# A take away game emulating the rule 110 cellular automaton

#### Urban Larsson, Chalmers University of Technology, Goteborg, Sweden INTEGERS 2011

November 23, 2011

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(日) (四) (日) (日) (日)

## Table of contents

Introduction

Cellular automata and Pascal's triangle

A take away game of rule 60

A take away game of rule 110

Undecidability of rule 110 CA

(D) (A) (A) (A)

# Emulate one dimensional cellular automata by normal play impartial games of take away

 One dimensional cellular automata generate two dimensional patterns,

Urban Larsson, Chalmers University of Technology, Goteborg A take away game emulating the rule 110 cellular automator

(D) (A) (A) (A) (A)

Emulate one dimensional cellular automata by normal play impartial games of take away

- One dimensional cellular automata generate two dimensional patterns,
- therefore we study take away games on two heaps.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

# Emulate one dimensional cellular automata by normal play impartial games of take away

- One dimensional cellular automata generate two dimensional patterns,
- therefore we study take away games on two heaps.
- The winning strategies emulate Wolfram's rule 60 and 110 cellular automata,

# Emulate one dimensional cellular automata by normal play impartial games of take away

- One dimensional cellular automata generate two dimensional patterns,
- therefore we study take away games on two heaps.
- The winning strategies emulate Wolfram's rule 60 and 110 cellular automata,
- ► and thereby, by recent results of Matthew Cook (2004) for rule 110,

# Emulate one dimensional cellular automata by normal play impartial games of take away

- One dimensional cellular automata generate two dimensional patterns,
- therefore we study take away games on two heaps.
- The winning strategies emulate Wolfram's rule 60 and 110 cellular automata,
- and thereby, by recent results of Matthew Cook (2004) for rule 110,
- ► a universal Turing Machine.

A note on a recent similar result

 CANT 2011 presentation, a "take away" game of finitely many moves on several heaps (joint work with Johan Wästlund):

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(日) (四) (日) (日) (日)

#### A note on a recent similar result

- CANT 2011 presentation, a "take away" game of finitely many moves on several heaps (joint work with Johan Wästlund):
- it is undecidable whether a given finite pattern occurs as winning positions for the previous player.

(D) (A) (A)

### A note on a recent similar result

- CANT 2011 presentation, a "take away" game of finitely many moves on several heaps (joint work with Johan Wästlund):
- it is undecidable whether a given finite pattern occurs as winning positions for the previous player.
- ► We have extended this result:

(D) (A) (A)

## A note on a recent similar result

- CANT 2011 presentation, a "take away" game of finitely many moves on several heaps (joint work with Johan Wästlund):
- it is undecidable whether a given finite pattern occurs as winning positions for the previous player.
- ▶ We have extended this result:
- it is undecidable whether two games, each with a finite number of moves, have identical winning strategies.

### Cellular automata

 Cellular automata give rise to interesting 2-dimensional patterns.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

・ロン ・四と ・ヨン ・ヨン

## Cellular automata

- Cellular automata give rise to interesting 2-dimensional patterns.
- Two states (0 and 1), and the state of cell i at time t is denoted by a<sub>i</sub><sup>t</sup>.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

イロト 不得下 イヨト イヨト

## Cellular automata

- Cellular automata give rise to interesting 2-dimensional patterns.
- Two states (0 and 1), and the state of cell i at time t is denoted by a<sup>t</sup><sub>i</sub>.
- Let starting configuration be ...000111....

Urban Larsson, Chalmers University of Technology, Goteborg A take away game emulating the rule 110 cellular automator

#### The rule 60 cellular automaton

► The update rule,

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

## The rule 60 cellular automaton

The update rule,

• the Boolean function  $f(x, y) = x \oplus y$ ,

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

## The rule 60 cellular automaton

- The update rule,
- the Boolean function  $f(x, y) = x \oplus y$ ,
- f(x,y) = 0 iff x = y.

