Discrete Dynamical Systems and Automata

Livat Tyapaev

National Research Saratov State University

International Conference on p-ADIC MATHEMATICAL PHYSICS AND ITS APPLICATIONS (p-ADICS.2015), Belgrade, September 8, 2015

Automata and dynamics

The automaton transformations over alphabet

$$\mathbb{F}_p = \{0, 1, \ldots, p-1\},$$

where p prime number coincide with the continuous transformation of a ring of p-adic integers $\mathbb{Z}_p$ in p-adic metric that satisfy the Lipschitz condition with a constant 1.

Automata and dynamics

The automaton transformations over alphabet

$$\mathbb{F}_p = \{0, 1, \ldots, p-1\},$$

where p prime number coincide with the continuous transformation of a ring of p-adic integers $\mathbb{Z}_p$ in p-adic metric that satisfy the Lipschitz condition with a constant 1.

An (synchronous) automaton (transducer) is 6-tuple $\mathcal{A} = (\mathfrak{I}, \mathcal{S}, \mathcal{O}, S, O, s_0)$ where $\mathfrak{I}$ is an input alphabet, $\mathcal{S}$ is a set of states, $\mathcal{O}$ is an output alphabet, $S : \mathfrak{I} \times \mathcal{S} \to \mathcal{S}$ is a state update map, $O : \mathfrak{I} \times \mathcal{S} \to \mathcal{O}$ is an output map,$s_0 \in \mathcal{S}$ is an initial state.

Automata and dynamics

The automaton transformations over alphabet

$$\mathbb{F}_p = \{0, 1, \ldots, p-1\},$$

where p prime number coincide with the continuous transformation of a ring of p-adic integers $\mathbb{Z}_p$ in p-adic metric that satisfy the Lipschitz condition with a constant 1.

An (synchronous) automaton (transducer) is 6-tuple $\mathcal{A} = (\mathfrak{I}, \mathcal{S}, \mathcal{O}, S, O, s_0)$ where $\mathfrak{I}$ is an input alphabet, $\mathcal{S}$ is a set of states, $\mathcal{O}$ is an output alphabet, $S : \mathfrak{I} \times \mathcal{S} \to \mathcal{S}$ is a state update map, $O : \mathfrak{I} \times \mathcal{S} \to \mathcal{O}$ is an output map,$s_0 \in \mathcal{S}$ is an initial state.

Note that $\mathfrak{I}, \mathcal{O}$ are finite alphabets, however $\mathcal{S}$ could be an infinite set of states.

Automata and dynamics

An (synchronous) automaton (transducer) is 6-tuple $\mathcal{A} = (\mathfrak{I}, \mathcal{S}, \mathcal{O}, S, O, s_0)$ where $\mathfrak{I}$ is an input alphabet, $\mathcal{S}$ is a set of states, $\mathcal{O}$ is an output alphabet, $S : \mathfrak{I} \times \mathcal{S} \to \mathcal{S}$ is a state update map, $O : \mathfrak{I} \times \mathcal{S} \to \mathcal{O}$ is an output map,$s_0 \in \mathcal{S}$ is an initial state.

Let's consider only accessible automata: where any state $s \in \mathcal{S}$ of automaton $\mathcal{A}$ is reachable from initial state s_0 after a finite input word u was fed to the automaton.

Automata and dynamics

An (synchronous) automaton (transducer) is 6-tuple $\mathcal{A} = (\mathfrak{I}, \mathcal{S}, \mathcal{O}, S, O, s_0)$ where $\mathfrak{I}$ is an input alphabet, $\mathcal{S}$ is a set of states, $\mathcal{O}$ is an output alphabet, $S : \mathfrak{I} \times \mathcal{S} \to \mathcal{S}$ is a state update map, $O : \mathfrak{I} \times \mathcal{S} \to \mathcal{O}$ is an output map,$s_0 \in \mathcal{S}$ is an initial state.

Let's consider only accessible automata: where any state $s \in \mathcal{S}$ of automaton $\mathcal{A}$ is reachable from initial state s_0 after a finite input word u was fed to the automaton.

We assume further that $\mathfrak{I} = \mathcal{O} = \mathbb{F}_p$. As every automaton $\mathcal{A}$ transforms the input sequence into output sequence, we may say that automaton $\mathcal{A}$ performs a transformation f_{s_0} on $\mathbb{Z}_p$.

Automata and dynamics

An (synchronous) automaton (transducer) is 6-tuple $\mathcal{A} = (\mathfrak{I}, \mathcal{S}, \mathcal{O}, S, O, s_0)$ where $\mathfrak{I}$ is an input alphabet, $\mathcal{S}$ is a set of states, $\mathcal{O}$ is an output alphabet, $S : \mathfrak{I} \times \mathcal{S} \to \mathcal{S}$ is a state update map, $O : \mathfrak{I} \times \mathcal{S} \to \mathcal{O}$ is an output map,$s_0 \in \mathcal{S}$ is an initial state.

