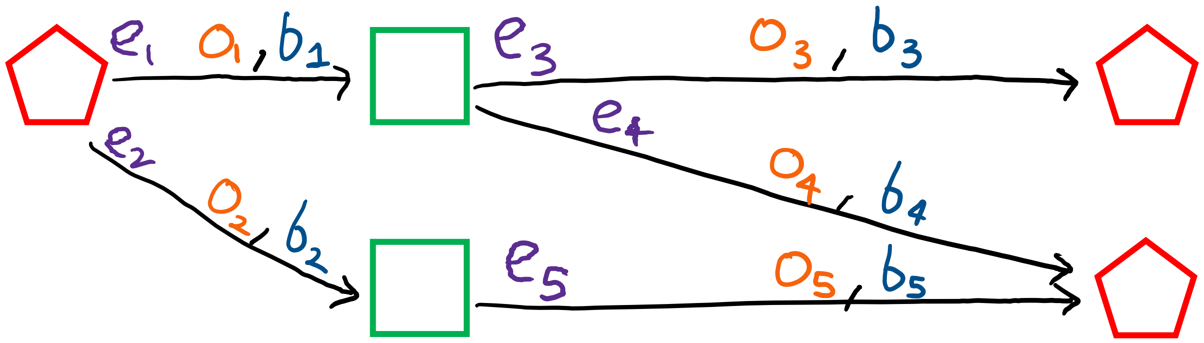
1. Adam selects a_i
2. Adam selects $p_i \xrightarrow{a_i} p_{i+1}$ in I
3. Eve selects $s_i \xrightarrow{a_i} s_{i+1}$ in S

Winning cond^n:

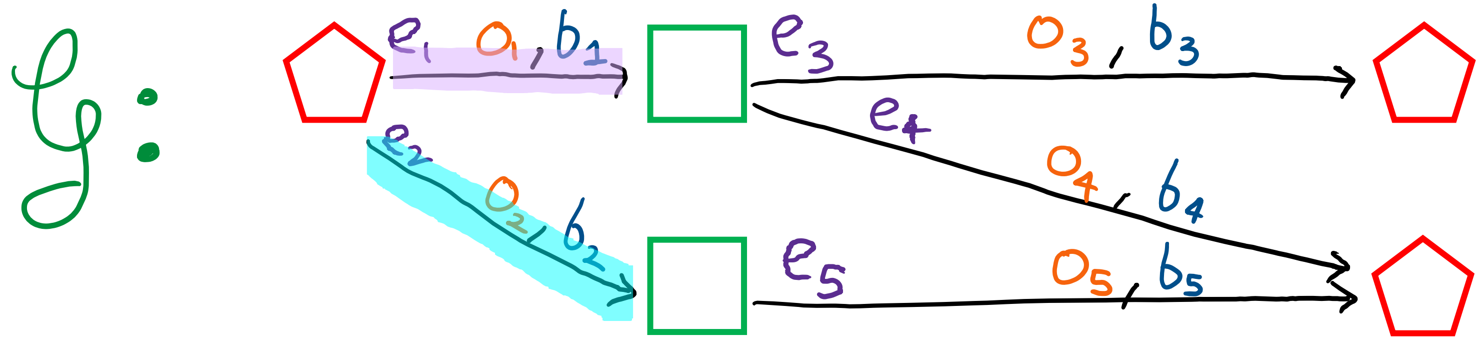
Adam's run in I is accepting

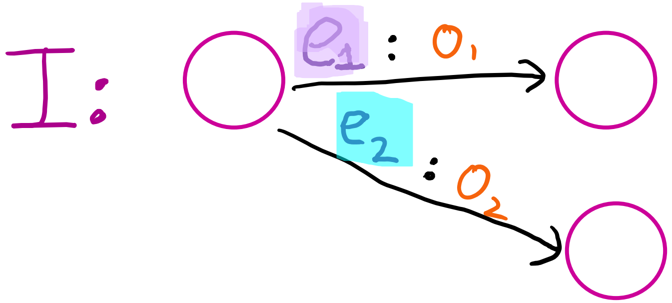
$\Rightarrow$ Eve's run in S is accepting

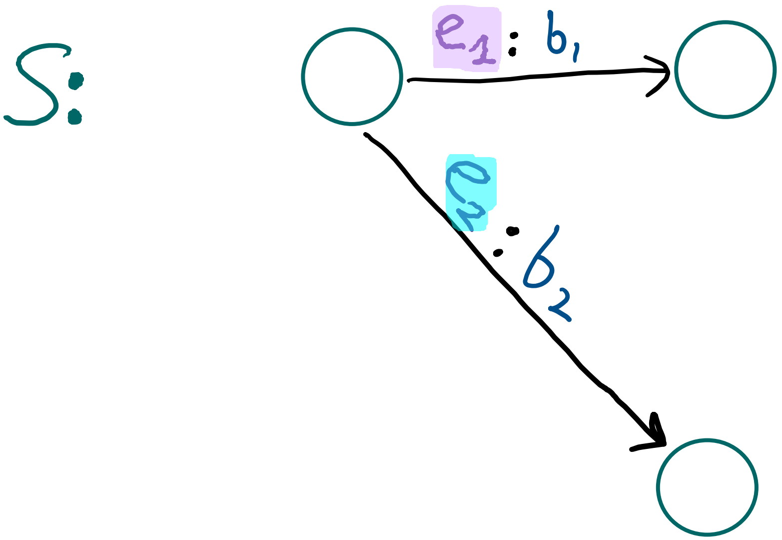
$\mathcal{G}$:



