ITT8040 — Cellular Automata
Lecture 5

Silvio Capobianco

Institute of Cybernetics at TUT

April 17, 2013

Tilings

A tile set is a triple $\mathcal{T} = (T, N, R)$ where:

- T is a finite set of tiles,
- $N = \{\vec{n}_1, \ldots, \vec{n}_m\} \subseteq \mathbb{Z}^2$ is a neighborhood, and
- $R \subseteq T^m$ is an m-ary matching rule.

A tiling on $\mathcal{T}$ is a map $t : \mathbb{Z}^2 \to T$.
A tiling t is valid at $\vec{n} \in \mathbb{Z}^2$ if and only if

$$(t(\vec{n} + \vec{n}_1), \ldots, t(\vec{n} + \vec{n}_m)) \in R$$

A tiling is valid if it is valid at every $\vec{n} \in \mathbb{Z}^2$.
Call $V(\mathcal{T})$ the set of valid tilings for the tile set $\mathcal{T}$.

Similarities with configurations and CA

Tilings behave as configurations whose states are tiles.
In turn, matching rules behave like "static CA with boolean values".

- If t is valid at $\vec{n}$ and $\tau = \tau_{\vec{r}}$ is a translation, then $\tau \circ t$ is valid at $\vec{n}' = \vec{n} - \vec{r}$.
  In particular: translations of valid tilings are valid.
- The limit of a sequence of valid tilings is valid.

By compactness, we also get:

if $\mathcal{T}$ admits, for every finite $D \subseteq \mathbb{Z}^2$,
a tiling that is valid at every $\vec{n} \in D$,
then $\mathcal{T}$ admits a valid tiling

Wang tiles

A Wang tile set has $N = V_1$, the von Neumann neighborhood of radius 1.

- Wang tiles can be represented as squares with colored sides.
- A tiling is valid at $\vec{n}$ if and only if the colors on the sides of the tile match with those of the neighbors.

Periodic tilings

A tiling t is $\vec{r}$-periodic if $t(\vec{n}+\vec{r}) = t(\vec{n})$ for every $\vec{n} \in \mathbb{Z}^2$.
t is totally periodic if it is $\vec{r}_i$-periodic for two linearly independent $\vec{r}_1, \vec{r}_2 \in \mathbb{Z}^2$.

Totally periodic tilings

If $\mathcal{T} = (T, N, R)$ admits an $\vec{r}$-periodic tiling for some $\vec{r} \neq \vec{0}$, then it admits a totally periodic tiling.

- We may suppose $N = M_R$, the Moore neighborhood of radius R. We may also suppose $\vec{r} = (a, b)$ with $b \geq R > 0$.
- Consider a subdivision of the plane in rectangles of width $w = 2(|a| + R)$ and height $h = b$, so that the $(n+1)$-th line is displaced by a from the n-th.
  (Thus, the lower left corners of the two are displaced by $\vec{r}$.)
- There must be a patch A that is repeated along a line—thus, by $\vec{r}$-periodicity, along all lines.
- The full horizontal strip from the first A-patch included to the second A-patch excluded can then be used to construct a valid periodic tiling.

Aperiodic tile sets

A tile set $\mathcal{T} = (T, N, R)$ is aperiodic if

- it admits a valid tiling, but
- it does not admit any valid periodic tiling.

Conjecture / Wishful thought: No aperiodic tile sets exist.

- This would yield decidability of the following problem: Given a tile set, determine if it admits a valid tiling.

Theorem: (Berger, 1966)
There exist aperiodic Wang tile sets.

Is it as ugly as it seems? Let's see ...

Constructions with aperiodic tile sets, I

Let $\mathcal{T} = (T, N, R)$ be an aperiodic Wang tile set.
Define a cellular automaton as follows:

- $S = T \sqcup \{q\}$, q being a new state;
- $(G(c))(\vec{n})$ is $c(\vec{n})$ if c is a valid tiling at $\vec{n}$, and q otherwise.

Then:

- Every periodic configurations becomes quiescent in finite time.
- Every valid tiling is a spatially aperiodic fixed point.
- The only spatially periodic fixed point is the quiescent configuration.

Constructions with aperiodic tile sets, II

Let $\mathcal{T} = (T, N, R)$ be an aperiodic Wang tile set.
Define a cellular automaton as follows:

- $S = T \sqcup \{q, p\}$, q and p being new states;
- $(G(c))(\vec{n})$ is p if c is a valid tiling at $\vec{n}$, and q otherwise.

