Robot Games with States in Dimension One

Reino Niskanen

Department of Computer Science
University of Liverpool, UK

10th International Workshop on Reachability Problems

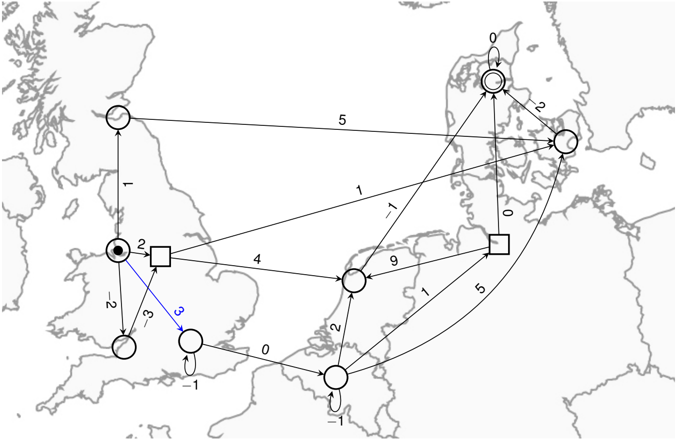
Introduction

Counter reachability games

Stress level

0

Counter reachability games



Stress level

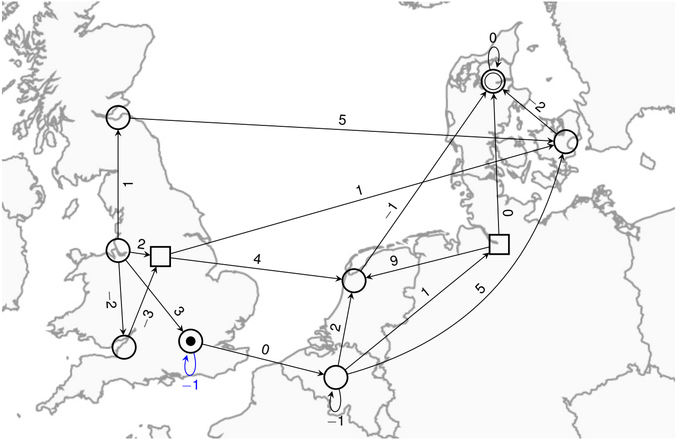
0

Counter reachability games

Stress level

$0 \rightarrow 3$

Counter reachability games



Stress level

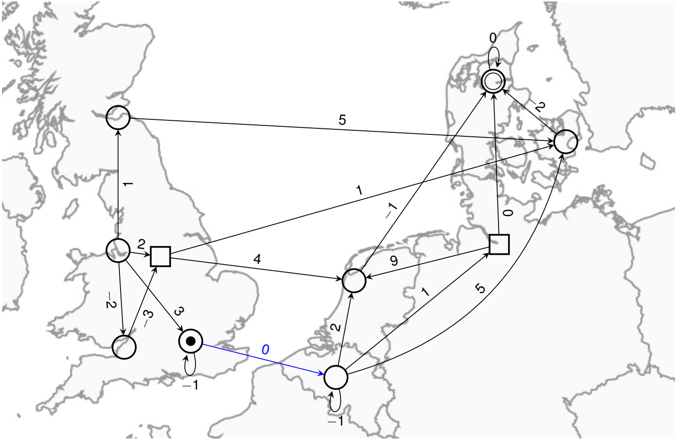
$0 \rightarrow 3$

Counter reachability games

Stress level

$0 \to 3 \to 2$

Counter reachability games



Stress level

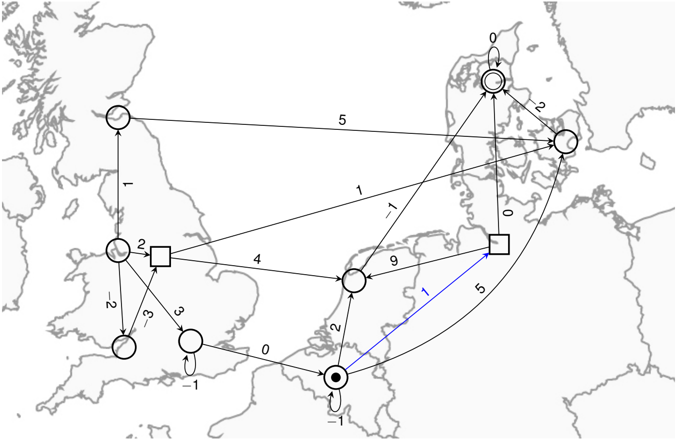
$0 \to 3 \to 2$

Counter reachability games