$$\text{Eve wins } \mathcal{G} \iff \mathcal{S} \text{ simulates } \mathcal{I}$$



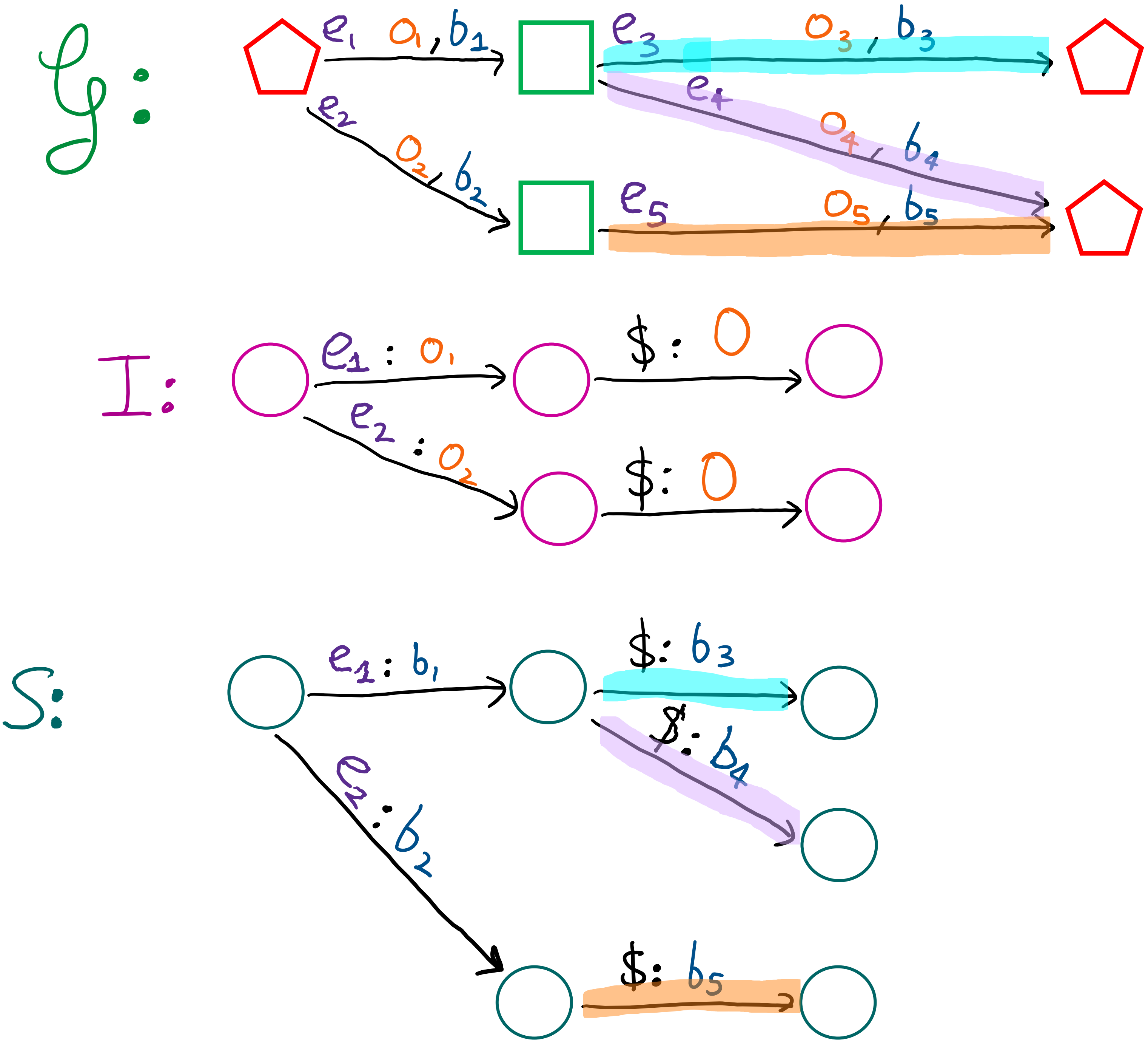




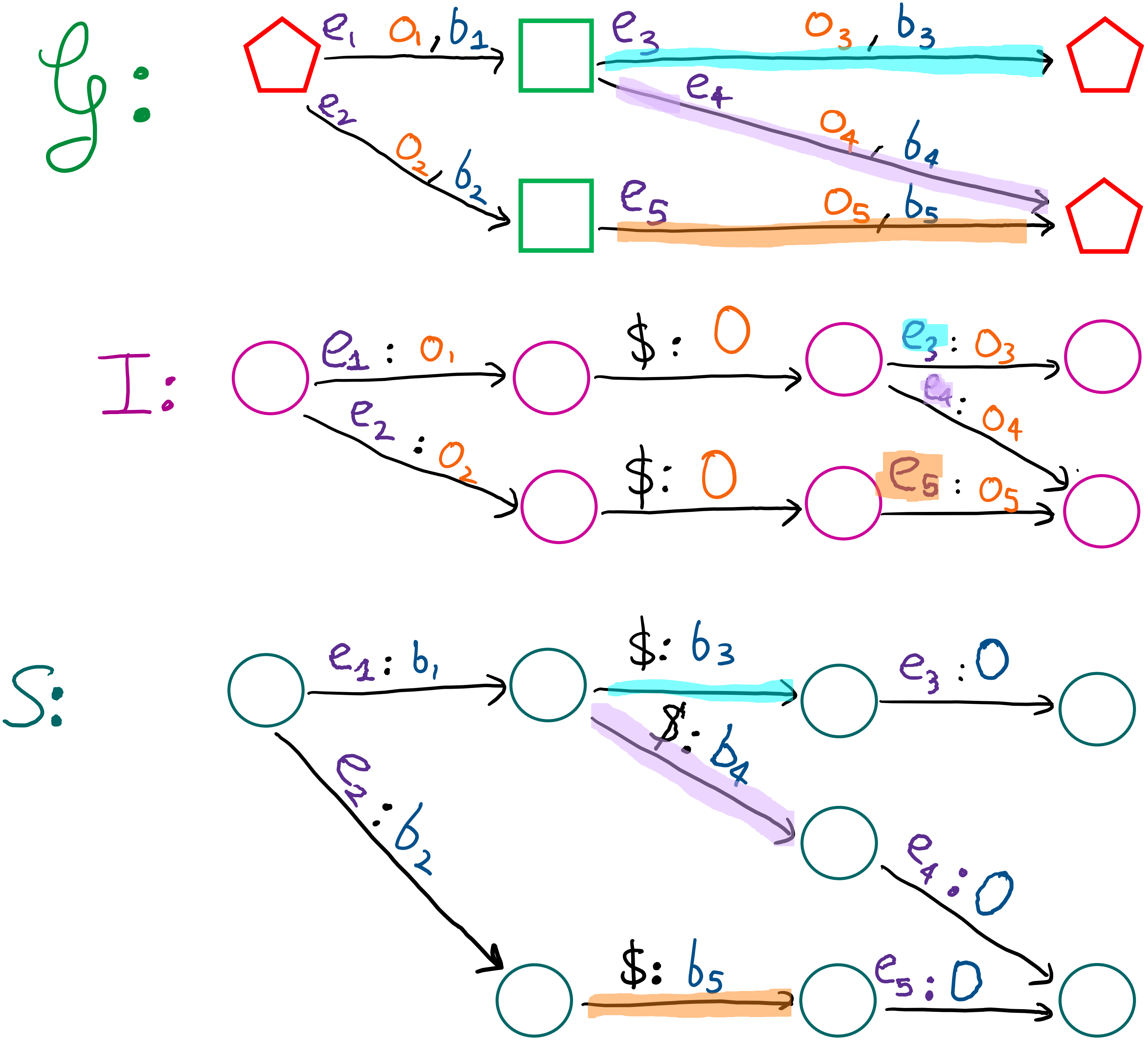