Then:

- The constant configuration $c(\vec{n}) = p$ for every $\vec{n} \in \mathbb{Z}^2$ only has aperiodic preimages.

In dimension 1, every periodic configuration which is not a Garden-of-Eden has a periodic preimage.

Constructions with aperiodic tile sets, III

Define a cellular automaton as in the previous slide, but let $(G(c))(\vec{n})$ be:

- $c(\vec{n})$ if c is a valid tiling at $\vec{n}$;
- q if c is not a valid tiling at $\vec{n}$ and $c(\vec{n}) \neq q$;
- p if $c(\vec{n}) = q$.

Then:

- Every fixed point is non-periodic.

In dimension 1, if a CA has a fixed point, then it also has a fixed point which is spatially periodic.

NW-deterministic tile sets

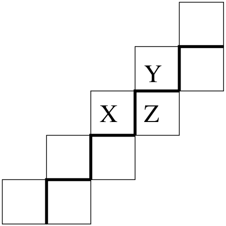
A set T of Wang tiles is NW-deterministic if for any two tiles a and b such that $a \neq b$, either their upper sides have different color, or their left sides have different color.

- That is: tiles are determined by their northwest corner.

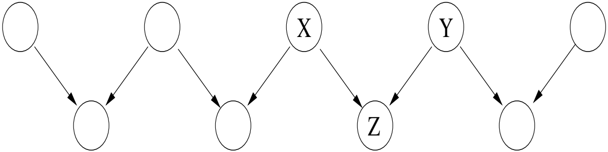
Similarly, there are NE-, SE-, and SW-deterministic sets of Wang tiles.

Most remarkable fact: There are aperiodic tile sets which are deterministic on **all four corners!**

In a valid tiling from a NW-deterministic tile set, every diagonal line in the SW-NE direction is completely determined by the line immediately above:



This situation is similar to that of a radius-1/2 1D CA:



A 1D CA with a single periodic point

Let T be a, aperiodic set of Wang tiles which is both NW- and SE-deterministic.
Let $S = T \sqcup \{q\}$, $N = \{0, 1\}$, and

$$f(x, y) = \begin{cases} z & \text{if } \begin{array}{|c|c|}\hline & y \\ \hline x & z \\ \hline \end{array} \text{ is a match}, \\ q & \text{otherwise}. \end{cases}$$

Let c be a spatially periodic configuration.

- If q never appears in $G^t(c)$ for $t \geq 0$, then a periodic valid tiling would exist.
- But if q appears, then it does in each spatial period and spreads to the left, so $G^T(c) = q$ for some finite time T.

Let now t be a valid tiling.

- Then t defines a bi-infinite, non-periodic orbit where q never appears.

Sets of directed tiles

A directed tile is a tile associated to a follower vector $\vec{f} \in \mathbb{Z}^2$.
A set of directed tiles is a quadruple $\mathcal{D} = (T, N, R, F)$ where $\mathcal{T} = (T, N, R)$ is a tile set and $F : T \to \mathbb{Z}^2$ is a function assigning to each tile its follower vector.
A path in a tiling t over a directed set of tiles (T, N, R, F) is a finite sequence of vectors $\vec{p}_1 \ldots, \vec{p}_k \in \mathbb{Z}^2$ such that

$$\vec{p}_{i+1} = \vec{p}_1 + F(t(\vec{p}_i)) \ \ \forall i = 1, \ldots, k-1 \,,$$

that is, a path in the plane determined by the direction of the tiles.

The plane filling property

A set of directed tiles has the plane filling property if it fulfills the following two properties:

1. It admits a valid tiling of the plane.
2. For every tiling t and one-way infinite path $\{\vec{p}_i\}_{i \geq 1}$ in t such that t is valid at each $\vec{p}_i$, there are arbitrarily large squares of cells such that all cells of the square are on the path.

That is, given a path and a device that runs along the path,

1. either the device finds a tiling error sooner or later,
2. or the device touches each point of squares of arbitrarily large size.

Fact: no periodic tiling can have the plane filling property.

Fact: there exists a set of Wang tiles with the plane filling property.

A non-reversible 2D CA such that G_P is injective

Let (T, N, R, F) be a tile set with the plane filling property.
Define a 2D CA with the von Neumann neighborhood,
$S = T \times \{0, 1\}$, and a local update function defined as follows:

- Call (a, b) the state of the cell at point $\vec{p}$.
- If the tiling is not correct at $\vec{p}$, leave (a, b) unchanged.
- If the tiling is correct at $\vec{p}$, and the value of the cell in the direction $F(t(\vec{p}))$ is (a', b'), update the state to $(a, b \operatorname{xor} b')$.