Stress level

$0 \to 3 \to 2 \to 2$

Counter reachability games



Stress level

$0 \to 3 \to 2 \to 2$

Counter reachability games

Stress level

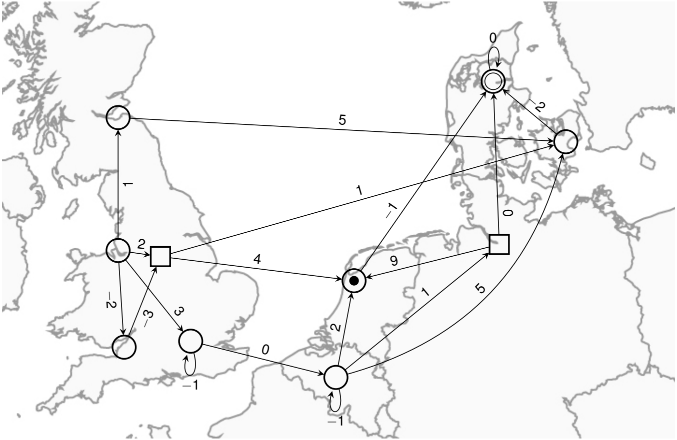
$0 \to 3 \to 2 \to 2 \to 3$

Counter reachability games

Stress level

$0 \to 3 \to 2 \to 2 \to 3$

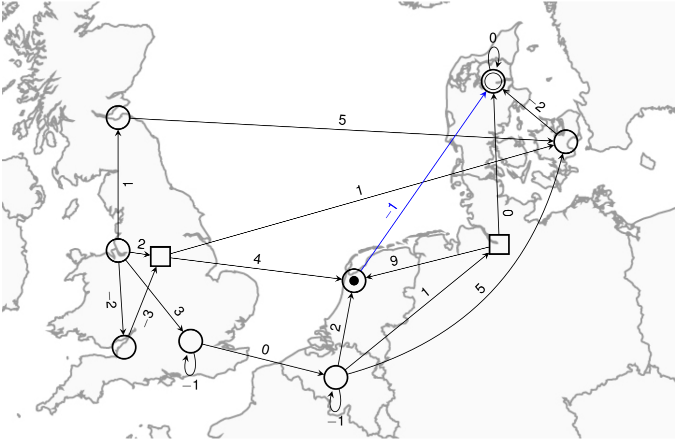
Counter reachability games



Stress level

$0 \to 3 \to 2 \to 2 \to 3 \to 12$

Counter reachability games



Stress level

$0 \to 3 \to 2 \to 2 \to 3 \to 12$

Counter reachability games

Stress level

$0 \to 3 \to 2 \to 2 \to 3 \to 12 \to 13$

Counter reachability games

Stress level

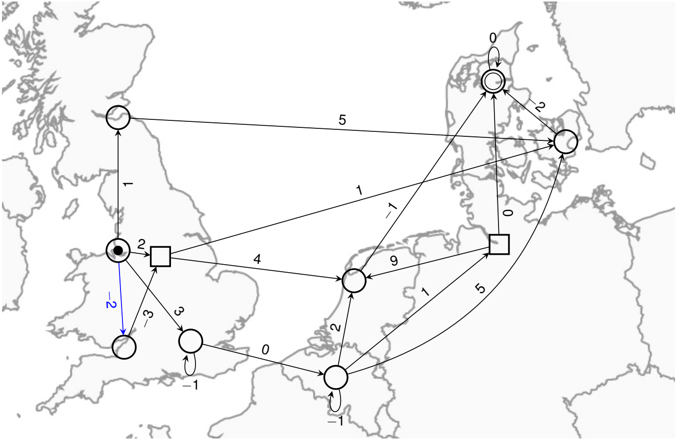
$0 \to 3 \to 2 \to 2 \to 3 \to 12 \to 13 \to \cdots$

