



HAL
open science

Zigzags in Turing machines

Anahi Gajardo, Pierre Guillon

► **To cite this version:**

| Anahi Gajardo, Pierre Guillon. Zigzags in Turing machines. 2010. hal-00460662v2

HAL Id: hal-00460662

<https://hal.science/hal-00460662v2>

Preprint submitted on 30 Dec 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Zigzags in Turing machines^{*}

Anahí Gajardo¹ and Pierre Guillon²

¹ Departamento de Ingeniería Matemática, Universidad de Concepción, Casilla 160-C, Concepción, Chile anahi@ing-mat.udec.cl

² DIM - CMM, UMI CNRS 2807, Universidad de Chile, Av. Blanco Encalada 2120, Santiago, Chile pguillon@dim.uchile.cl

Abstract. We study one-head machines through symbolic and topological dynamics. In particular, a subshift is associated to the system, and we are interested in its complexity in terms of realtime recognition. We emphasize the class of one-head machines whose subshift can be recognized by a deterministic pushdown automaton. We prove that this class corresponds to particular restrictions on the head movement, and to equicontinuity in associated dynamical systems.

Keywords: Turing machines, discrete dynamical systems, subshifts, formal languages.

We study the dynamics of a system consisting in a finite automaton (the head) that can write and move over an infinite tape, like a Turing machine. We use the approach of symbolic and topological dynamics. Our interest is to understand its properties and limitations, and how dynamical properties are related to computational complexity.

This approach was initiated by Kůrka in [1] with two different topologies: one focused on the machine head, and the other on the tape. The first approach was further developed in [2,3]. More recently, in [4,5], a third kind of dynamical system was associated to Turing machines, taking advantage of the following specificity: changes happen only in the head position whilst the rest of the configuration remains unaltered. The whole evolution can therefore be described by the sequence of states taken by the head and the symbols that it reads. This observation actually yields a factor map between Kůrka's first dynamical system and a one-sided subshift.

In [4], it has been proved that machines with a sofic subshift correspond to machines whose head makes only bounded cycles. We prove here a similar characterization of machines with a shift that can be recognized by a deterministic pushdown automaton. Moreover, we establish links between these two properties and equicontinuity in all three spaces.

In the first section, we recall the definitions and fundamental results. The second section is devoted to defining the different dynamical systems associated to one-head machines, and to stating basic results about equicontinuity within

^{*} This work has been supported by ECOS-Sud Project and CONICYT FONDECYT #1090568.

these systems. In the last section, we define the class of bounded-zigzag machines and state our main results.

1 Preliminaries

Consider a finite alphabet A , and \mathbb{M} to stand either for \mathbb{N} or for \mathbb{Z} . For a finite word $u \in A^*$, we will note $|u|$ its length, and index its letters from 0 to $|u| - 1$, unless specified otherwise. We denote $A^{\leq m}$ the set of words on A of length at most $m \in \mathbb{N}$. If $i, j \in \mathbb{Z}$ and $i \leq j$, $\llbracket i, j \rrbracket$ will denote the closed interval of integers i, \dots, j , $\llbracket i, j \llbracket = \llbracket i, j - 1 \rrbracket$, etc. A point $x \in A^{\mathbb{M}}$ will be called *configuration*. For a configuration or a word x , we define $x_{\llbracket i, j \rrbracket} = x_i \dots x_j$. $A \sqcup B$ will denote the disjoint union of two sets A and B .

1.1 Topological dynamics

A *dynamical system* (DS) is a pair (X, F) where X is a metric space and F a continuous self-map of X . Sometimes the space will be implicit.

The orbit of a point $x \in X$ is the set of the $F^t(x)$ for all *iteration* $t > 0$. A point x is called *preperiodic* if there exist two naturals q, p such that $F^{q+p}(x) = F^q(x)$. If q and p are minimal, then q is called the *transient* and p the *period*. When $t = 0$, x is called *periodic*.

A point $x \in X$ is *isolated* if there is an $\varepsilon > 0$ such that the ball of radius ε and center x contains only x . A point $x \in X$ is *equicontinuous* for F if, for any $\varepsilon > 0$, there exists some $\delta > 0$ such that, for any $y \in X$ with $d(x, y) < \delta$, we have that, for all $t \in \mathbb{N}$, $d(F^t(x), F^t(y)) < \varepsilon$. The DS (X, F) is *equicontinuous* if, for any $\varepsilon > 0$, there exists some $\delta > 0$ such that, for any $x, y \in X$ with $d(x, y) < \delta$, we have that, for all $t \in \mathbb{N}$, $d(F^t(x), F^t(y)) < \varepsilon$. When X is compact, this is equivalent to having only equicontinuous points. The DS (X, F) is *almost equicontinuous* if it has a residual set of equicontinuous points.