Let's consider only accessible automata: where any state $s \in \mathcal{S}$ of automaton $\mathcal{A}$ is reachable from initial state s_0 after a finite input word u was fed to the automaton.

We assume further that $\mathfrak{I} = \mathcal{O} = \mathbb{F}_p$. As every automaton $\mathcal{A}$ transforms the input sequence into output sequence, we may say that automaton $\mathcal{A}$ performs a transformation f_{s_0} on $\mathbb{Z}_p$.

Automata can be considered as (non-autonomous) dynamical systems on the space of p-adic integers $\mathbb{Z}_p$.

Automata and dynamics

For every accessible automaton $\mathcal{A}$ we associate a family of automata $\mathcal{A}_s = (\mathfrak{I}, \mathcal{S}, \mathcal{O}, S, O, s)$, and corresponding family $\mathcal{F} = \{f_s : s \in \mathcal{S}\}$ of 1-Lipschitz transformations $f_{\mathcal{A}_s}$, $s \in \mathcal{S}$ on $\mathbb{Z}_p$.

Automata and dynamics

For every accessible automaton $\mathcal{A}$ we associate a family of automata $\mathcal{A}_s = (\mathfrak{I}, \mathcal{S}, \mathcal{O}, S, O, s)$, and corresponding family $\mathcal{F} = \{f_s : s \in \mathcal{S}\}$ of 1-Lipschitz transformations $f_{\mathcal{A}_s}$, $s \in \mathcal{S}$ on $\mathbb{Z}_p$.

Reminder: transitivity of families of mappings

A family $\mathcal{F}$ of transformations on the set M is called transitive if for any pair $(a, b) \in M \times M$ there exists $f \in \mathcal{F}$ such that $f(a) = b$.
Note that a bijective transformation $f : M \to M$ is said to be transitive, whenever the family $\{e, f^{\pm 1}, f^{\pm 2}, \ldots\}$ is transitive.

Automata and dynamics

For every accessible automaton $\mathcal{A}$ we associate a family of automata $\mathcal{A}_s = (\mathfrak{I}, \mathcal{S}, \mathcal{O}, S, O, s)$, and corresponding family $\mathcal{F} = \{f_s : s \in \mathcal{S}\}$ of 1-Lipschitz transformations $f_{\mathcal{A}_s}$, $s \in \mathcal{S}$ on $\mathbb{Z}_p$.

Reminder: transitivity of families of mappings

A family $\mathcal{F}$ of transformations on the set M is called transitive if for any pair $(a, b) \in M \times M$ there exists $f \in \mathcal{F}$ such that $f(a) = b$.
Note that a bijective transformation $f : M \to M$ is said to be transitive, whenever the family $\{e, f^{\pm 1}, f^{\pm 2}, \dots\}$ is transitive.

Complete transitive

Automaton $\mathcal{A}$ is said to be completely transitive, if family $f_s \bmod p^n : \mathbb{Z}/p^n\mathbb{Z} \to \mathbb{Z}/p^n\mathbb{Z}$, $s \in \mathcal{S}$ is transitive over $\mathbb{Z}/p^n\mathbb{Z}$,

Automata and dynamics

V.S. Anashin showed that an automaton $\mathcal{A}$ is completely transitive if and only if the Lebesgue measure $\alpha(f_{s_0}) = 1$ for a closure of all the points $(\frac{x \mod p^k}{p^k}, \frac{f_{s_0}(x) \mod p^k}{p^k})$, $x \in \mathbb{Z}_p$, $k = 1, 2, 3, \ldots$ in the Euclidean unit square $\mathbb{E}^2 = [0,1] \times [0,1] \subset \mathbb{R}^2$.

Geometrical images of automata

Let's enumerate symbols of the alphabet $\mathbb{F}_p = \{0, 1, \ldots, p-1\}$ with natural numbers $\alpha_i \in \mathbb{F} = \{1, \ldots, p\}$. Next let's correspond to the word $u = \alpha_{k-1} \ldots \alpha_1 \alpha_0$ over the alphabet $\mathbb{F}$ the rational number $\vec{u} = \alpha_0 + \frac{\alpha_1}{p+1} + \ldots + \frac{\alpha_{k-1}}{(p+1)^{k-1}}$.