Counter reachability games

Stress level

0

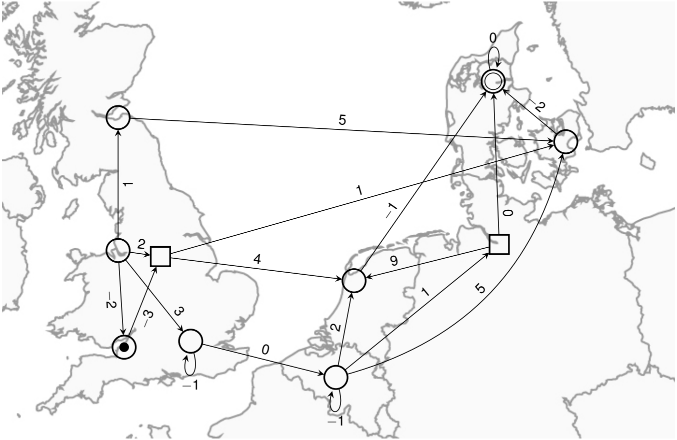
Counter reachability games



Stress level

0

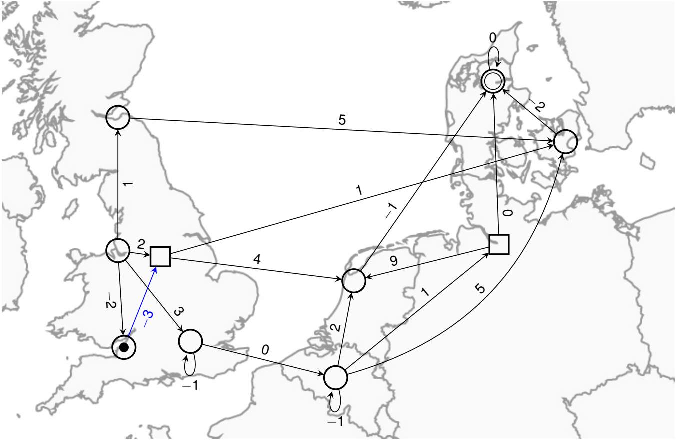
Counter reachability games



Stress level

$0 \to -2$

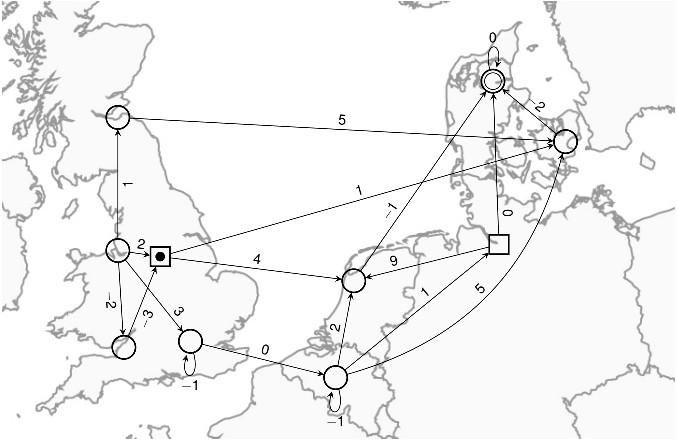
Counter reachability games



Stress level

$0 \to -2$

Counter reachability games



Stress level

$0 \to -2 \to -5$

Counter reachability games

Stress level

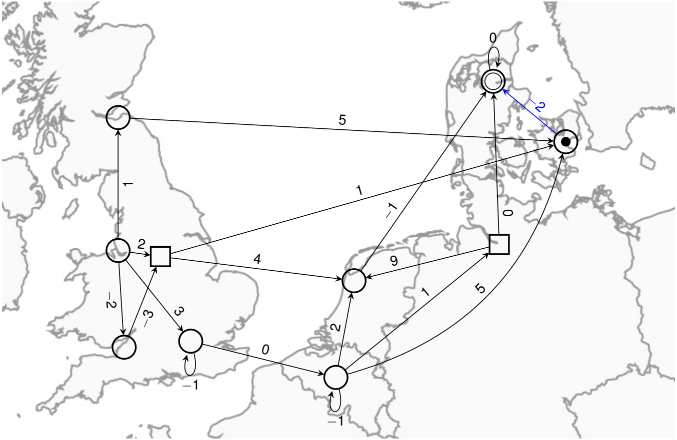
$0 \to -2 \to -5$

Counter reachability games

Stress level

$0 \to -2 \to -5 \to -4$

Counter reachability games



Stress level

$0 \to -2 \to -5 \to -4$

Counter reachability games

Stress level

$0 \to -2 \to -5 \to -4 \to -6$

Counter reachability games

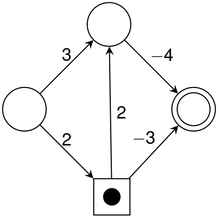
Stress level

$0 \to -2 \to -5 \to -4 \to -6 \to \cdots$

Definitions

Counter reachability games

- Played on a labeled directed graph $G = (V, E)$ with edges labeled by $\mathbf{x} \in \mathbb{Z}^n$.
- Two players: Eve ($\bigcirc$) , Adam ($\Box$).
- A configuration $[v, \mathbf{z}] \in V \times \mathbb{Z}^n$.
- A successor configuration is $[v', \mathbf{z} + \mathbf{z}']$, where $[v, \mathbf{z}', v'] \in E$ and the owner of v chose it.

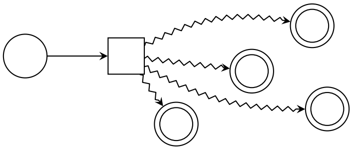


- The initial and target configurations.
- A play is a finite or an infinite sequence of configurations.
- Eve wins if the target configuration is reachable in a play starting from the initial configuration. Otherwise Adam wins.

Counter reachability games

A winning strategy

Eve has a winning strategy if the target configuration is reachable for every choice of Adam.



The decision problem

Given a graph $G = (V, E)$*, initial and target configurations,* $[v_0, \mathbf{z}_0]$ *and* $[v_f, (0, \ldots, 0)]$*. Does there exist a winning strategy for Eve to reach* $[v_f, (0, \ldots, 0)]$ *from* $[v_0, \mathbf{z}_0]$*?*

Counter reachability games

The decision problem

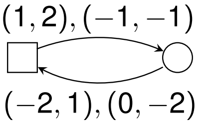
Given a graph $G = (V, E)$, initial and target configurations, $[v_0, \mathbf{z}_0]$ and $[v_f, (0, \ldots, 0)]$. Does there exist a winning strategy for Eve to reach $[v_f, (0, \ldots, 0)]$ from $[v_0, \mathbf{z}_0]$?

Known for counter reachability games:

One-dimensional	**EXPSPACE**-complete
Two-dimensional	Undecidable

Robot games

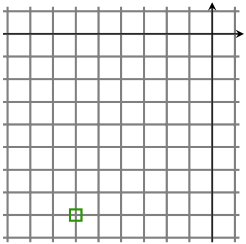
What if we have a simpler graph?



- Proposed by Doyen and Rabinovich in 2011.
- **EXPTIME**-complete in dimension one [Arul, Reichert, QAPL 2013].
- Undecidable in dimension two [N., Potapov, Reichert, MFCS 2016].