Eve wins $\mathcal{G}$

$\Longleftrightarrow$

$\mathcal{S}$ simulates $\mathcal{I}$



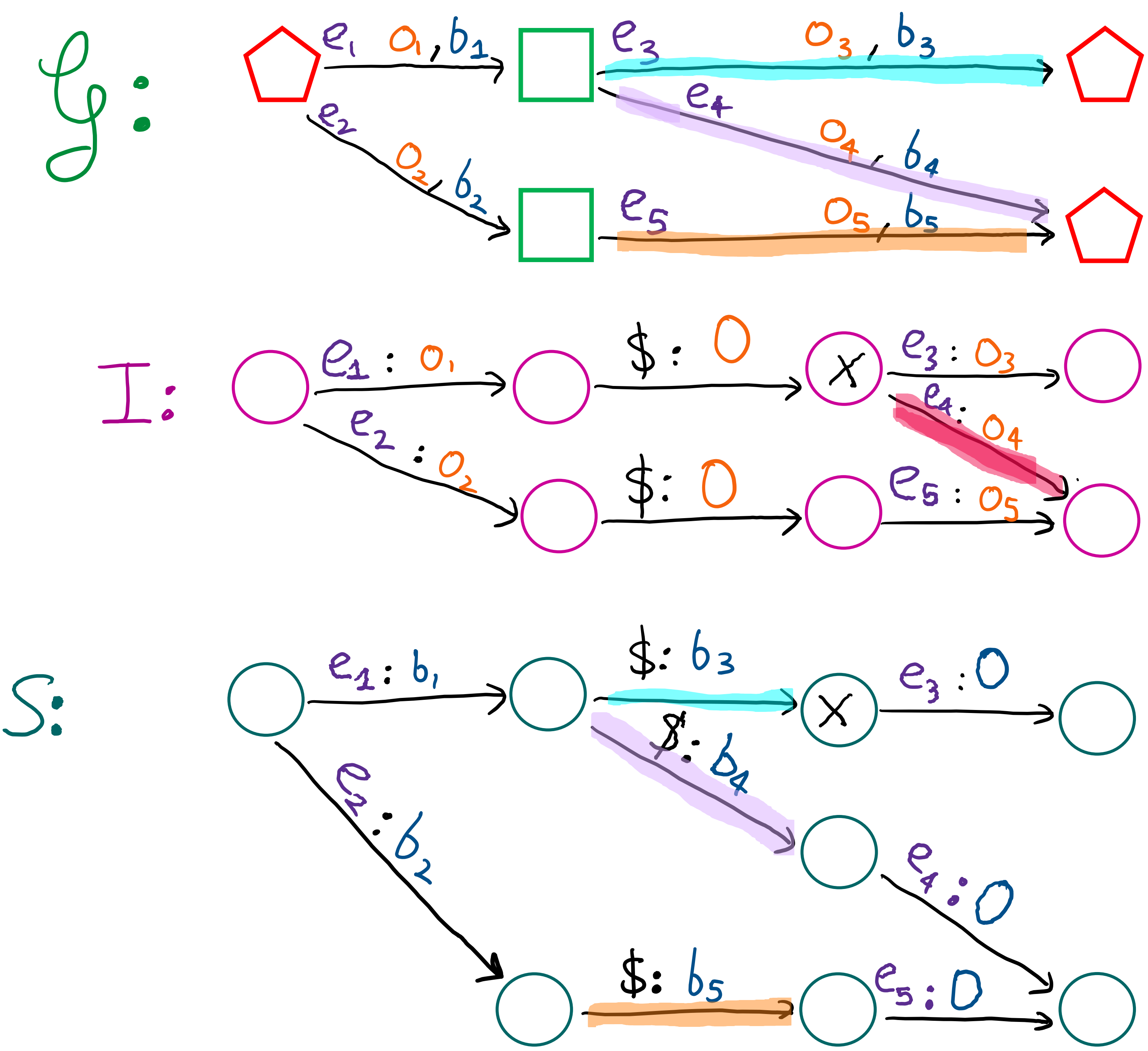
Eve wins $\mathcal{G}$ $\iff$ S simulates I