### The rule 60 cellular automaton

- The update rule,
- the Boolean function  $f(x, y) = x \oplus y$ ,
- f(x, y) = 0 iff x = y.
- The states are updated by  $a_i^{t+1} = f(a_{i-1}^t, a_i^t)$ ,

(日) (四) (日) (日) (日)

## The rule 60 cellular automaton

- The update rule,
- the Boolean function  $f(x, y) = x \oplus y$ ,
- f(x, y) = 0 iff x = y.
- The states are updated by  $a_i^{t+1} = f(a_{i-1}^t, a_i^t)$ ,
- ▶ The 1s are represented by red squares.

## The rule 60 cellular automaton

- The update rule,
- the Boolean function  $f(x, y) = x \oplus y$ ,
- f(x, y) = 0 iff x = y.
- The states are updated by  $a_i^{t+1} = f(a_{i-1}^t, a_i^t)$ ,
- The 1s are represented by red squares.
- The leftmost column represents  $a_1^t$ .

#### Example



Figure: The CA given by  $f(x, y) = x \oplus y$  (Wolfram's rule 60).

#### Example



Figure: The CA given by  $f(x, y) = x \oplus y$  (Wolfram's rule 60).

#### Example



Figure: The CA given by  $f(x, y) = x \oplus y$  (Wolfram's rule 60).

#### Example



Figure: The CA given by  $f(x, y) = x \oplus y$  (Wolfram's rule 60).

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

#### Example



Figure: The CA given by  $f(x, y) = x \oplus y$  (Wolfram's rule 60).

#### Example



Figure: The CA given by  $f(x, y) = x \oplus y$  (Wolfram's rule 60).

・ロト ・ 同ト ・ ヨト ・ ヨト

## The pattern is Pascal's triangle modulo 2



Figure: The CA given by  $f(x, y) = x \oplus y$  (Wolfram's rule 60).

-

## A take away game of rule 60

► A take away game emulating the behavior of rule 60.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(D) (A) (A) (A)

## A take away game of rule 60

- ► A take away game emulating the behavior of rule 60.
- ► A heap of matches and a heap of tokens.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(D) (A) (A) (A)

# A take away game of rule 60

- ► A take away game emulating the behavior of rule 60.
- A heap of matches and a heap of tokens.
- ► 2 players alternate in moving.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(日) (四) (日) (日) (日)

# A take away game of rule 60

- ► A take away game emulating the behavior of rule 60.
- A heap of matches and a heap of tokens.
- 2 players alternate in moving.
- A player can remove any number of matches, at least one and at most the whole heap,

# A take away game of rule 60

- ► A take away game emulating the behavior of rule 60.
- A heap of matches and a heap of tokens.
- 2 players alternate in moving.
- A player can remove any number of matches, at least one and at most the whole heap,
- ► and a number of tokens 0 ≤ t ≤ m<sub>p</sub> limited by the number of matches m<sub>p</sub> the previous player removed.

A take away game of rule 60

Ending condition: who wins?

Urban Larsson, Chalmers University of Technology, Goteborg A take away game emulating the rule 110 cellular automator

(D) (A) (A) (A)

## A take away game of rule 60

- Ending condition: who wins?
- ► Normal play, last move wins (a player unable to move loses).

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(日) (四) (日) (日) (日)

# A take away game of rule 60

- Ending condition: who wins?
- Normal play, last move wins (a player unable to move loses).
- Matches simulate "time", hence removal of final match wins.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

# A take away game of rule 60

- Ending condition: who wins?
- Normal play, last move wins (a player unable to move loses).
- Matches simulate "time", hence removal of final match wins.
- Exception, the final match can be removed if and only if the heap of tokens is emptied in the same move.
# A take away game of rule 60

- Ending condition: who wins?
- Normal play, last move wins (a player unable to move loses).
- Matches simulate "time", hence removal of final match wins.
- Exception, the final match can be removed if and only if the heap of tokens is emptied in the same move.
- Note  $t \leq m_p$ , so if there are less than  $m_p$  tokens left,

# A take away game of rule 60

- Ending condition: who wins?
- Normal play, last move wins (a player unable to move loses).
- Matches simulate "time", hence removal of final match wins.
- Exception, the final match can be removed if and only if the heap of tokens is emptied in the same move.
- ▶ Note  $t \le m_p$ , so if there are less than  $m_p$  tokens left,
- a winning move is to remove all tokens and all matches.