Another way to look at robot games

- Played on integer lattice $\mathbb{Z}^n$.
- Adam and Eve move a token on the lattice.
- Eve's goal is to reach $(0, \ldots, 0)$. Adam's goal is to avoid it.

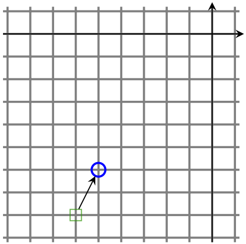


Adam's moves: $\{(1, 2), (2, 0)\}$

Eve's moves: $\{(2, 2), (1, 4)\}$

Another way to look at robot games

- Played on integer lattice $\mathbb{Z}^n$.
- Adam and Eve move a token on the lattice.
- Eve's goal is to reach $(0, \ldots, 0)$. Adam's goal is to avoid it.

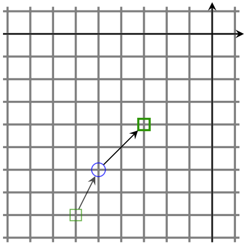


Adam's moves: $\{(1, 2), (2, 0)\}$

Eve's moves: $\{(2, 2), (1, 4)\}$

Another way to look at robot games

- Played on integer lattice $\mathbb{Z}^n$.
- Adam and Eve move a token on the lattice.
- Eve's goal is to reach $(0, \ldots, 0)$. Adam's goal is to avoid it.

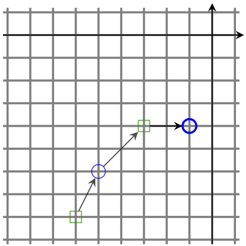


Adam's moves: $\{(1, 2), (2, 0)\}$

Eve's moves: $\{(2, 2), (1, 4)\}$

Another way to look at robot games

- Played on integer lattice $\mathbb{Z}^n$.
- Adam and Eve move a token on the lattice.
- Eve's goal is to reach $(0, \ldots, 0)$. Adam's goal is to avoid it.

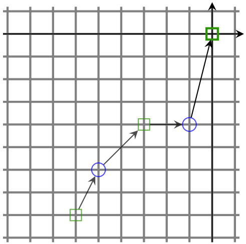


Adam's moves: $\{(1,2),(2,0)\}$

Eve's moves: $\{(2,2),(1,4)\}$

Another way to look at robot games

- Played on integer lattice $\mathbb{Z}^n$.
- Adam and Eve move a token on the lattice.
- Eve's goal is to reach $(0, \ldots, 0)$. Adam's goal is to avoid it.

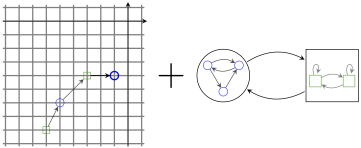


Adam's moves: $\{(1,2),(2,0)\}$

Eve's moves: $\{(2,2),(1,4)\}$

Robot games with states

- A mix between counter reachability games and robot games.
- Robot games but players have internal states as well.
- Undecidable in dimension two [N., Potapov, Reichert, MFCS 2016].



Known results

Game	Dimension	
	1	≥ 2
counter reachability games	**EXPSPACE**-complete	U
robot games with states	?	U
robot games	**EXPTIME**-complete	U

Known results

Game	Dimension	
	1	≥ 2
counter reachability games	**EXPSPACE**-complete	U
robot games with states	?	U
robot games	**EXPTIME**-complete	U

Theorem

Whether Eve has a winning strategy in a one-dimensional robot game with states is ***EXPSPACE****-complete.*

Robot games with states

Inherited complexity bounds

Lemma

*Deciding winner is **EXPTIME**-hard.*

Lemma

*Deciding winner is in **EXPSPACE**.*

Inherited complexity bounds

Lemma

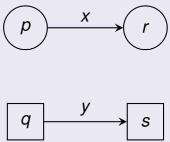
*Deciding winner is **EXPTIME**-hard.*

Lemma

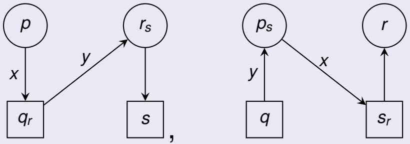
*Deciding winner is in **EXPSPACE**.*

Proof.

In RGS:



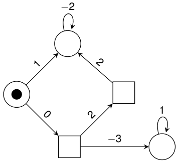
In CRG:



□

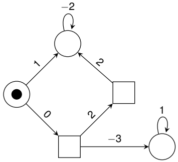
Proof idea

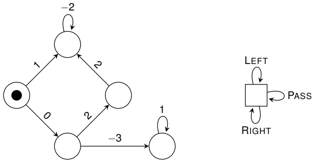
- Given a one-dimensional CRG G.
- wlog $\deg(\Box) \leq 2$.
- Construct a robot game with states where
  - Eve simulates the whole graph.
  - Adam tells which choice he would have made in the original CRG.



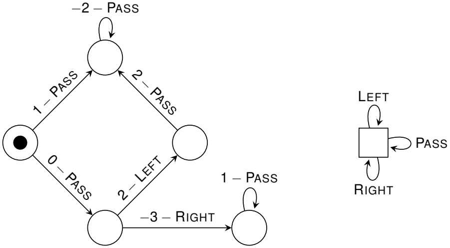
Proof idea

- Given a one-dimensional CRG G.
- wlog $\deg(\Box) \leq 2$.
- Construct a robot game with states where
  - Eve simulates the whole graph.
  - Adam tells which choice he would have made in the original CRG.