Eve wins $\mathcal{G}$

$\iff$

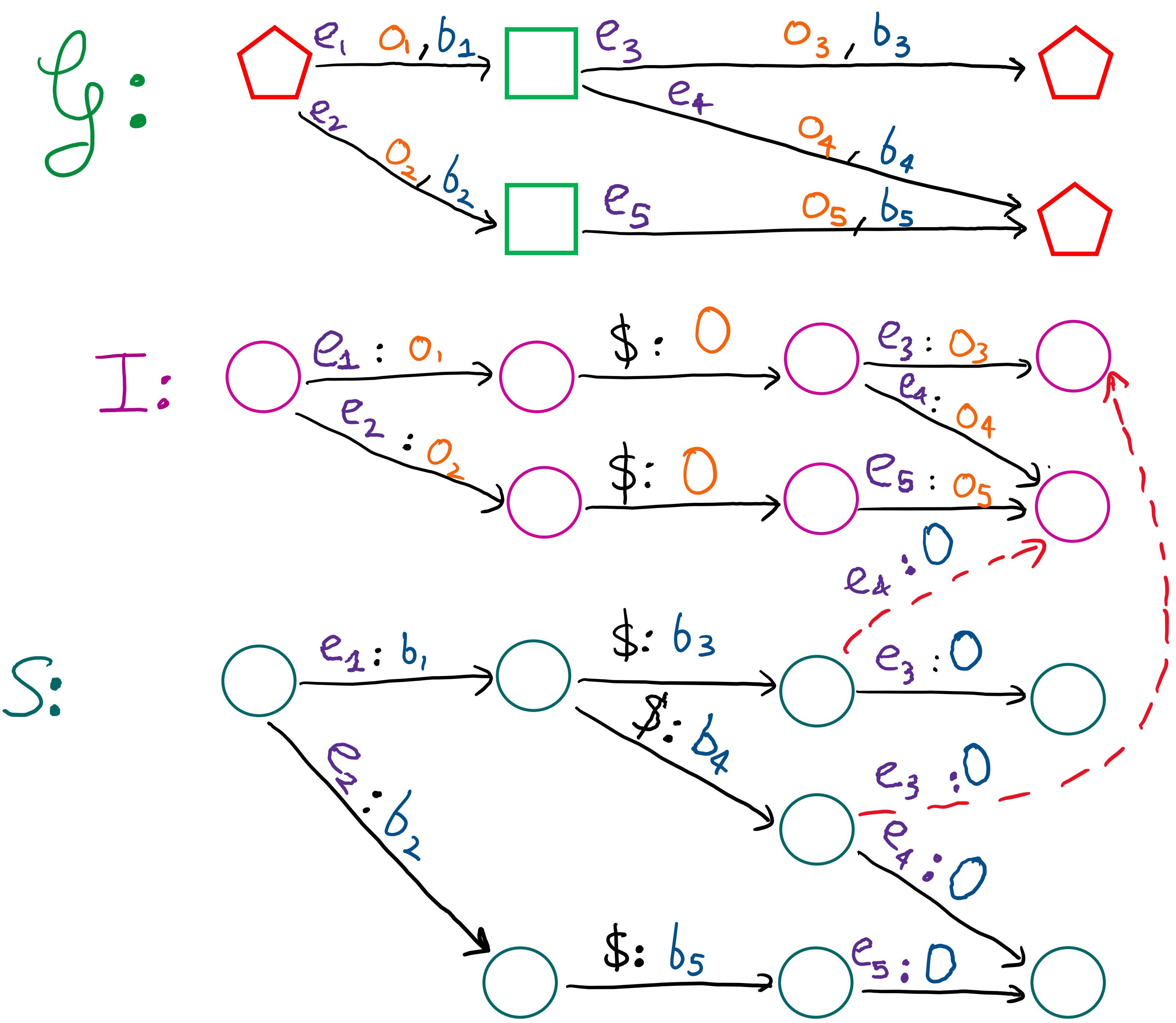
$\mathcal{S}$ simulates $\mathcal{I}$



Eve wins $\mathcal{G}$

$\iff$

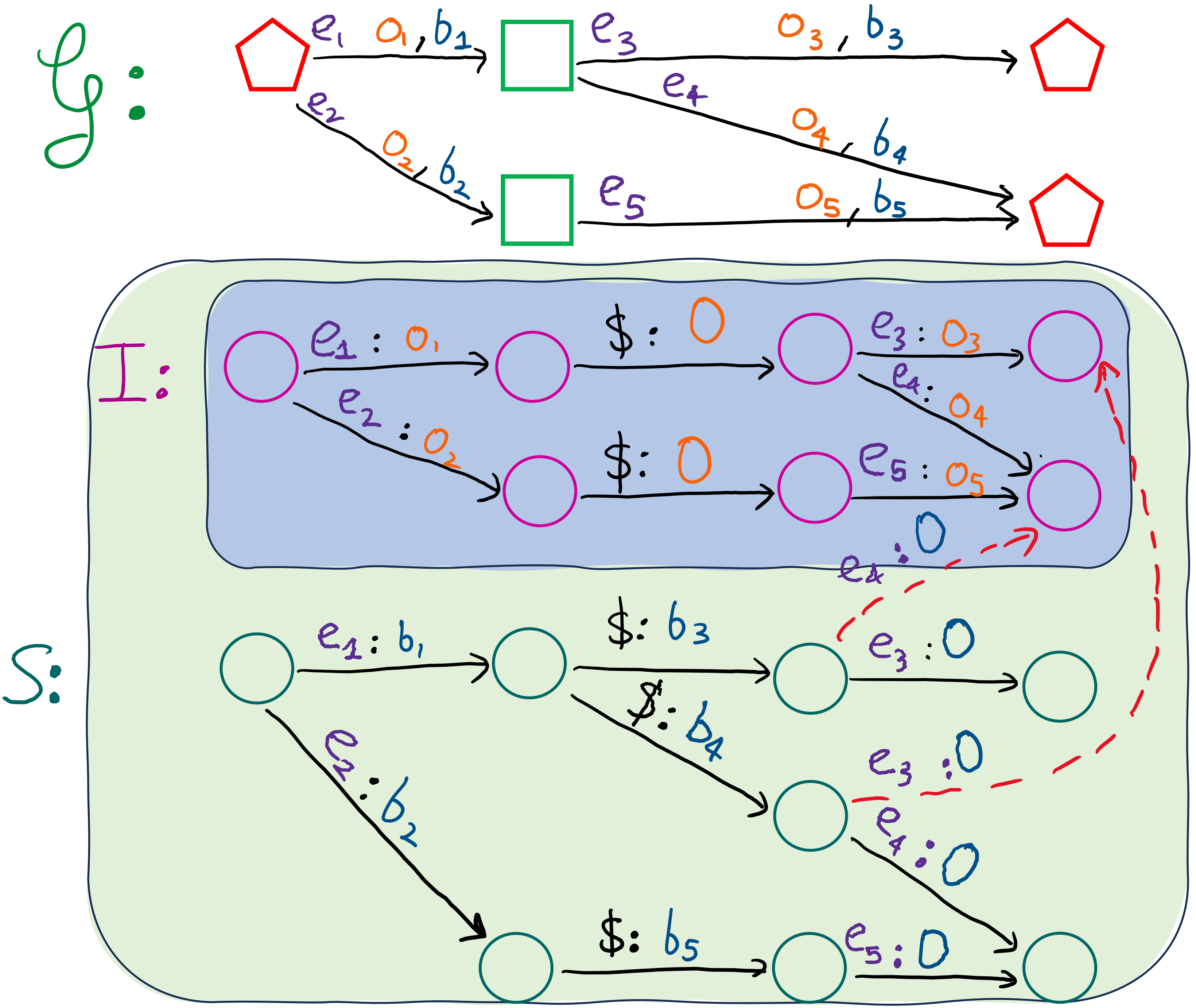