#### A terminal player 2 winning position

Player 2 removed the rightmost previous-match. Hence  $0 \le t \le 1$ . Player 1 cannot remove both tokens; neither the final match.



Urban Larsson, Chalmers University of Technology, Goteborg A take away game emulating the rule 110 cellular automator

#### A player 1 winning position

Player 2 removed the rightmost two previous-matches. Hence  $0 \le t \le 2$ . Both tokens can be removed and the final match.



Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

#### A player 1 winning move

Player 2 removed the rightmost two previous-matches. Hence  $0 \le t \le 2$ . Both tokens can be removed and the final match.



Urban Larsson, Chalmers University of Technology, Goteborg A take away game emulating the rule 110 cellular automator

・ 同 ト ・ ヨ ト ・ ヨ ト

# A more advanced position, (4, 5, 3)

In general we denote a position by  $(\#tape, \#time, m_p)$ , where #tape denotes the number of tokens and #time the number of matches. From (4, 5, 3), at most 3 tokens can be removed. Player 1 cannot win by removing all matches but one, because then player 2 removes all tokens together with the final match.



Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

# A more advanced position, (4, 5, 3)

In general we denote a position by  $(\#tape, \#time, m_p)$ , where #tape denotes the number of tokens and #time the number of matches. From (4, 5, 3), at most 3 tokens can be removed. Player 1 cannot win by removing all matches but one, because then player 2 removes all tokens together with the final match.



Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

・ 同 ト ・ ヨ ト ・ ヨ ト

# A more advanced position, (4, 5, 3)

In general we denote a position by  $(\#tape, \#time, m_p)$ , where #tape denotes the number of tokens and #time the number of matches. From (4, 5, 3), at most 3 tokens can be removed. Player 1 cannot win by removing all matches but one, because then player 2 removes all tokens together with the final match.



Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

# The rule 60 CA gives the winning strategy

In fact, by the following result, there is no winning move for the first player from (4, 5, 3).

#### Theorem

A position (x, y, z) is a second player win if and only if the rule 60 CA satisfies  $a_x^y = \ldots = a_x^{y+z-1} = 0$  and  $a_x^{y-1} = 1$  if y > 0, with initial condition  $a_x^0 = 1$  iff x > 0.

That is, the winning strategy is given by Pascal's triangle modulo 2. We indicate the winning pattern as follows.

#### The rule 60 game's position (4, 5, 3)



Figure: The green pattern indicates a second player winning position  $\#tape = 4, \#time = 5, m_p = 3.$ 

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(D) (A) (A) (A)

# CA, Rule 110

The CA given by Stephen Wolfram's rule 110 with initial string a central (finite) data together with left and right periodic patterns was proved undecidable by Matthew Cook around year 2000.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

・ロト ・回ト ・ヨト ・ヨト

# CA, Rule 110

- The CA given by Stephen Wolfram's rule 110 with initial string a central (finite) data together with left and right periodic patterns was proved undecidable by Matthew Cook around year 2000.
- Update rule, a boolean function with a three cell input  $a_i^{t+1} = f(a_{i-1}^t, a_i^t, a_{i+1}^t)$ :

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

# CA, Rule 110

- The CA given by Stephen Wolfram's rule 110 with initial string a central (finite) data together with left and right periodic patterns was proved undecidable by Matthew Cook around year 2000.
- Update rule, a boolean function with a three cell input  $a_i^{t+1} = f(a_{i-1}^t, a_i^t, a_{i+1}^t)$ :
- f(x, y, z) = 0 iff x = y = z = 1 or y = z = 0.

# CA, Rule 110

- The CA given by Stephen Wolfram's rule 110 with initial string a central (finite) data together with left and right periodic patterns was proved undecidable by Matthew Cook around year 2000.
- Update rule, a boolean function with a three cell input  $a_i^{t+1} = f(a_{i-1}^t, a_i^t, a_{i+1}^t)$ :
- f(x, y, z) = 0 iff x = y = z = 1 or y = z = 0.
- ► The CA pattern is more complex.