Geometrical images of automata

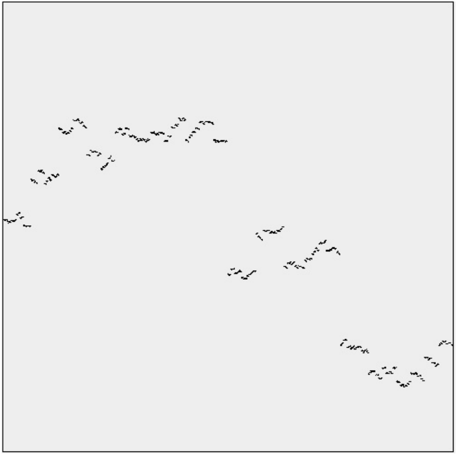
Let's enumerate symbols of the alphabet $\mathbb{F}_p = \{0, 1, \ldots, p-1\}$ with natural numbers $\alpha_i \in \mathbb{F} = \{1, \ldots, p\}$. Next let's correspond to the word $u = \alpha_{k-1} \ldots \alpha_1 \alpha_0$ over the alphabet $\mathbb{F}$ the rational number $\vec{u} = \alpha_0 + \frac{\alpha_1}{p+1} + \ldots + \frac{\alpha_{k-1}}{(p+1)^{k-1}}$. We consider all points of the Euclidean square $\Gamma = [1, p+1) \times [1, p+1) \subset \mathbb{R}^2$ of the form $(\vec{u}, f_{s_0}(\vec{u}))$ where u runs through all finite words. The set of these points $\Omega(f_{s_0}) \subset \Gamma$ is called a geometrical image of automaton $\mathcal{A}$.

Geometrical images of automata

Let's enumerate symbols of the alphabet $\mathbb{F}_p = \{0, 1, \ldots, p-1\}$ with natural numbers $\alpha_i \in \mathbb{F} = \{1, \ldots, p\}$. Next let's correspond to the word $u = \alpha_{k-1} \ldots \alpha_1 \alpha_0$ over the alphabet $\mathbb{F}$ the rational number $\vec{u} = \alpha_0 + \frac{\alpha_1}{p+1} + \ldots + \frac{\alpha_{k-1}}{(p+1)^{k-1}}$. We consider all points of the Euclidean square $\Gamma = [1, p+1) \times [1, p+1) \subset \mathbb{R}^2$ of the form $(\vec{u}, f_{s_0}(\vec{u}))$ where u runs through all finite words. The set of these points $\Omega(f_{s_0}) \subset \Gamma$ is called a geometrical image of automaton $\mathcal{A}$. The certain interest for us presents the task to describe a transitive family of automaton maps by means of the geometrical images.

Geometrical images of automata

Example of geomentical image of automaton:



Geometrical images of automata

For every state $s \in \mathcal{S}$ of an automaton $\mathcal{A}$ we associate a map $R_s : \mathbb{F}_p \to \mathbb{F}_p$ that transforms input symbol $x \in \mathbb{F}_p$ into output symbol $y \in \mathbb{F}_p$. Note, that R_s is not necessarily a surjective map.

Geometrical images of automata

For every state $s \in \mathcal{S}$ of an automaton $\mathcal{A}$ we associate a map $R_s : \mathbb{F}_p \to \mathbb{F}_p$ that transforms input symbol $x \in \mathbb{F}_p$ into output symbol $y \in \mathbb{F}_p$. Note, that R_s is not necessarily a surjective map. Consider an automaton $\mathcal{A}$ and the family $\{R_s : s \in \mathcal{S}\}$.

Geometrical images of automata

For every state $s \in \mathcal{S}$ of an automaton $\mathcal{A}$ we associate a map $R_s : \mathbb{F}_p \to \mathbb{F}_p$ that transforms input symbol $x \in \mathbb{F}_p$ into output symbol $y \in \mathbb{F}_p$. Note, that R_s is not necessarily a surjective map. Consider an automaton $\mathcal{A}$ and the family $\{R_s : s \in \mathcal{S}\}$.

A correspondence of each state $s \in \mathcal{S}$ of a map R_s creates a new automaton $\mathcal{B}$. Let's consider automata class $\mathcal{K}(\mathcal{A})$ that is constructed this way. Denote $\Omega_{\mathcal{B}}(f_s)$ a geometrical image of automaton $\mathcal{B}$.

Geometrical images of automata

For every state $s \in \mathcal{S}$ of an automaton $\mathcal{A}$ we associate a map $R_s : \mathbb{F}_p \to \mathbb{F}_p$ that transforms input symbol $x \in \mathbb{F}_p$ into output symbol $y \in \mathbb{F}_p$. Note, that R_s is not necessarily a surjective map. Consider an automaton $\mathcal{A}$ and the family $\{R_s : s \in \mathcal{S}\}$.

A correspondence of each state $s \in \mathcal{S}$ of a map R_s creates a new automaton $\mathcal{B}$. Let's consider automata class $\mathcal{K}(\mathcal{A})$ that is constructed this way. Denote $\Omega_{\mathcal{B}}(f_s)$ a geometrical image of automaton $\mathcal{B}$.

Theorem 1 (L.T., 2013)

An automaton $\mathcal{A}$ is completely transitive if and only if there exists the automaton $\mathcal{B} \in \mathcal{K}(\mathcal{A})$ and the geometrical images $\omega(f_{s_0}) \subset \Omega(f_{s_0})$, $\omega(f_s) \subset \Omega_{\mathcal{B}}(f_s)$ such that $\omega(f_{s_0})$ and $\omega(f_s)$ are affine equivalents.