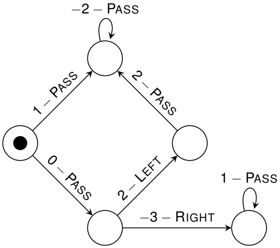


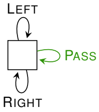


Proof idea

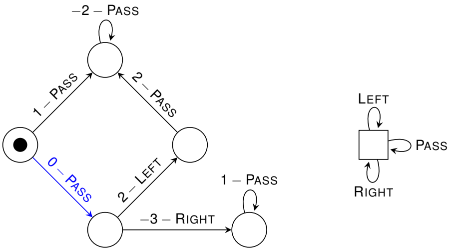


Proof idea

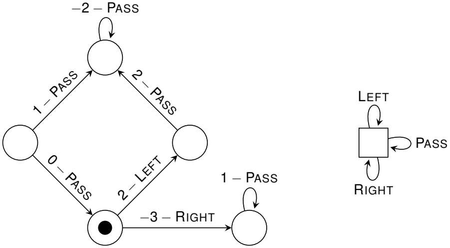




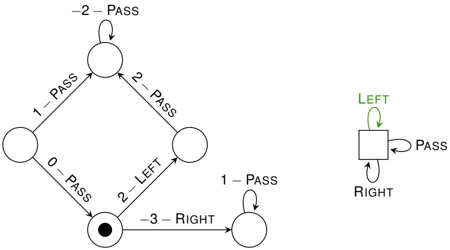
Proof idea



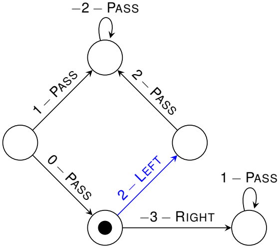
Proof idea

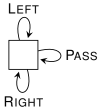


Proof idea

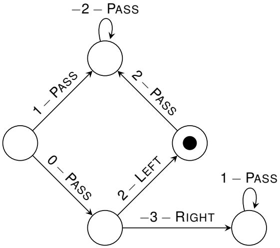


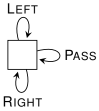
Proof idea



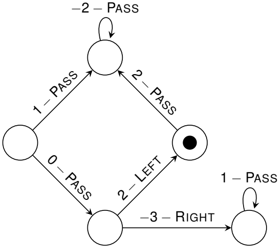


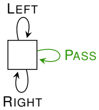
Proof idea



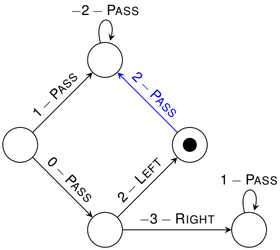


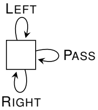
Proof idea



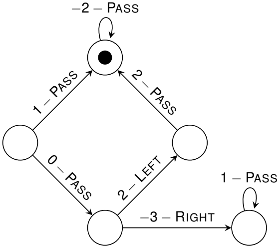


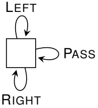
Proof idea



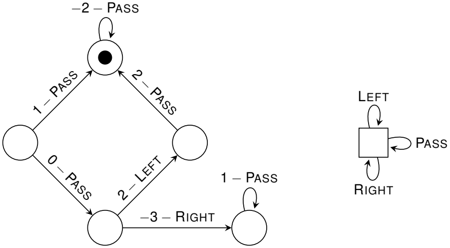


Proof idea





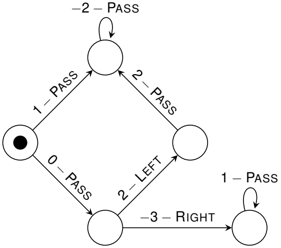
Proof idea

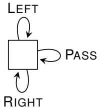


Lemma

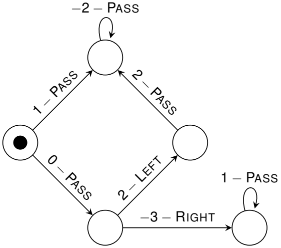
If both players play correctly, then the winner in the RGS is the same as the winner in the CRG.

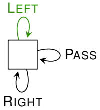
What if they don't play correctly?



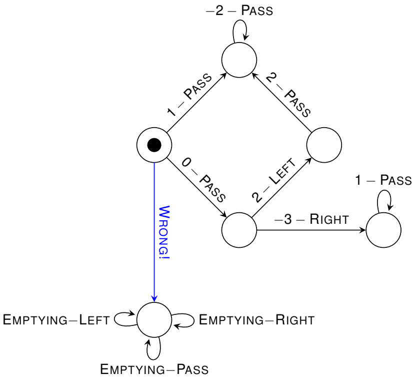


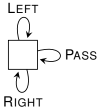
What if they don't play correctly?