CA, rule 110, time moves upwards, initial condition a single "1". A systems of interacting "gliders".



Urban Larsson, Chalmers University of Technology, Goteborg A take away game emulating the rule 110 cellular automator

#### New rules of game

Requirements for the rule 110 game:

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(D) (A) (A) (A)

#### New rules of game

- Requirements for the rule 110 game:
- winning strategy should emulate the rule 110 CA,

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator



- Requirements for the rule 110 game:
- winning strategy should emulate the rule 110 CA,
- and thereby an universal Turing Machine.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

#### New rules of game

► Method:

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

イロン イヨン イヨン イヨン

臣

### New rules of game

- Method:
- generalize the rules of the rule 60 game,

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(日) (同) (三) (三)

## New rules of game

- Method:
- generalize the rules of the rule 60 game,
- develop an appropriate ending condition,

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(D) (A) (A) (A)

# New rules of game

- Method:
- generalize the rules of the rule 60 game,
- develop an appropriate ending condition,
- corresponding to an arbitrary initial string of rule 110.

#### A take away game of rule 110

A heap of matches and a heap of "colored" tokens, each token being either black or white.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

#### A take away game of rule 110

- A heap of matches and a heap of "colored" tokens, each token being either black or white.
- 2 players alternate in moving.

## A take away game of rule 110

- A heap of matches and a heap of "colored" tokens, each token being either black or white.
- 2 players alternate in moving.
- A player can remove any number of matches, at least one and at most the whole heap,

# A take away game of rule 110

- A heap of matches and a heap of "colored" tokens, each token being either black or white.
- 2 players alternate in moving.
- A player can remove any number of matches, at least one and at most the whole heap,
- ► and a number t of tokens 0 ≤ m − 1 ≤ t ≤ m + m<sub>p</sub> bounded by the number of matches removed in this move m, and by the other player in the previous move, m<sub>p</sub>.

# A take away game of rule 110

- A heap of matches and a heap of "colored" tokens, each token being either black or white.
- 2 players alternate in moving.
- A player can remove any number of matches, at least one and at most the whole heap,
- ▶ and a number t of tokens 0 ≤ m − 1 ≤ t ≤ m + m<sub>p</sub> bounded by the number of matches removed in this move m, and by the other player in the previous move, m<sub>p</sub>.
- If  $\#time \ge \#tape$ , both heaps can be removed.

#### A take away game of rule 110

Ending condition:

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

#### A take away game of rule 110

Ending condition:

► Normal play, last move wins.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(D) (A) (A) (A)

# A take away game of rule 110

- Ending condition:
- Normal play, last move wins.
- ► Matches simulate "time", hence removal of final match wins.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

# A take away game of rule 110

- Ending condition:
- Normal play, last move wins.
- ► Matches simulate "time", hence removal of final match wins.
- Exception, the final match cannot be removed if there is a black token among the top *m* tokens.

イロト イポト イヨト イヨト

# A take away game of rule 110

- Ending condition:
- Normal play, last move wins.
- ► Matches simulate "time", hence removal of final match wins.
- Exception, the final match cannot be removed if there is a black token among the top *m* tokens.
- Note  $t \le m + m_p$ , so if there are less than  $m + m_p$  tokens left,

イロト イポト イヨト イヨト

# A take away game of rule 110

- Ending condition:
- Normal play, last move wins.
- ► Matches simulate "time", hence removal of final match wins.
- Exception, the final match cannot be removed if there is a black token among the top *m* tokens.
- ▶ Note  $t \le m + m_p$ , so if there are less than  $m + m_p$  tokens left,
- a winning move is to remove all tokens and all matches.

#### Rule 110 game

A position of the rule 110 game. (Player 2 wins.)



Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

#### The rule 110 CA gives the winning strategy

Let  $A = \ldots a_0^0 a_1^0 \ldots$  be a doubly infinite binary string corresponding to the initial condition of a rule 110 CA. Let

$$S = S(\xi, \#tape, A) = a_{\xi}^0 \dots a_{\xi+\#tape-1}^0$$

encode a finite binary string (empty if #tape = 0) corresponding to an ordered heap of black (1s) and white (0s) tokens. A position of a rule 110 *A*-game is denoted by  $(S, \#tape, \#time, m_p)$ .

