2. Natural Computing – Cellular Automata

Slides 1–24 of CA-Kari.pdf

* Turingmachine: tape, state, control unit

transition function δ:

initial state

blank symbol

$(A,0) \longmapsto (B,1,1)$

$(A,1) \longmapsto (B,1,-1)$

$(B,0) \longmapsto (A,1,-1)$

$(B,1) \longmapsto (q_a,1,1)$

↑ accepting state

blank tape: $\ldots 0000\overset{A}{0}000 \ldots$

$\vdash \ldots 00001\overset{B}{0}00 \ldots$

$\vdash \ldots 0000\overset{A}{1}100 \ldots$

$\vdash \ldots 000\overset{B}{0}1100 \ldots$

$\vdash \ldots 00\overset{A}{0}11100 \ldots$

$\vdash \ldots 001\overset{B}{1}1100 \ldots$

$\vdash \ldots 0011\overset{q_a}{1}100$

Every recursively enumerable language is recognized by a Turing machine

Game-of-Life can simulate any Turing machine !!

Rule 110 is also computationally universal !?

Slides 25 – 35

* Cellular Automata – Definitions

Slides 36 – 51

Injective
$$\begin{tikzcd} c_1 \arrow[dr, "G"'] & & c_2 \arrow[dl, "G"] \\ & c & \end{tikzcd} \Rightarrow c_1 = c_2$$

Surjective
$$\begin{tikzcd} \forall\ c \\ \exists\ c' \arrow[u, "G"'] \end{tikzcd}$$

Bijective = Injective + Surjective

Periodic : $\exists_{k,c}\ G^k(c) = c \longrightarrow$ W2

Fixed point $\quad G(c) = c$

Orbit: $\quad c \to G(c) \to G^2(c) \to \cdots$

Nilpotent $\exists_{n,c}\ G^n(S^{\mathbb{Z}^d}) = \{c\} \longrightarrow$ W1

CA { Shift function $\sigma_i : N = \{\underline{e}_i\},\ f = \text{identity}$

Translation $\quad \tau_{\underline{y}} : N = \{-\underline{y}\},\ f = id$

In $\tau_{\underline{y}}(c)$ cell $\underline{x}+\underline{y}$ has state $c(\underline{x})$

$\tau_{\underline{y}} \circ G = G \circ \tau_{\underline{y}}$: All CA commute with translations.

* **Topology**

Introduce a metric (distance measure) between configurations

→ pattern $g : D \to S$

D: finite domain $\subseteq \mathbb{Z}^d$

$$Cyl(g) = \{ c \in S^{\mathbb{Z}^d} : c(\underline{x}) = g(\underline{x}) \text{ for } \underline{x} \in D \}$$

1) A union of open balls
2) Open balls are cylinders

$\Big\}$ cylinders form a basis of the topology

→ can talk about convergence, limits

→ " " " continuous functions

<u>Th1</u>: (G is a CA) <u>iff</u> (G is continuous & G commutes with translations)

<u>Cor1</u>: (G is reversible) <u>iff</u> (G is a bijection)

Slide 61, Slide 70, 71

* **Balance**

Rule 110 $\Big\{$ 5 neighborhood patterns → 1
3 " " → 0

Ex2

e: $\overset{1}{|xxx|}\overset{2}{xxx}|\overset{3}{xxx}|$ ... $\overset{k}{|xxx|}$ ⇝ 3^k choices

c: 0 x x 0 x x 0 x x ... x x 0 ⇝ $(2^2)^{k-1} = 4^{k-1}$ choices

Some choice of c does not have a pre-image → not surjective

$G = (d, S, N, f)$

$$\begin{tikzcd}
\text{pattern } g = f \circ h : D \arrow[r] & S \arrow[d, phantom, ""] \\
\text{pattern } h : N(D) \arrow[u, "G^{(N(D)\to D)}"] \arrow[r] & S
\end{tikzcd}$$

add all neighborhoods

$g(x) = f(h(x+x_1), \dots, h(x+x_m))$ where $N = \{x_1, \dots, x_m\}$

g Orphan: $\neg \exists_{\text{pattern } h}\ G^{(N(D)\to D)}(h) = g$

A configuration c that contains an orphan pattern is Garden-of-Eden ($\neg\exists$ preimage of c)