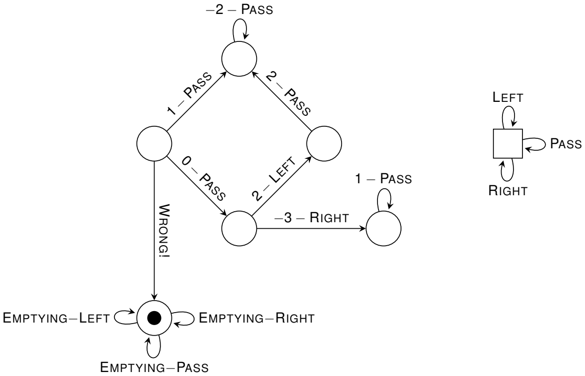


What if they don't play correctly?

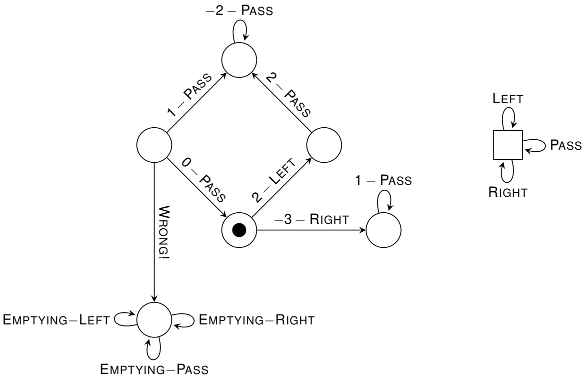




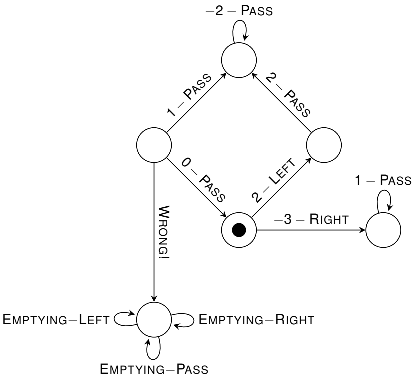
What if they don't play correctly?

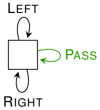


What if they don't play correctly?

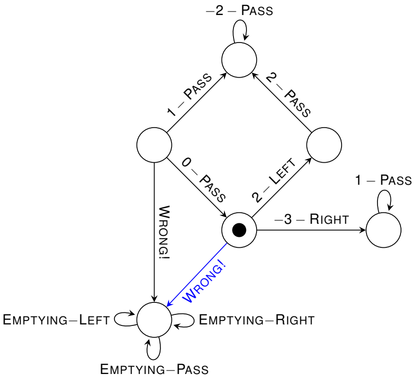


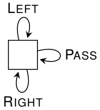
What if they don't play correctly?



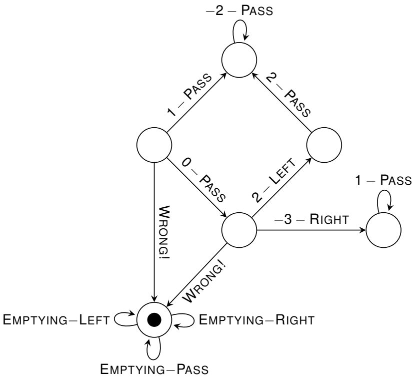


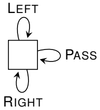
What if they don't play correctly?



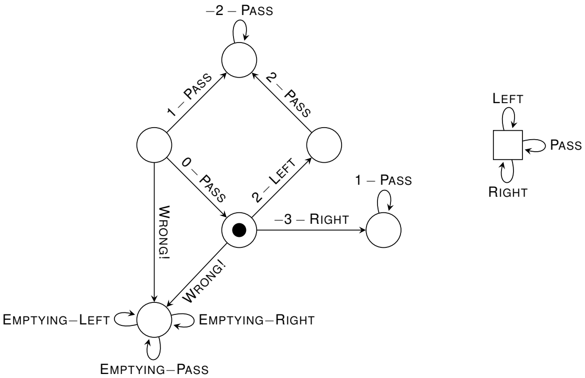


What if they don't play correctly?

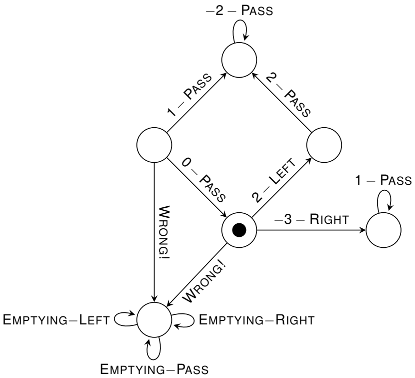


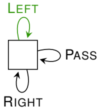


What if they don't play correctly?

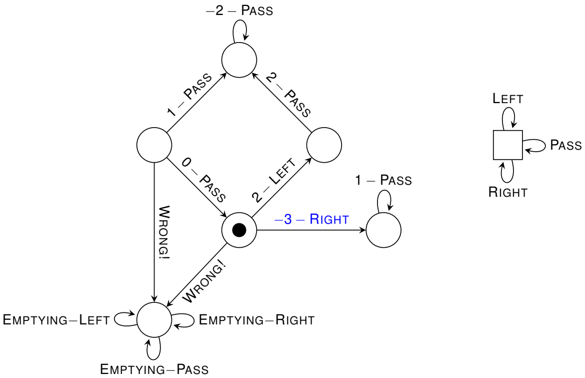


What if they don't play correctly?