Then G is not injective, but G_P is.

- For a correct tiling, both the all-0 and the all-1 configurations update to all-0
- However, let c_0 and c_1 be periodic, different at $\vec{p}$, and with same image.
- Then the tiling component is the same, and there is an infinite path along which c_0 and c_1 differ.
- But such path must cover arbitrarily large squares—against periodicity.

Algorithmic questions

An algorithm is

- a mechanical procedure,
- specified by a finite set of instructions,
- to solve some well-defined computational problem.

We will focus on yes-no problems:

- given an object x (instance) and a property P,
- determine whether or not x satisfies P.

Example:

given a d-dimensional cellular automaton,
determine whether or not it is reversible.

The complement of a problem obtained by swapping the "yes" and "no" instances.

Semi-algorithms

A semi-algorithm for a decision problem is a mechanical procedure, specified by a finite set of instructions, such that:

- if the answer is "yes", then the procedure terminates and returns "yes";
- if the answer is "no", then either the procedure terminates and returns "no", or it does not terminate.

For example, the procedure:

- given a cellular automaton (S, d, N, f),
- compose it with every possible cellular automaton (S, d, N', f') until you find one such that the composition is the identity

is a semi-algorithm for CA reversibility.

Decidability

A decision problem is decidable if there exists an algorithm for it.
It is semi-decidable if there exists a semi-algorithm for it.

- Every decidable problem is semi-decidable.
- If a problem is decidable, then so is its complement.
- Most noteworthy:

  if a problem and its complement are both semi-decidable,
  then the problem is decidable

Semi-decidability of a problem tells absolutely nothing about semi-decidability of its complement.

Examples of semi-decidable problems

Reversibility of general CA is semi-decidable.

Surjectivity of general CA has a semi-decidable complement:

- given a cellular automaton (S, d, N, f),
- for all finite $D' \subseteq \mathbb{Z}^d$ and $g' : D' \to S$ do
  - if (D', g') is an orphan then return "non-surjective"

is a semi-algorithm for non-surjectivity.

Encoding

To write a program, is to encode an algorithm over an alphabet.

- We call $\langle A \rangle$ the encoding of the algorithm A in the chosen language, e.g., strings of bits.
- Similarly, we encode instances of the problem which the semi-algorithm is associated to.
- We suppose that such encoding is effective in the sense that there exists an algorithm to decide whether an arbitrary string is the encoding of some algorithm.
- We also suppose that the encoding is natural in the sense that it does not affect the problem, e.g., it does not embed the output of the algorithm in the encoding of the instance.

Semi-algorithm halting is undecidable

Suppose that there exists an algorithm $\texttt{Halt}$ such that:

- given any semi-algorithm A and string w,
- $\texttt{Halt}(\langle A \rangle, w)$ returns True if A halts on w, and False otherwise.

Construct the semi-algorithm $\texttt{Diag}$ as follows:

$$\texttt{Diag}(\langle A \rangle) = \text{if } \texttt{Halt}(\langle A \rangle, \langle A \rangle) \text{ then loop else halt}$$

What is then $\texttt{Halt}(\langle \texttt{Diag} \rangle, \langle \texttt{Diag} \rangle)$?

Semi-algorithm *possible* halting is undecidable

Suppose that there exists an algorithm `SometimesHalt` such that:

- given any semi-algorithm A,
- `SometimesHalt`$(\langle A \rangle)$ returns True if there exists w such that A halts on w, and False otherwise.

Given A and w, construct the semi-algorithm B as follows:

$$\mathsf{B(u)} = \text{if } u \neq w \text{ then loop else A(w)}$$

Then:

- B halts on w if A halts on w.
- B does not halt on any input if A does not halt on w.
- There is an algorithm that constructs $\langle B \rangle$, given $\langle A \rangle$ and w.

What is then `SometimesHalt`$(\langle B \rangle)$?

Turing reduction

Let P and Q be decision problems.
A many-to-one reduction from P to Q is an algorithm S such that

$$\text{for every instance } x \text{ of } P,$$
$$P(x) = \mathrm{yes} \text{ if and only if } Q(S(x)) = \mathrm{yes}$$

Let S be a many-to-one reduction from P to Q.

- If Q is decidable then P is decidable.
- If P is undecidable then Q is undecidable.