<u>Prop 2</u> CA has orphan <u>iff</u> CA is non-surjective

Surjective CA $\Rightarrow \forall_{\text{patterns } g \text{ of } D}\ \#\{h : G^{(N(D)\to D)}(h) = g\} = \text{const}$

(D = singleton) $\Rightarrow$ Each state in the local rule table appears equal # times

<u>Garden-of-Eden</u>

$\underline{q}$ is stable if $f(\underline{q}, \dots, \underline{q}) = \underline{q}$ (in figures $\underline{q}$ corresponds to the blank cell)

$c \in S^{\mathbb{Z}^d}$ is $\underline{q}$-finite <u>iff</u> $\{\underline{x} \in \mathbb{Z}^d : c(\underline{x}) \neq \underline{q}\}$ is finite

$\mathcal{F}_q \subseteq S^{\mathbb{Z}^d}$ is the subset of q-finite configurations

If $c \in \mathcal{F}_q$ then $G(c) \in \mathcal{F}_q \Rightarrow$ Define $G_F : \mathcal{F}_q \to \mathcal{F}_q$

In other words: G_F is the global update rule for ~~cellular~~ configurations with a finite # non-blank cells.

We call G preinjective <u>iff</u> G_F is injective

Clearly, injective $\Rightarrow$ pre-injective

<u>Rule 110</u> is not surjective: e.g., 01010 is an orphan

Ex4

C : $\cdots 000\,xxx \cdots\cdots xxx\,00 \cdots$ $\Rightarrow$ $2^{5k-2} = 32^k/4$ choices

$\downarrow$

$G(c)$ $00\,|\underbrace{*****}_{1}|\underbrace{*****}_{2}|\cdots|\underbrace{*****}_{k}|$ $\Rightarrow$ at most $(2^5-1)^k = 31^k$ choices

orphan cannot occur

There must be 2 0-finite configurations with the same image

<u>Th 3</u>: (G is pre-injective) $\Rightarrow$ (G is surjective)

<u>Cor</u> injective $\equiv$ bijective $\equiv$ reversible !!

Also <u>Th 4</u> (G is surjective) $\Rightarrow$ (G is pre-injective) (Slide 93)

<u>Rule 110</u> $c_1 = \cdots 00001101 0000 \cdots$

$c_2 = \cdots 000010110000 \cdots$

$\Big\}$ $G(c_1) = G(c_2)$ (not pre-injective)

$P_1 = 011010 \rightarrow 1111$

$P_2 = 010110 \rightarrow$, P_1 and P_2 only differ in the center two positions

Rule 110 has radius-1 neighborhood

So, without affecting $G(c)$ we may replace any pattern P_1 by P_2 and vice versa ?

c : - - - 00 0xxx - - - xxx0 00 - - $\quad (2^6-1)^k$ choices without P_2 $\quad = 63^k$

Ex5 $\downarrow$

$G(c)$: - - 00 $\underbrace{[******]}_{1}$ - - - $\underbrace{[******]}_{k}$ 00 - - $\quad (2^6)^k$ choices $= 64^k$

There must be a pattern that has no pre-image

G is not surjective

Algorithmic questions

Decidability in 1D : $\exists_{alg.}$ that on input always correctly determines (after finite computation) if the input has the desired property

Chaotic behavior

1) Means sensitivity to small perturbations

Equicontinuous: $\quad$ Observation window E $\quad \exists D$ $\quad$ States outside D do not affect states in E

Equicontinuous iff CA is eventually periodic

2) Mixing of configuration space is another property associated to chaos

CA is <u>transitive</u> if $\forall_{\text{cylinders } U,V} \; \exists_n \quad G^n(U) \cap V \neq \emptyset$

$\Uparrow$ (⇒ <u>sensitivity</u> to initial cond.)

CA is <u>mixing</u> if ..... $\exists_n \, \forall_{k \geq n} \; G^k(U) \cap V \neq \emptyset$

$\Uparrow$

<u>positively expansive</u>

Next class: CH6

- Read Sections 1 & 2 in detail
- Glance over Sect. 3 (definitions are repeated)
- Read Section 4
- Read 5: glance over 5.2
- Read 6.1, 6.2 in detail
- Glance over 6.3 — 8
- Glance over 9 (introduces CH7)

What does universal computation/machine mean to you?
Have we come to a final answer with respect to defining universality?
Is the concept of universality important, worth studying?
What do _you_ need to understand about universality?