Geometrical images of automata

An automaton $\mathcal{A}$ is completely transitive if and only if there exists the automaton $\mathcal{B} \in \mathcal{K}(\mathcal{A})$ and the geometrical images $\omega(f_{s_0}) \subset \Omega(f_{s_0})$, $\omega(f_s) \subset \Omega_{\mathcal{B}}(f_s)$ such that $\omega(f_{s_0})$ and $\omega(f_s)$ are affine equivalents.

Sketch of proof. If we consider the words of finite length k as prefixes of infinite words over the alphabet $\mathbb{F}_p$, that in their turn are elements of ring of p-adic integers $\mathbb{Z}_p$, the coincidence of these prefixes means that the distance between infinite words on a ultrametric space $\mathbb{Z}_p$ is equal to p^{-k}. That means, that the distance between f_s- images not more than p^{-k}, because the mappings $f_s : \mathbb{Z}_p \to \mathbb{Z}_p$ are 1-Lipschitz.

In addition, all such infinite words have a common prefix with length $k \geq 0$ as p-adic integers fall into the ball $B_{p^{-k}} = \{z \in \mathbb{Z}_p : |z - a|_p \leq p^{-k}\}$ with radius p^{-k} and center at $a \in \mathbb{Z}_p$. Moreover, the word u of length k as prefix of $z \in \mathbb{Z}_p$ is the reduction of z modulo p^k, i.e. $u \in \mathbb{Z}/p^k\mathbb{Z}$.

In addition, all such infinite words have a common prefix with length $k \geq 0$ as p-adic integers fall into the ball $B_{p^{-k}} = \{z \in \mathbb{Z}_p : |z - a|_p \leq p^{-k}\}$ with radius p^{-k} and center at $a \in \mathbb{Z}_p$. Moreover, the word u of length k as prefix of $z \in \mathbb{Z}_p$ is the reduction of z modulo p^k, i.e. $u \in \mathbb{Z}/p^k\mathbb{Z}$.
Furthermore, if $f_{s_0}(z) \equiv f_s(z) \pmod{p^k}$, then $f_{s_0}(z)$ and $f_s(z)$ falls into one ball, and points of $\Omega(f_{s_0})$, $\Omega(f_s)$ that correspond to input/output words of length k coincide and form elements of $\omega(f_{s_0})$ and $\omega(f_s)$ sets.

Obviously $f_{s_0}(z) \equiv f_s(z) \pmod{p^k}$ implies that $f_{s_0}(z) \equiv f_s(z) \pmod{p^\ell}$ for all $\ell = 1, \ldots, k-1$, therefore the points of images of words of length ℓ coincide.

Obviously $f_{s_0}(z) \equiv f_s(z) \pmod{p^k}$ implies that $f_{s_0}(z) \equiv f_s(z) \pmod{p^\ell}$ for all $\ell = 1, \ldots, k-1$, therefore the points of images of words of length ℓ coincide.
The case of $f_{s_0}(z) \not\equiv f_s(z) \pmod{p}$ means that $f_{s_0}(z)$, $f_s(z)$ fall into different balls of radius p^{-1} and the points in $\omega(f_{s_0})$ and $\omega(f_s)$ by appropriate input/output words of length 1 is compatible by affine transformation
$\phi_1 : (\vec{u}, f_s(\vec{u})) \mapsto (\vec{u}, f_{s_0}(\vec{u}) + d)$, $d \in \mathbb{Z}$.

If $f_{s_0}(z) \not\equiv f_s(z) \pmod{p^k}$ and $f_{s_0}(z) \equiv f_s(z) \pmod{p^{k-1}}$, then $f_{s_0}(z), f_s(z) \in B_{p^{k-1}}$ and points of images $\omega(f_{s_0})$ and $\omega(f_s)$ for words with length k are compatible by transformation $\phi_k : (\vec{u}, f_s(\vec{u})) \mapsto (\vec{u}, f_{s_0}(\vec{u}) + d),\ d \in \mathbb{Q}$.

If $f_{s_0}(z) \not\equiv f_s(z) \pmod{p^k}$ and $f_{s_0}(z) \equiv f_s(z) \pmod{p^{k-1}}$, then $f_{s_0}(z), f_s(z) \in B_{p^{k-1}}$ and points of images $\omega(f_{s_0})$ and $\omega(f_s)$ for words with length k are compatible by transformation $\phi_k : (\vec{u}, f_s(\vec{u})) \mapsto (\vec{u}, f_{s_0}(\vec{u}) + d)$, $d \in \mathbb{Q}$.

The transitivity of family $\mathcal{F}$ for an automaton $\mathcal{A}$ means that for a given pair u, w of strings with length k there exists a finite word v such that sends the automaton out of s_0 into the state s such that $f_s(\vec{u}) = \vec{w}$.