$\mathcal{S}$ simulates $\mathcal{I}$



Eve wins $\mathcal{G}$

$\Longleftrightarrow$

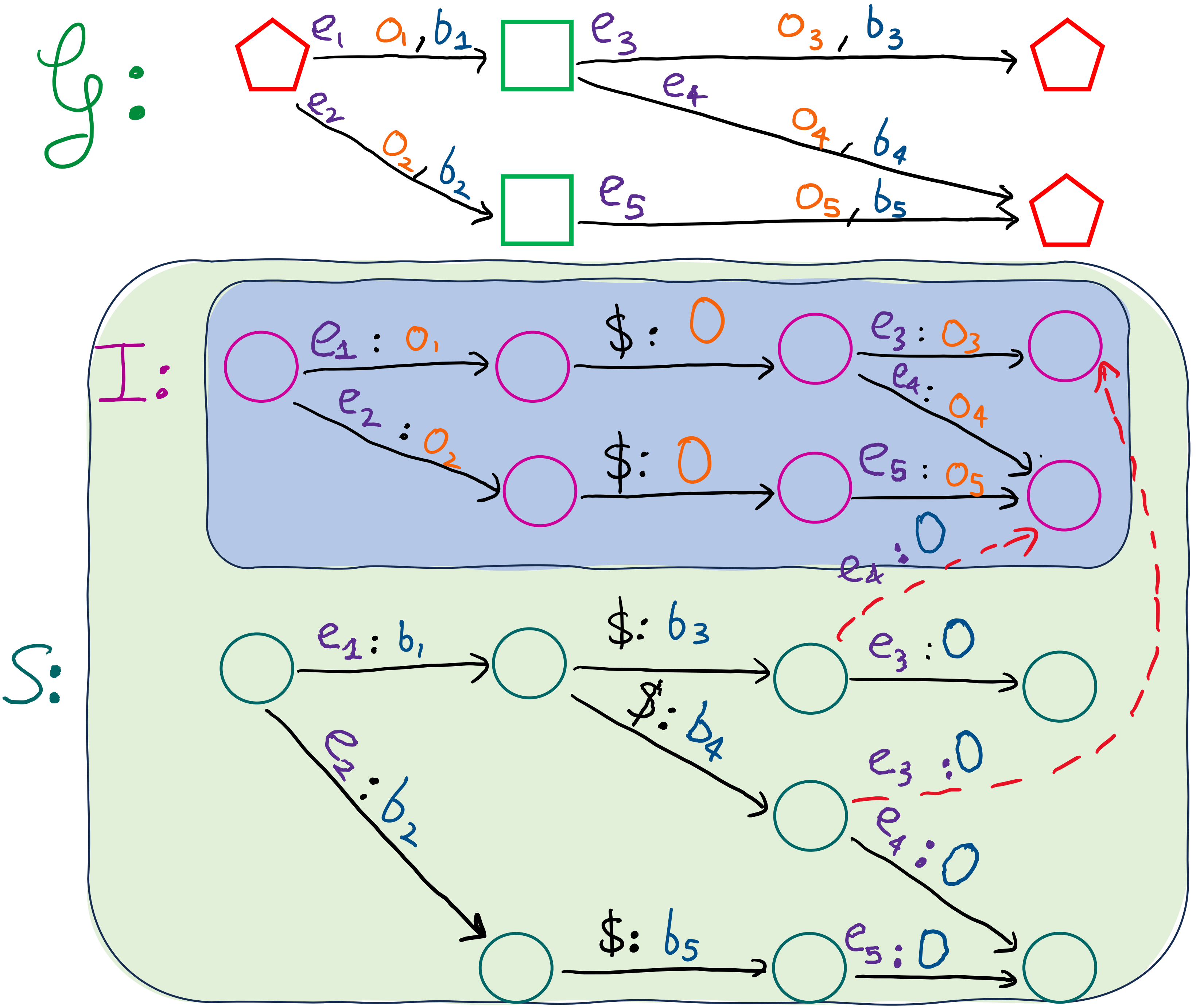
$\mathcal{S}$ simulates $\mathcal{I}$