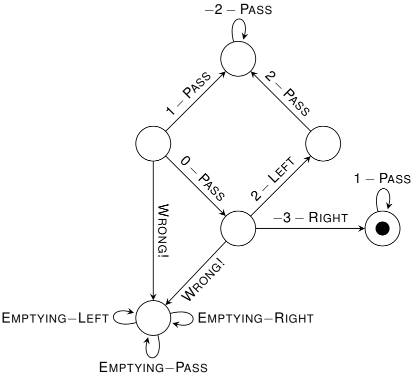


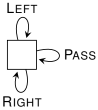


What if they don't play correctly?

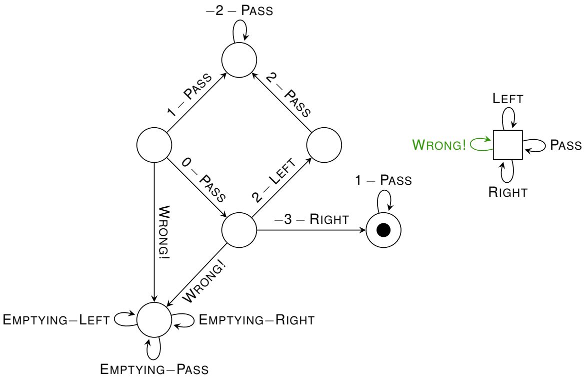


What if they don't play correctly?

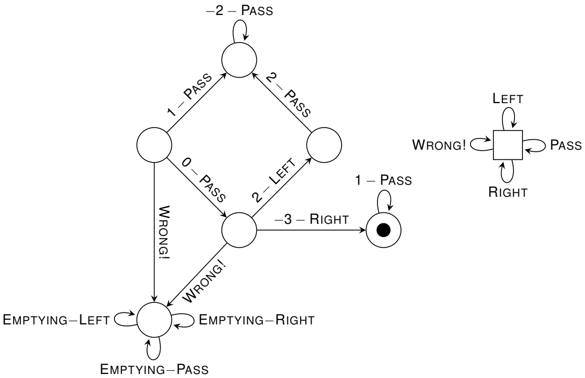




What if they don't play correctly?



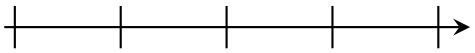
What if they don't play correctly?



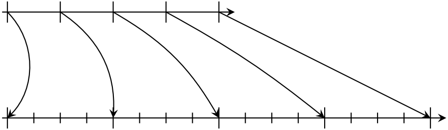
Lemma

If one of the players plays incorrectly, then the opponent is the winner.

What are PASS, LEFT, RIGHT, etc?

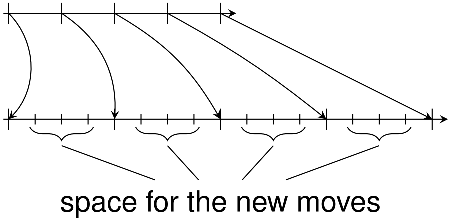


What are PASS, LEFT, RIGHT, etc?



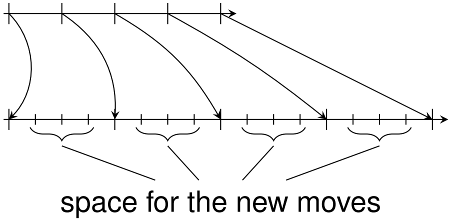
- The original values are multiplied by 4.

What are PASS, LEFT, RIGHT, etc?



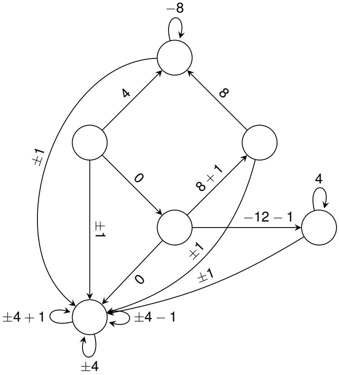
- The original values are multiplied by 4.

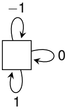
What are PASS, LEFT, RIGHT, etc?



- The original values are multiplied by 4.
- LEFT, RIGHT and WRONG! modify the value mod 4.
- PASS is 0.

The result





Theorem

Whether Eve has a winning strategy in a one-dimensional robot game with states is ***EXPSPACE****-complete.*

Flat robot games with states

Effect of states

Adam \ Eve	stateless	states
stateless	**EXPTIME**-c	**EXPSPACE**-c
states	?	**EXPSPACE**-c

Effect of states

Adam \ Eve	stateless	states
stateless	**EXPTIME**-c	**EXPSPACE**-c
states	?	**EXPSPACE**-c

What if Adam's states are flat?

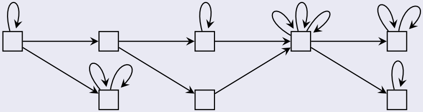
Effect of states

Adam \ Eve	stateless	states
stateless	**EXPTIME**-c	**EXPSPACE**-c
states	?	**EXPSPACE**-c

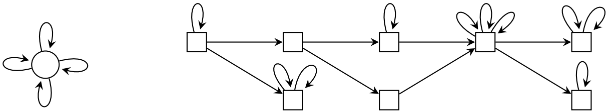
What if Adam's states are flat?

Flat automata

The underlying graph is directed acyclic graph with self-loop.