The set of maps R_s generates automaton $\mathcal{B} \in \mathcal{K}(\mathcal{A})$ such that the point $(\vec{u}, f_{s_0}(\vec{u})) \in \omega(f_{s_o})$ is transferred into a point $(\vec{u}, \vec{w}) \in \omega(f_s) \subset \Omega_{\mathcal{B}}(f_s)$ by transformation $\phi : (\vec{u}, fs_0(\vec{u})) \mapsto (\vec{u}, cf_{s_0}(\vec{u}) + d)$, $c, d \in \mathbb{Q}$.

The set of maps R_s generates automaton $\mathcal{B} \in \mathcal{K}(\mathcal{A})$ such that the point $(\vec{u}, f_{s_0}(\vec{u})) \in \omega(f_{s_o})$ is transferred into a point $(\vec{u}, \vec{w}) \in \omega(f_s) \subset \Omega_{\mathcal{B}}(f_s)$ by transformation $\phi : (\vec{u}, fs_0(\vec{u})) \mapsto (\vec{u}, cf_{s_0}(\vec{u}) + d)$, $c, d \in \mathbb{Q}$.
Sufficiency.

The set of maps R_s generates automaton $\mathcal{B} \in \mathcal{K}(\mathcal{A})$ such that the point $(\vec{u}, f_{s_0}(\vec{u})) \in \omega(f_{s_o})$ is transferred into a point $(\vec{u}, \vec{w}) \in \omega(f_s) \subset \Omega_{\mathcal{B}}(f_s)$ by transformation $\phi : (\vec{u}, fs_0(\vec{u})) \mapsto (\vec{u}, cf_{s_0}(\vec{u}) + d)$, $c, d \in \mathbb{Q}$.
Sufficiency. If the images $\omega(f_{s_0})$ and $\omega(f_s)$ coincide, then $f_{s_0}(\vec{u}) \equiv f_s(\vec{u}) \pmod{p^k}$ for some u of length k, and $f_s \in \mathcal{F}$. If $\omega(f_{s_0})$, $\omega(f_s)$ are ϕ-equivalent, then $f_s \in \mathcal{F}$. q.e.d.

Geometrical images of automata

V.S. Anashin proposed to consider the automaton mappings by means of β-expansion.The β-expansions are radix expansions in non-integer bases; Given $x \in \mathbb{R}$, $x \geq 0$ and $\beta \in \mathbb{R}$, $\beta > 1$ we call representation of the form $x = \sum_{n=1}^{\infty} \chi_n \beta^{-n}$ a β-expansion, where $\chi_i \in \{0, 1, \dots, \lfloor \beta \rfloor\}$. For word $x = x_{n-1} \dots\ x_1 x_0$, $x_i \in \mathbb{F}_p$ we put into correspondence a point
$\overleftarrow{x} = (\beta^{-n}(x_{n-1}\beta^{n-1} + \dots + x_1\beta + x_0)) \mod 1 \in [0, 1)$.

Geometrical images of automata

V.S. Anashin proposed to consider the automaton mappings by means of β-expansion.The β-expansions are radix expansions in non-integer bases; Given $x \in \mathbb{R}$, $x \geq 0$ and $\beta \in \mathbb{R}$, $\beta > 1$ we call representation of the form $x = \sum_{n=1}^{\infty} \chi_n \beta^{-n}$ a β-expansion, where $\chi_i \in \{0, 1, \ldots, \lfloor \beta \rfloor\}$. For word $x = x_{n-1} \ldots\ x_1 x_0$, $x_i \in \mathbb{F}_p$ we put into correspondence a point
$\overleftarrow{x} = (\beta^{-n}(x_{n-1}\beta^{n-1} + \ldots + x_1\beta + x_0)) \mod 1 \in [0, 1)$.
Considering all points $(\overleftarrow{x}, f_{s_0}(\overleftarrow{x}))$, where x ranges over all finite words, we get a set of points in the Euclidean unit square $\mathbb{E}^2 = [0, 1] \times [0, 1]$. A closure of all such points is the geometrical image of automaton $\mathcal{A}$ (β-plot of the automaton).

Geometrical images of automata

V.S. Anashin proposed to consider the automaton mappings by means of β-expansion.The β-expansions are radix expansions in non-integer bases; Given $x \in \mathbb{R}$, $x \geq 0$ and $\beta \in \mathbb{R}$, $\beta > 1$ we call representation of the form $x = \sum_{n=1}^{\infty} \chi_n \beta^{-n}$ a β-expansion, where $\chi_i \in \{0, 1, \ldots, \lfloor\beta\rfloor\}$. For word $x = x_{n-1} \ldots\ x_1 x_0$, $x_i \in \mathbb{F}_p$ we put into correspondence a point
$\overleftarrow{x} = (\beta^{-n}(x_{n-1}\beta^{n-1} + \ldots + x_1\beta + x_0)) \mod 1 \in [0, 1)$.
Considering all points $(\overleftarrow{x}, f_{s_0}(\overleftarrow{x}))$, where x ranges over all finite words, we get a set of points in the Euclidean unit square $\mathbb{E}^2 = [0, 1] \times [0, 1]$. A closure of all such points is the geometrical image of automaton $\mathcal{A}$ (β-plot of the automaton).
The problem of characterizing of the transitive family of automaton mappings in the β-plots remains open.