Eve wins $\mathcal{G}$

$\Longleftrightarrow$

$\mathcal{S}$ simulates $\mathcal{I}$



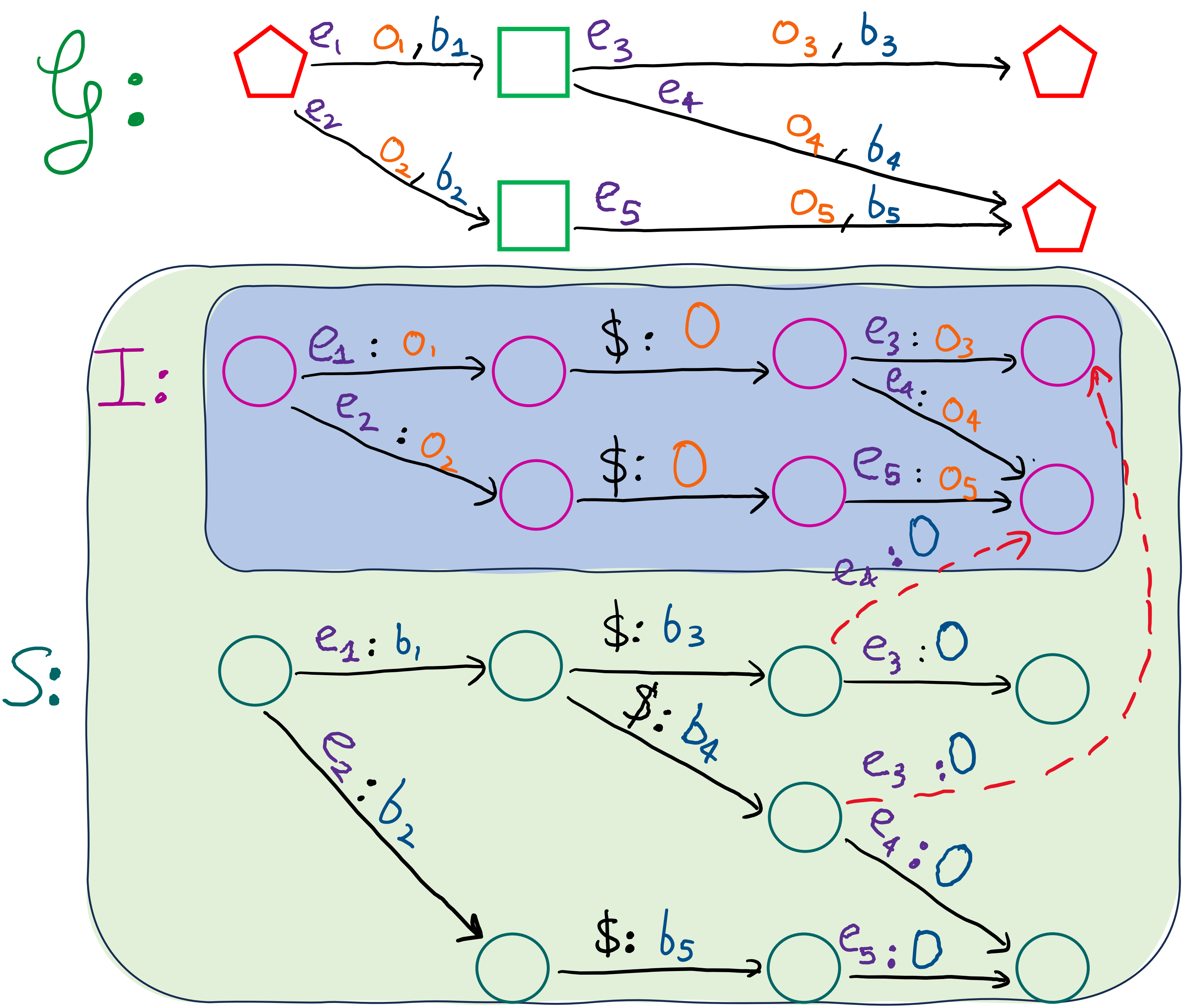
Eve wins $\mathcal{G}$

$\Longleftrightarrow$