#### The rule 110 CA gives the winning strategy

Let  $A = \ldots a_0^0 a_1^0 \ldots$  be a doubly infinite binary string corresponding to the initial condition of a rule 110 CA. Let

$$S = S(\xi, \#tape, A) = a_{\xi}^0 \dots a_{\xi+\#tape-1}^0$$

encode a finite binary string (empty if #tape = 0) corresponding to an ordered heap of black (1s) and white (0s) tokens. A position of a rule 110 *A*-game is denoted by  $(S, \#tape, \#time, m_p)$ . For simplicity set  $\xi = 0$ .
## The rule 110 CA gives the winning strategy

The x-coordinate of the CA corresponds to #tape - #time; the y-coordinate to #time. Suppose that  $x \ge y + m_p \ge 1$ . (Otherwise encode S from  $A' = \dots 00a_1^0 a_2^0 \dots$ ).

#### Theorem

A position of a rule 110 A-game  $(S, x + y, y, m_p)$  is a previous player win if and only if the rule 110 CA which takes A as input satisfies  $a_x^y = \ldots = a_x^{y+m_p-1} = 0$  and  $a_x^{y-1} = 1$  if y > 0.

(D) (A) (A) (A) (A)

## The rule 110 CA gives the winning strategy

The x-coordinate of the CA corresponds to #tape - #time; the y-coordinate to #time. Suppose that  $x \ge y + m_p \ge 1$ . (Otherwise encode S from  $A' = \dots 00a_1^0a_2^0\dots$ ).

#### Theorem

A position of a rule 110 A-game  $(S, x + y, y, m_p)$  is a previous player win if and only if the rule 110 CA which takes A as input satisfies  $a_x^y = \ldots = a_x^{y+m_p-1} = 0$  and  $a_x^{y-1} = 1$  if y > 0.

Thus, the statement for rule 110 is in analogy to that of rule 60, the difference is the ending condition (which can be implemented for rule 60 in analogy). That is, the winning strategy is given by the updates of the rule 110 CA.

(日) (周) (王) (王) (王)

## Updates of rule 110 CA together with game positions



Figure: Some rule 110 game positions  $(m_p \text{ omitted})$  for the finite ending condition S = 11010011101100 together with CA updates. Example: green position  $(110100, 6, 2, m_p)$  second player wins if  $1 \le m_p \le 3$ . For the other positions the first player wins independent of  $m_p$ .

(ロ) (同) (三) (三)

## Undecidability of rule 110 CA

Let  $\underline{LCR}$  denote a doubly infinite binary string with periodic Left and Right patterns and a Central data pattern. Matthew Cook programmed an universal Turing machine to halt iff a given finite bit-string (namely 01101001101000 obtained from the so-called F glider) occurs in the update of an  $\underline{LCR}$  rule 110 CA, where C is interpreted as the finite data input to the program.

(日) (部) (E) (E) (E)

## Undecidability of rule 110 CA

Let  $\underline{LCR}$  denote a doubly infinite binary string with periodic Left and Right patterns and a Central data pattern. Matthew Cook programmed an universal Turing machine to halt iff a given finite bit-string (namely 01101001101000 obtained from the so-called F glider) occurs in the update of an  $\underline{LCR}$  rule 110 CA, where C is interpreted as the finite data input to the program.

#### Theorem (Cook04)

It is algorithmically undecidable whether a given finite binary string occurs in the rule 110 CA with an  $\underline{LCR}$  initial condition.

イロト イポト イヨト イヨト 二日

# Undecidability of rule 110 CA

Let  $\underline{LCR}$  denote a doubly infinite binary string with periodic Left and Right patterns and a Central data pattern. Matthew Cook programmed an universal Turing machine to halt iff a given finite bit-string (namely 01101001101000 obtained from the so-called F glider) occurs in the update of an  $\underline{LCR}$  rule 110 CA, where C is interpreted as the finite data input to the program.