Words

Let $\mathcal{A} = (\mathfrak{I}, \mathcal{S}, \mathcal{O}, S, O, s_0)$ be an automaton, where $\mathfrak{I} = \mathcal{O} = \mathbb{F}_p$. State update function S and output function O can be continued to the set $\mathfrak{I}^* \times \mathcal{S}$ ($\mathfrak{I}^*$ is a set of all finite words over alphabet $\mathfrak{I}$) according to the following recurrent rules:

$$S(e, s) = s, O(e, s) = e,$$
$$S(x \cdot w, s) = S(w, S(x, s)),$$
$$O(x \cdot w, s) = O(x, s) \cdot O(w, S(x.s)),$$

where $x \in \mathfrak{I}$, $s \in \mathcal{S}$, and $w \in \mathfrak{I}^*$ are arbitrary elements. The automaton $\mathcal{A}$ defines a function $O(\cdot, s_0) : \mathfrak{I}^* \to \mathcal{O}^*$ that specifies the action of the automaton on finite words.

Words

Moreover, state update function S and output function O can be continued to the set $\mathfrak{I}^{\infty} \times \mathcal{S}$ ($\mathfrak{I}^{\infty}$ is a set of all infinite words over alphabet $\mathfrak{I}$).

Words

Moreover, state update function S and output function O can be continued to the set $\mathfrak{I}^\infty \times \mathcal{S}$ ($\mathfrak{I}^\infty$ is a set of all infinite words over alphabet $\mathfrak{I}$).

A mapping $f_{s_0} : \mathfrak{I}^\infty \to \mathcal{O}^\infty$ is said to be defined by automaton $\mathcal{A}$ if $f_{s_0}(w) = O(w, s_0)$ for any $w \in \mathfrak{I}^\infty$.

Words

Moreover, state update function S and output function O can be continued to the set $\mathfrak{I}^\infty \times \mathcal{S}$ ($\mathfrak{I}^\infty$ is a set of all infinite words over alphabet $\mathfrak{I}$).

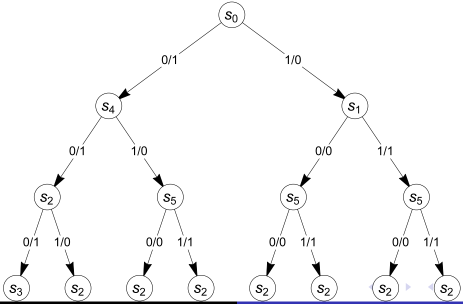
A mapping $f_{s_0} : \mathfrak{I}^\infty \to \mathcal{O}^\infty$ is said to be defined by automaton $\mathcal{A}$ if $f_{s_0}(w) = O(w, s_0)$ for any $w \in \mathfrak{I}^\infty$.

The mapping defined by an automaton is called the action on infinite words of this automaton.

Words

Remind that (initial) automaton action on a input sequences can be represented as labeled infinite homogeneous tree $T(\mathcal{A})$. The vertices of such tree correspond to the states of automaton, and for every symbol $x \in \mathfrak{I}$ of the input alphabet an arrow labeled by $x|O(x,s)$ starts from state s to state $S(x,s)$. The root of this tree $T(\mathcal{A})$ associated with initial state s_0 of automaton $\mathcal{A}$. From each vertex goes exactly $\#\mathfrak{I} = p$ arrows. To find out action of the automaton $\mathcal{A}$ on the word $w \in \mathfrak{I}^\infty$, we should move, starting from root of the tree $T(\mathcal{A})$, along the arrows of the tree so that the word w reads on the left parts of the labels along the arrows; then the product of all right parts of the labels will be equal to $O(w, s_0) \in \mathcal{O}^\infty$.

For instance, $\mathcal{A} = (\{0,1\}, \mathcal{S}, \{0,1\}, S, O, s_0)$, where $\mathcal{S} = \{s_0, s_1, s_2, \ldots\}$ can be represented as labeled directed tree $T(\mathcal{A})$. The vertices of the tree correspond to the states of automaton, and for every symbol $x \in \{0,1\}$ of the input alphabet an arrow labeled by $x|O(x,s)$ starts from state s to state $S(x,s)$: $S(0,s_0) = s_4$, $O(0,s_0) = 1$, $S(1,s_4) = s_5$, $O(1,s_4) = 0, \ldots$



Dynamics; Measure-preserving maps

Dynamical system on a measuarable spase $\mathbb{S}$ is understood as a triple $(\mathbb{S}, \mu, f)$, where $\mathbb{S}$ is a set endowed with a measure μ, and $f : \mathbb{S} \to \mathbb{S}$ is a measurable function.

Dynamics; Measure-preserving maps

Dynamical system on a measuarable spase $\mathbb{S}$ is understood as a triple $(\mathbb{S}, \mu, f)$, where $\mathbb{S}$ is a set endowed with a measure μ, and $f : \mathbb{S} \to \mathbb{S}$ is a measurable function.