S simulates I

Theorem 2: Deciding simulation is NP-complete.

26. NP-hardness for checking History-Determinism

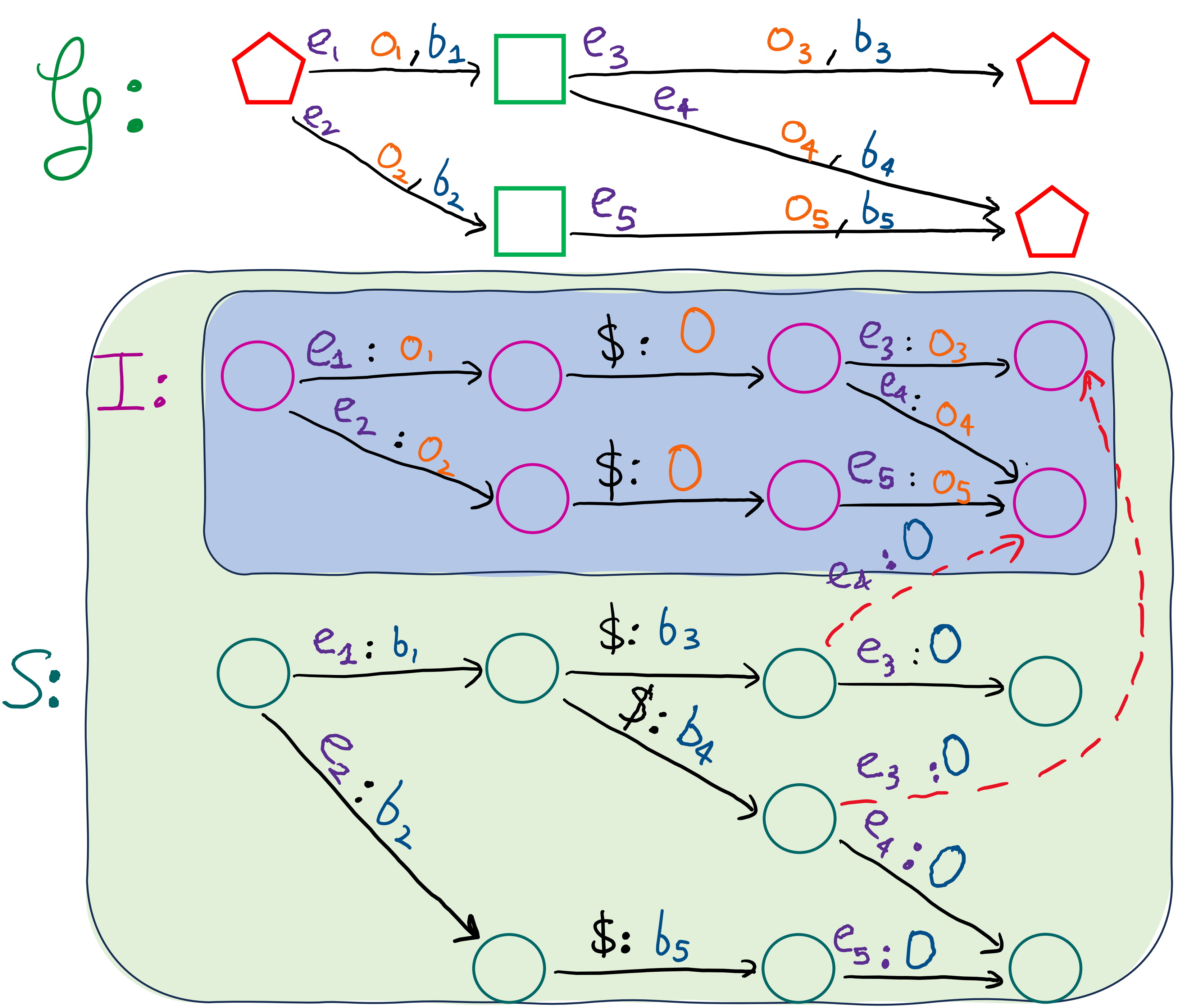


Good 2-D parity games:

Paths satisfying B

$\subseteq$

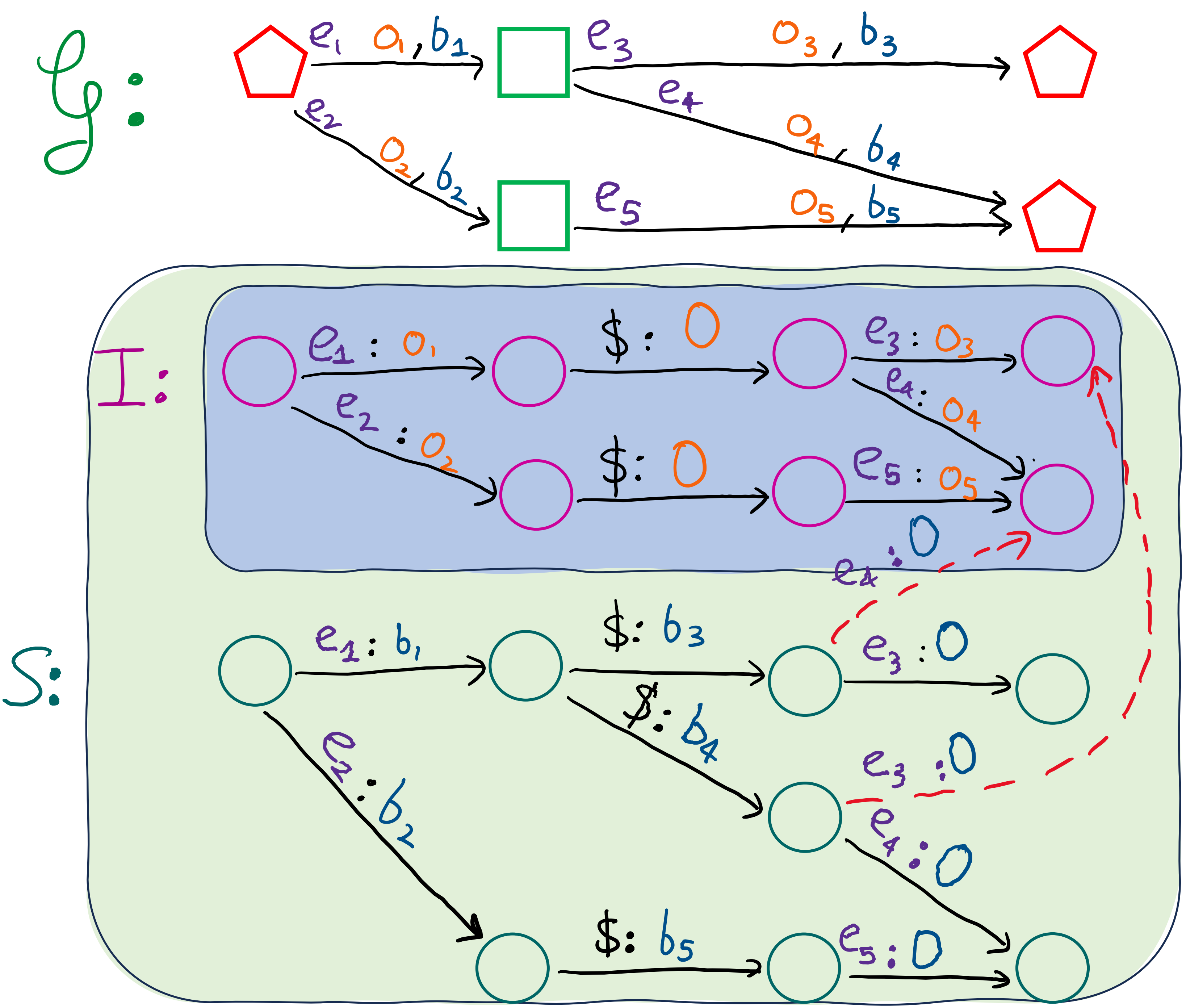
Paths satisfying O



Good 2-D parity games:

Paths satisfying B $\subseteq$ Paths satisfying O

Lemma: Deciding if Eve wins $\mathcal{G}$ is NP-hard.



Good 2-D parity games:

Paths satisfying B

$\subseteq$

Paths satisfying O

Lemma: Deciding if Eve wins $\mathcal{G}$ is NP-hard.

Eve wins $\mathcal{G}$

$\Longleftrightarrow$

S is HD

Theorem: Checking history-determinism is NP-hard for parity automata.

Theorem: Checking history-determinism is NP-hard for parity automata.

Open: What is the complexity of checking history-determinism?

Upper bound: EXPTIME

Henzinger, Piterman'06

Theorem: Checking history-determinism is NP-hard for parity automata.

Open: What is the complexity of checking history-determinism?

Upper bound: EXPTIME — Henzinger, Piterman '06

2-token conjecture $\Rightarrow$ PSPACE upper bound.

Bagnol, Kuperberg 2018