#### Theorem (Cook04)

It is algorithmically undecidable whether a given finite binary string occurs in the rule 110 CA with an  $\underline{LCR}$  initial condition.

#### Corollary (Cook04)

It is undecidable whether the central data pattern will ultimately become periodic.

Urban Larsson, Chalmers University of Technology, Goteborg A take away game emulating the rule 110 cellular automator

# Undecidability of rule 110 CA

Therefore we can program an universal Turing machine to halt iff a finite consecutive sequence of moves is optimal.

#### Theorem

It is algorithmically undecidable whether a given finite sequence of moves has alternating previous player winning positions for  $\underline{LCR}$ -rule 110 CA games.

(日) (四) (日) (日) (日)

## Undecidable F glider occurrence in LCR rule 110 CA



Tape

Figure: An F glider embedded in Rule 110 ether.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(4月) (日) (日)

### An undecidable path of optimal moves



Tape

(日) (四) (日) (日) (日)

Figure: An optimal sequence of consecutive moves traversing an F glider.

Urban Larsson, Chalmers University of Technology, Goteborg A take away game emulating the rule 110 cellular automator

## Periodicity of P-positions and the rule 110 game

#### Theorem

It is undecidable whether the central pattern of second player winning positions will ultimately become periodic for  $\underline{LCR}$ -rule 110 games.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

(日) (四) (日) (日) (日)

### Convergence of winning strategies

We say that the  $A_1$ - and  $A_2$ -rule 110 games converge if their (previous player) winning strategies are identical for all sufficiently large heap-sizes. Let  $a_x^y(Z)$  denote the bit in cell (x, y) given the initial condition Z of the rule 110 CA.

Urban Larsson, Chalmers University of Technology, Gotebore A take away game emulating the rule 110 cellular automator

イロト イポト イヨト イヨト

## Convergence of winning strategies

We say that the  $A_1$ - and  $A_2$ -rule 110 games converge if their (previous player) winning strategies are identical for all sufficiently large heap-sizes. Let  $a_x^y(Z)$  denote the bit in cell (x, y) given the initial condition Z of the rule 110 CA.

#### Lemma

The winning strategies of the  $A_1$  and  $A_2$  rule 110 games converge iff there are  $x_0$  and  $y_0$  such that  $a_x^y(A_1) = a_x^y(A_2)$ , for all heaps with  $x \ge x_0$  tokens and  $y \ge y_0$  matches.

## Convergence of winning strategies

We say that the  $A_1$ - and  $A_2$ -rule 110 games converge if their (previous player) winning strategies are identical for all sufficiently large heap-sizes. Let  $a_x^y(Z)$  denote the bit in cell (x, y) given the initial condition Z of the rule 110 CA.

#### Lemma

The winning strategies of the  $A_1$  and  $A_2$  rule 110 games converge iff there are  $x_0$  and  $y_0$  such that  $a_x^y(A_1) = a_x^y(A_2)$ , for all heaps with  $x \ge x_0$  tokens and  $y \ge y_0$  matches.

That is, the CAs converge iff the winning strategies of the corresponding games converge.

<ロ> (四) (四) (注) (注) (注) (注)

# Decidability and convergence of winning strategies

### Corollary

It is decidable whether the winning strategies of the rule 110 games  $\underline{L}C'\underline{R}$  and  $\underline{L}C\underline{R}$  converge if and only if it is decidable whether  $a_x^y(\underline{L}C'\underline{R}) = a_x^y(\underline{L}C\underline{R})$ , for all sufficiently large x and y.

(ロ) (同) (三) (三)

# Decidability and convergence of winning strategies

### Corollary

It is decidable whether the winning strategies of the rule 110 games  $\underline{L}C'\underline{R}$  and  $\underline{L}C\underline{R}$  converge if and only if it is decidable whether  $a_x^y(\underline{L}C'\underline{R}) = a_x^y(\underline{L}C\underline{R})$ , for all sufficiently large x and y.

In private communication with Matthew Cook: The latter part is an open question (a week ago).

(D) (A) (A) (A) (A)