A trajectory of the dynamical system is a sequense $x_0, x_1 = f(x_0), \ldots, x_i = f(x_{i-1}) = f^i(x_0), \ldots$ of points of the space $\mathbb{S}$, x_0 is called an initial point of the trajectory.

Reminder: meausure-preserging maps

A mapping $F : \mathbb{S} \to \mathbb{S}$ of measurable space $\mathbb{S}$ into a measurable space $\mathbb{Y}$ endowed with probabilistic measure μ and ν, respectively, is said to be *measure-preserving* whenever $\mu(F^{-1}(S)) = \nu(S)$ for each measurable subset $S \subseteq \mathbb{S}$.

Dynamics; Measure-preserving maps

A mapping $F : \mathbb{S} \to \mathbb{S}$ of measurable space $\mathbb{S}$ onto $\mathbb{S}$ endowed with probabilistic measure μ, is said to be *measure-preserving* whenever $\mu(F^{-1}(S)) = \mu(S)$ for each measurable subset $S \subseteq \mathbb{S}$.

Dynamics; Measure-preserving maps

A mapping $F : \mathbb{S} \to \mathbb{S}$ of measurable space $\mathbb{S}$ onto $\mathbb{S}$ endowed with probabilistic measure μ, is said to be *measure-preserving* whenever $\mu(F^{-1}(S)) = \mu(S)$ for each measurable subset $S \subseteq \mathbb{S}$.

Consider dynamical system $(\mathbb{Z}_p, \mu, f_{\mathcal{A}})$ on $\mathbb{Z}_p$, where map $f_{s_0} : \mathbb{Z}_p \to \mathbb{Z}_p$ defined by some automaton $\mathcal{A} = (\mathbb{F}_p, \mathcal{S}, \mathbb{F}_p, S, O, s_0)$. The ring $\mathbb{Z}_p$ can be endowed with a probability measure μ, thus becoming a probability space. The latter measure is a normalized Haar measure. The base of elementary measurable subsets are all balls $B_{p^{-k}}(a)$ of non-zero radii p^{-k}; and we put $\mu(B_{p^{-k}}(a)) = p^{-k}$.

Asynchronous automata

Now consider the acynchronous automata.

An *asynchronous automaton (transducer)* is a 6-tuple $\mathcal{B} = (\mathfrak{I}, \mathcal{S}, \mathcal{O}, S, O, s_o)$, where

- $\mathfrak{I}$, $\mathcal{O}$ are finite alphabets,
- $\mathcal{S}$ is a set of states,
- $S : \mathfrak{I} \times \mathcal{S} \to \mathcal{S}$ is the state update function,
- $O : \mathfrak{I} \times \mathcal{S} \to \mathcal{O}^*$, where $\mathcal{O}^*$ denotes the set of all finite strings (words) over $\mathcal{O}$, and
- s_0 is the initial state.

Asynchronous automata

Given a transducer $\mathcal{B}$ and an infinite input string $\ldots\alpha_2\alpha_1\alpha_0 \in \mathfrak{I}^\infty$, the corresponding output sequence $\{\beta_i\}$ is defined recursively by

$$\beta_i = \mathcal{O}(\alpha_i, s_i)$$

The concatenation $\ldots\beta_2\beta_1\beta_0$ of the output sequence is called the output string. This string is usually infinite, but will be finite if only finitely many β_i's are nonempty. We say that the transducer is *nondegenerate* if every infinite input string results in an infinite output string. In this case, the function $f_{s_0} : \mathfrak{I}^\infty \to \mathcal{O}^\infty$ mapping each input string to the corresponding output string is called the *rational* function defined by the given transducer.

Asynchronous automata

Given a transducer $\mathcal{B}$ and an infinite input string $\dots\alpha_2\alpha_1\alpha_0 \in \mathfrak{I}^\infty$, the corresponding output sequence $\{\beta_i\}$ is defined recursively by

$$\beta_i = \mathcal{O}(\alpha_i, s_i)$$

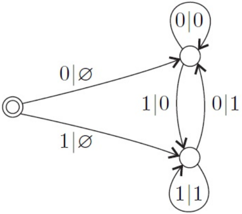
The concatenation $\dots\beta_2\beta_1\beta_0$ of the output sequence is called the output string. This string is usually infinite, but will be finite if only finitely many β_i's are nonempty. We say that the transducer is *nondegenerate* if every infinite input string results in an infinite output string. In this case, the function $f_{s_0} : \mathfrak{I}^\infty \to \mathcal{O}^\infty$ mapping each input string to the corresponding output string is called the *rational* function defined by the given transducer. Rational functions are always continuous, and any invertible rational function is a homeomorphism.

Asynchronous automata

Roughly speaking, an asynchronous transducer is a automaton that converts an input string of arbitrary length to an output string. The transducer reads one symbol at a time, changing its internal state and outputting a finite sequence of symbols at each step.