A DS (X, F) is a *factor* of a DS (Y, G) if $\phi G = F \phi$ for some continuous onto map $\phi : Y \rightarrow X$, then called a *factor map*.

1.2 Subshifts

We can endow the space $A^{\mathbb{M}}$ of configurations with the product of the discrete topology of A . It is based on the cylinders $[u]_i = \{x \in A^{\mathbb{M}} \mid x_{\llbracket i, i+k \llbracket} = u\}$, where $i \in \mathbb{M}$, $k \in \mathbb{N}$ and $u \in A^k$; this notation shall be extended to semi-infinite words. If $\mathbb{M} = \mathbb{Z}$, $u \in A^{2r+1}$ and $r \in \mathbb{N}$, we note $[u] = [u]_{-r}$.

This topology corresponds to the metric $d : x, y \mapsto 2^{-\min_{x_i \neq y_i} |i|}$. In other words, $d(x, y) \leq 2^{-i} \Leftrightarrow x_{\llbracket -i, i \rrbracket} = y_{\llbracket -i, i \rrbracket}$; two points are "close to each other" if they coincide "around position 0". It is easy to extend this metric to spaces $A^{\mathbb{M}} \times Q$ and $A^{\mathbb{M}} \times Q \times \mathbb{Z}$. In that setting, $A^{\mathbb{M}}$ and $A^{\mathbb{M}} \times Q$ are compact, but $A^{\mathbb{M}} \times Q \times \mathbb{Z}$ is not.

The *shift* map is the function $\sigma : A^{\mathbb{M}} \rightarrow A^{\mathbb{M}}$ defined by $\sigma(x)_i = x_{i+1}$. A *subshift* Σ is a closed subset of $A^{\mathbb{M}}$ which is also invariant by σ . It can be seen as a compact DS where the map is σ .

A subshift Σ is characterized by its *language*, containing all finite patterns that appear in some of its configurations: $\mathcal{L}(\Sigma) = \{z_{[i,j]} \mid z \in \Sigma \text{ and } i, j \in \mathbb{M}\}$. We denote $\mathcal{L}_n(\Sigma) = \mathcal{L}(\Sigma) \cap A^n$. If the language $\mathcal{L}(\Sigma)$ is regular, then we say that Σ is *sofic*. Equivalently, a sofic subshift can be seen as the set of labels of infinite paths in some finite arc-labeled graph; this graph basically corresponds to the finite automaton that recognizes its language, without initial nor terminal state.

Any subshift can also be defined from a set of forbidden finite patterns $\mathcal{F} \subset A^*$ by $\Sigma = \{z \in A^{\mathbb{M}} \mid \forall i, j \in \mathbb{M}, z_{[i,j]} \notin \mathcal{F}\}$. If \mathcal{F} can be chosen to be finite, then Σ is a subshift *of finite type* (SFT).

A DS F on $A^{\mathbb{M}}$ is completely determined by the family of its factor subshifts, *i.e.* the factors which are also subshifts in some alphabet. Up to some letter renaming, all factor subshifts of F are of the form $(\mathcal{P}(F^j(x)))_{j \in \mathbb{N}}$, where \mathcal{P} is a finite partition of X into closed open sets, and $\mathcal{P}(y)$ denotes the unique element of this partition which contains $y \in X$.

1.3 Deterministic pushdown automata

Definition 1 A deterministic pushdown automaton (DPDA) is a tuple $(A, \Omega, \Gamma, \perp, \lambda, o_0, F)$ where A is the input alphabet, Ω is the set of states, Γ is the stack alphabet, $\perp \in \Gamma$ is the stack bottom, o_0 is the initial state, $F \subset \Omega$ is the subset of terminal states and $\lambda : A \times \Omega \times \Gamma \rightarrow \Omega \times \Gamma^{\leq 2}$ is the transition function such that: if $\lambda(a, o, \perp) = (o', \mu)$, then μ contains exactly one \perp , which is on its end, and if $\lambda(a, o, \beta) = (o', \mu)$ with $\beta \neq \perp$, then μ does not contain any \perp .