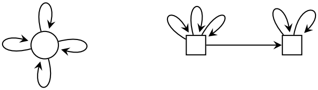
Flat robot games with states



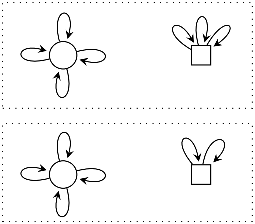
Theorem

Deciding who has a winning strategy in a one-dimensional flat robot game with states is ***EXPTIME****-complete.*

Proof idea

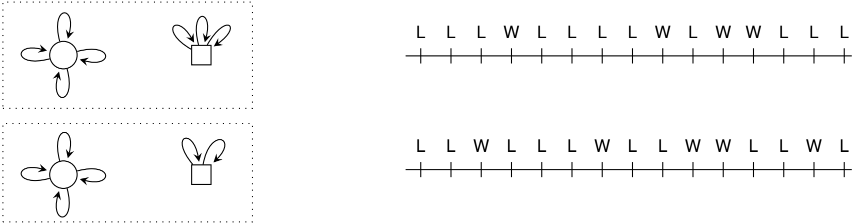


Proof idea



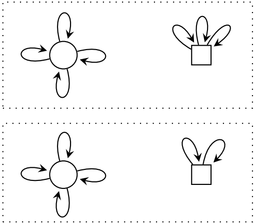
- As there exists a linear order, we can consider several stateless games that are connected.

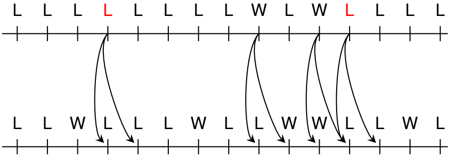
Proof idea



- As there exists a linear order, we can consider several stateless games that are connected.
- The algorithm for robot games gives a description of the winning values for Eve.

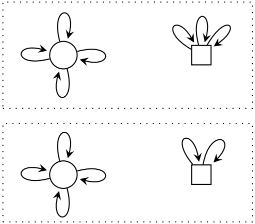
Proof idea

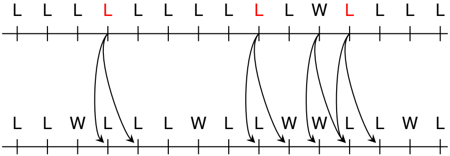




- As there exists a linear order, we can consider several stateless games that are connected.
- The algorithm for robot games gives a description of the winning values for Eve.
- Check which winning values of the first game become losing when moving to the second game.

Proof idea





- As there exists a linear order, we can consider several stateless games that are connected.
- The algorithm for robot games gives a description of the winning values for Eve.
- Check which winning values of the first game become losing when moving to the second game.
- Check how new losing values affect other winning values.

Which values become losing?

Check which winning values of the first game become losing when moving to the second game.

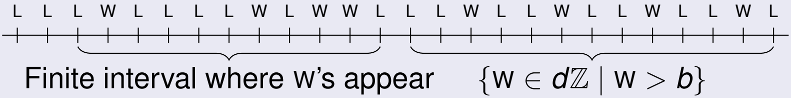
Check how new losing values affect other winning values.

Which values become losing?

Check which winning values of the first game become losing when moving to the second game.

Check how new losing values affect other winning values.

Both are doable in **PTIME** because the winning sets in robot games are (essentially)



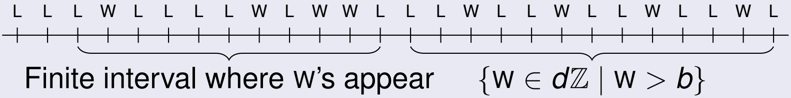
Which values become losing?

Check which winning values of the first game become losing when moving to the second game.

Simple case analysis.

Check how new losing values affect other winning values.

Both are doable in **PTIME** because the winning sets in robot games are (essentially)



Which values become losing?

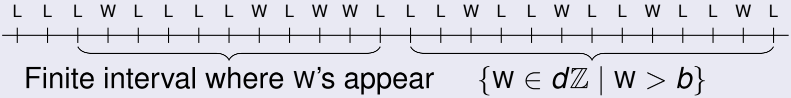
Check which winning values of the first game become losing when moving to the second game.

Simple case analysis.

Check how new losing values affect other winning values.

We can construct an equivalent game on a graph where winning values are computed using the attractor construction.

Both are doable in **PTIME** because the winning sets in robot games are (essentially)



Conclusion

Summary

Theorem

Given one-dimensional robot games with states. Deciding which player has a winning strategy is ***EXPSPACE****-complete.*

Game	Dimension 1	Dimension ≥ 2
counter reachability games	**EXPSPACE**-complete	U
robot games with states	?	U
flat robot games with states	?	?
robot games	**EXPTIME**-complete	U

Summary

Theorem

Given one-dimensional robot games with states. Deciding which player has a winning strategy is ***EXPSPACE****-complete.*

Game	Dimension	
	1	≥ 2
counter reachability games	**EXPSPACE**-complete	U
robot games with states	**EXPSPACE**-complete	U
flat robot games with states	**EXPTIME**-complete	?
robot games	**EXPTIME**-complete	U

Future work

- Investigate further what kind of state structure increases the complexity.
- Decidability of stateless VASS games.

Thank you for your attention!