Asynchronous automata

Roughly speaking, an asynchronous transducer is a automaton that converts an input string of arbitrary length to an output string. The transducer reads one symbol at a time, changing its internal state and outputting a finite sequence of symbols at each step. Asynchronous transducers are a natural generalization of synchronous transducers, which are required to output exactly one symbol for every symbol read. For instance, the asynchronous automaton represented by Moor diagram:



Asynchronous automata

Let's define a function of special type for asynchronous transducer where input and output alphabets are same. More over, let's $\mathfrak{I} = \mathcal{O} = \mathbb{F}_p = \{0, 1, \dots p - 1\}$.

Asynchronous automata

Let's define a function of special type for asynchronous transducer where input and output alphabets are same. More over, let's $\mathfrak{I} = \mathcal{O} = \mathbb{F}_p = \{0, 1, \dots p-1\}$.

A mapping $f_{s_0} : \mathbb{Z}_p \to \mathbb{Z}_p$ is called *n-unit delay* whenever given an asynchronous transducer $\mathcal{B} = (\mathbb{F}_p, \mathcal{S}, \mathbb{F}_p, S, O, s_0)$ traslated input string $\alpha = \dots \alpha_n \dots \alpha_1 \alpha_0$ (viewed as a infinite stream of symbols) into output string of form $\beta = \dots \beta_{n+1} \beta_n$.

Asynchronous automata

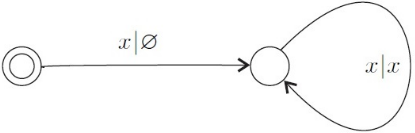
We assume that an asynchronous transducer works in a framework of discrete time steps.

Asynchronous automata

We assume that an asynchronous transducer works in a framework of discrete time steps. A unit-delay mapping is one that simply produces the output one time unit later.

Asynchronous automata

We assume that an asynchronous transducer works in a framework of discrete time steps. A unit-delay mapping is one that simply produces the output one time unit later. For example, a unilateral shift is defined by finite asynchronous automaton with unit-delay, that is the initial state of the automaton, irrespective of the incomming letter, outputs an empty word; after that, the automaton outputs the incoming words without changes.



Asynchronous automata

A unilateral shift is defined by finite asynchronous automaton with unit-delay, that is the initial state of the automaton, irrespective of the incomming letter, outputs an empty word; after that, the automaton outputs the incoming words without changes.

Specifically, if the transducer reads as input a symbol α_t at time t, it will produce the symbol as output at time $t+1$. At time $t=0$, the transducer outputs nothing.

Asynchronous automata

A unilateral shift is defined by finite asynchronous automaton with unit-delay, that is the initial state of the automaton, irrespective of the incomming letter, outputs an empty word; after that, the automaton outputs the incoming words without changes.

Specifically, if the transducer reads as input a symbol α_t at time t, it will produce the symbol as output at time $t+1$. At time $t=0$, the transducer outputs nothing. We indicate this by saying that the transducer translates input $\ldots\alpha_2\alpha_1\alpha_0$ into output $\ldots\beta_2\beta_1 e$, where e is empty word.

Asynchronous automata

A n-unit delay transducer is one that produces the output n times unit later; that is, the input $\ldots\alpha_n\ldots\alpha_1\alpha_0$ translated into output $\ldots\beta_{n+1}\beta_n e^n$,

Asynchronous automata

A n-unit delay transducer is one that produces the output n times unit later; that is, the input $\ldots\alpha_n\ldots\alpha_1\alpha_0$ translated into output $\ldots\beta_{n+1}\beta_n e^n$, meaning again that transducer produces no output for the first n times slots.

Theorem 2

A n-unit delay mapping $f_{s_0}:\mathbb{Z}_p\to\mathbb{Z}_p$ is continuous.

Criteria of preserve the meausure

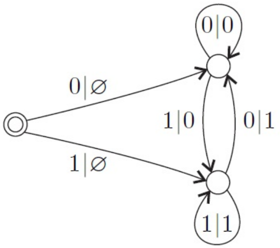
Given n-unit delay transducer $\mathcal{B} = (\mathbb{F}_p, \mathcal{S}, \mathbb{F}_p, S, O, s_0)$, and tree $T(\mathcal{B})$ we call a reachable set of states V of automaton $\mathcal{B}$, if each vertex of tree $T(\mathcal{B})$ associated with element of V is accessible for n steps from the root s_0. Element $s \in V$ is called reachable state.

Theorem 3

A n-unit delay mapping $f_{s_0} : \mathbb{Z}_p \to \mathbb{Z}_p$ is measure-preserving, if and only if there exists exactly p^n reachable states $s \in V$ such that for any finite output word β the action of the automaton $\mathcal{B}$ on suitable input words $\alpha(\beta)$ coincide with output word $O(s, \alpha(\beta)) = \beta$.

Thank You!

Asynchronous automaton



Unilateral shift