An (infinite) arc-labeled graph G is associated to the automaton. Its set of vertices is $\Omega \times (\Gamma \setminus \{\perp\})^* \perp$, and there exists an arc from (e, μ) to (f, ν) labeled a if and only if $\nu = \rho \mu_{[1, |\mu| - 1]}$ and $\lambda(a, e, \mu_0) = (f, \rho)$. The word μ is called the stack content.

The language L recognized by the automaton consists of all words w in A^* such that there exists a finite path in G with label w , starting on vertex (o_0, \perp) and ending in some vertex (o, μ) with $o \in F$. A subshift is recognized by the automaton if its language is recognized by the automaton.

2 Turing Machines

In this article, a Turing Machine (TM) is a triple (A, Q, δ) , where A and Q are the finite *tape alphabet* and *set of state*, and $\delta : A \times Q \rightarrow A \times Q \times \{-1, 1\}$ the *rule*. We do not particularize any halting state. We can see the TM as evolving on a bi-infinite tape. The phase space is $X = A^{\mathbb{Z}} \times Q \times \mathbb{Z}$. Any element of X is called a configuration and represents the state of the tape, the state of the head and its position. We consider here the topology introduced in Section 1.1. Thus, the farther the head is from the center, the less important become the read symbols, but the head state and position remain important. On this (non-compact) space, $T : X \rightarrow X$ by $T(x, q, i) = (x_{[-\infty, i[} a x_{i, \infty[}, p, i + d)$ if $\delta(x_i, q) = (a, p, d)$ gives

the corresponding DS. We can extend the shift function to TM configurations by $\sigma : (x, q, i) \mapsto (\sigma(x), q, i - 1)$, and it clearly commutes with T .

We can represent the head state and position by adding a “mark” on the tape. If we want a compact space, this corresponds to the following phase space:

$$X_H = \{x \in (A \sqcup (A \times Q))^{\mathbb{Z}} \mid |\{i \in \mathbb{Z} \mid x_i \in A \times Q\}| \leq 1\}$$

where the head position is implicitly given by the only cell with a symbol in $(A \times Q)$, and the function $T_H : X_H \rightarrow X_H$ is defined by $T_H(x]_{-\infty, i}[(b, q)x]_{i, \infty}[) = y]_{-\infty, i+d}[(y_{i+d}, p)y]_{i+d, \infty}[)$, where $y = x]_{-\infty, i}[[ax]_{i, \infty}[)$ and $\delta(b, q) = (a, p, d)$, and $T_H(x) = x$ if x does not contain any symbol in $A \times Q$. With the topology of X_H as a subshift of $(A \sqcup (A \times Q))^{\mathbb{Z}}$, the head state and movement are less important when the head is far from 0. This model corresponds to the TM *with moving head* defined by K urka in [1], which highlights the tape configuration. It is a particular case of cellular automaton, *i.e.* based on some uniformly-applied local rule. We can intuitively see a continuous injection $\Phi : X \rightarrow X_H$ such that $\Phi T = T_H \Phi$ and $\Phi \sigma = \sigma \Phi$.

Focusing on the movements and states of the head, [1] also defines the system *with moving tape* $T_T : X_T \rightarrow X_T$ on the (compact) space $X_T = A^{\mathbb{Z}} \times Q$ by $T_T(x, q) = (\sigma^d(x]_{-\infty, 0}[[ax]_{0, \infty}[), p)$ if $\delta(x_0, q) = (a, p, d)$. Here the head is assumed to be always at position 0, and the tape is shifted at each step according to the rule. There is a continuous non-injective surjection $\Psi : X \rightarrow X_T$ such that $\Psi T = T_T \Psi$.

Finally, we can have a vision centered on the head and which emphasizes only the relevant part of the configuration, as in [4,5]. The system S_T is the one-sided subshift on alphabet $Q \times A$, which is the image of the factor map $\tau_T : X_T \rightarrow S_T$ defined by $\tau_T(x, q)_t = (y_0, p)$ if $(y, p) = T_T^t(x, q)$. In other words, it represents the sequence of pairs corresponding to the successive states of the head and the letters that it reads.

$$\begin{array}{ccccccc} X_H & \xleftarrow{\Phi} & X & \xrightarrow{\Psi} & X_T & \xrightarrow{\tau_T} & S_T \\ T_H \downarrow & & T \downarrow & & T_T \downarrow & & \sigma \downarrow \\ X_H & \xleftarrow{\Phi} & X & \xrightarrow{\Psi} & X_T & \xrightarrow{\tau_T} & S_T \end{array}$$

Similarly, we will note S_H the one-sided subshift on alphabet $Q \sqcup (A \times Q)$ which is the image of the factor map $\tau_H : X_H \rightarrow S_H$ defined by $\tau_H(x)_t = T_H^t(x)_0$. Unlike S_T , this subshift does not always contain the relevant information, since the head can be completely absent.

2.1 Equicontinuous configurations

Topological notions can actually formalize various types of head movements. One first example is equicontinuity of the DS T_T . It is strongly related with periodicity, as the next remark establishes. This is natural since the symbol that the head reads in X_T is always at position 0. Hence, if the head visits an infinite

number of cells, say to the right, any perturbation on the initial configuration will get to position 0, and thus will become largely significant for this topology. We conclude the following.

Remark 1 *Let $x \in X$ be a configuration and T a machine over X . The following statements are equivalent:*

1. *The head position on x is bounded.*
2. *x is **preperiodic** for T .*
3. *$\Phi(x)$ is **preperiodic** for T_H .*
4. *$\Psi(x)$ is **equicontinuous** for T_T .*
5. *$\tau_T\Psi(x)$ is **preperiodic** and **isolated** –i.e. **equicontinuous–** in S_T .*

Moreover, if one of the above occurs, then $\Psi(x)$ is preperiodic for T_T , x is equicontinuous for T and $\Phi(x)$ is equicontinuous for T_H . The set of equicontinuous configurations for T_T is a union of cylinders of X_T .

If $\Psi(x)$ is preperiodic for T_T , then $\tau_T\Psi(x)$ is also periodic (for σ), but x need not be periodic for T . For example, a machine that simply moves to the left on every symbol will produce a periodic point for T_T if the initial configuration x is spatially periodic. From the previous remark, such a point is not equicontinuous, and $\tau_T\Psi(x)$ is a non-isolated periodic point in S_T , because any perturbation of x will produce a neighbor of $\tau_T\Psi(x)$ in S_T . Periodic points for T generate isolated periodic points in S_T because, once the system falls in the periodic behavior, its future is fixed.

Preperiodicity in T also implies equicontinuity in T_H , but T_H may have other equicontinuous points. The previously mentioned machine which always go to the left produces equicontinuous points for T_H which are not equicontinuous nor preperiodic for T_T .

The following proposition states that the equicontinuity of preperiodic configurations is transmitted to factor subshifts of T_H , which will be helpful in the sequel.

Proposition 1 *If $z \in S_H$ is a preperiodic word involving the machine head infinitely often, then it is isolated.*

Proof. We can assume that z is periodic, and then include the transient evolution in a larger ball. Let $p \in \mathbb{N} \setminus \{0\}$ be the period of z ; let us prove that the ball $U = [z]_{[0, |Q||A|^{p+1}(p+1)^2]}_0$ of S_H is equal to $\{z\}$. Let $z' \in U$ and $x \in \tau_H^{-1}(z')$. It can be seen that the head computing over z' always remains between the positions $\lfloor -p/2 \rfloor$ and $\lfloor p/2 \rfloor$, which correspond to at most $|Q||A|^{p+1}(p+1)$ distinct finite patterns. Hence there are $i < j \leq |Q||A|^{p+1}(p+1)$ such that $T^i(x) = T^j(x)$; as a consequence $\sigma^i(z')$ is $(j-i)$ -periodic. Together with $\sigma^i(z)$, they are both $(j-i)p$ -periodic and coincide on their first $(j-i)p$ letters, since $(j-i)p \leq |Q||A|^{p+1}(p+1)^2 - i$. As a conclusion, $z' = z$. \square

2.2 Preperiodic machines

When all the configurations are uniformly preperiodic, we say that the system is preperiodic, *i.e.* there exist q, p such that $T^{q+p} = T^q$. In the present case, global preperiodicity of each of the considered systems comes directly from local preperiodicity of T ; and it is equivalent to global equicontinuity of each of the systems as the next theorem establishes.

Theorem 1 *Considering a machine, the following statements are equivalent:*

1. *The head position is (uniformly) bounded.*
2. *Any configuration of X (or X_H, X_T) is **preperiodic**.*
3. *T (or T_H, T_T, S_T, S_H) is **preperiodic**.*
4. *T (or T_H, T_T, S_T, S_H) is **equicontinuous**.*
5. *S_T (or S_H) is **finite**.*

Proof. We give only a sketch of the main implications.

- It is quite obvious from the commutation diagrams that the preperiodicity of T, T_H and T_T are equivalent, and they imply those of S_T and S_H . They also imply, from Remark 1, that the head position is bounded.
- Clearly, the equicontinuity of T and T_H are equivalent.
- It is known from cellular automata theory that the equicontinuity of T_H , its preperiodicity, that of all its configuration and the finiteness of S_H are equivalent.
- If the head position on all configurations is bounded, then from Remark 1 they are all equicontinuous for T_T . X_T being compact, T_T is equicontinuous.
- It is obvious that S_T is finite if and only if the head reads a bounded part of the initial configuration. □

2.3 Sofic machines

Now we allow computations where the head can go arbitrarily "far", but without ever making "large" movements back.

Definition 2 *We say that a machine makes a right-cycle (left-cycle) of width $N \in \mathbb{N}$ over a configuration $x \in A^{\mathbb{Z}} \times Q \times \mathbb{Z}$ and a cell $i \in \mathbb{Z}$ if there exist time steps $0 = t_0 < t_1 < t_2$ such that the head position is i at time 0 and t_2 , and is $i + N$ ($i - N$) at time t_1 .*

In this section, we consider machines whose cycles have bounded width, *i.e.* there exists an integer N such that the machine cannot make any cycle wider than N . These machines have been studied in [5,4], where it was proved that they are exactly the machines for which S_T is sofic.

Theorem 2 *Considering a machine, the following statements are equivalent:*

1. *S_T is **sofic**.*
2. *All configurations of X_H that contain the head are **equicontinuous**.*

Proof.

- 1 \Rightarrow 2 From [4], we know that there exists an integer $N \in \mathbb{N}$ such that the machine cannot make any cycle wider than $N \in \mathbb{N}$, and let $x \in X_H$ a configuration containing the head within $\llbracket -k, k \rrbracket$, for some $k \in \mathbb{N}$. Let us show that if $y \in [x_{\llbracket -k-N, k+N \rrbracket}]$, then for every $t \in \mathbb{N}$ we have $T_H^t(y) \in [T_H^t(x)_{\llbracket -k, k \rrbracket}]$. Let us remark that while the head is inside $\llbracket -k - N, k + N \rrbracket$, we necessarily have $T_H^t(y) \in [T_H^t(x)_{\llbracket -k, k \rrbracket}]$. Let us suppose that there exists $j \in \mathbb{N}$ such that the head is outside $\llbracket -k - N, k + N \rrbracket$ at time j and let us take this j minimal. Then the heads of $T_H^j(x)$ and $T_H^j(y)$ are outside $\llbracket -k - N, k + N \rrbracket$. At some moment, the head has gone from k to $k + N$ (or from $-k - N$); if it comes back to $\llbracket -k, k \rrbracket$, it would make a cycle. Therefore, the head cannot come back to $\llbracket -k, k \rrbracket$, and this is true both for x and y , and we have the result.
- 2 \Rightarrow 1 Conversely, assume that the head can do arbitrarily wide right-cycles in cell 0, *i.e.* for each $j \in \mathbb{N}$ there exists a cylinder $[u^j]_0$ of X_H with $u^j \in (A \times Q)^{A^{n_j}}$, with $n_j > j$, such that over each configuration of $[u^j]_0$, the head starts at 0, it visits the whole interval $\llbracket 0, n_j \rrbracket$ and comes back to cell 0. Let us take some configuration c^j in each cylinder $[u^j]_0$. By compactness, the sequence $(c^j)_{j \in \mathbb{N}}$ admits an adhering value c , on which the head necessarily goes infinitely far to the right without ever coming back to cell 0. By construction, for any N , there is some $j \in \mathbb{N}$ such that the configuration $c_{\llbracket -N, N \rrbracket}^j = c_{\llbracket -N, N \rrbracket}$. But there exists a time $t \in \mathbb{N}$ such that $T_H^t(c^j)$ has the head in cell 0, whilst $T_H^t(c)$ has not; hence c is not equicontinuous. \square

From [5], any of the former properties implies that any configuration is either preperiodic or gives rise to a movement of the head arbitrarily far in some direction, but the converse is not true. Any configuration of S_H is preperiodic, hence this subshift is numerable.

3 Bounded-zigzag machines

Whilst the sofic machines did not allow any large cycle, we can wonder what happens when allowing a single one, or a finite number of these. The first step is to allow one cycle of arbitrary width but to forbid two overlapped unbounded cycles (zigzags). We remark that two independent cycles, each on a different direction, are allowed in this case.

Definition 3 *We say that a machine makes a right-zigzag (resp., left-zigzag) of width $N \in \mathbb{N}$ over a configuration $x \in A^{\mathbb{Z}} \times Q \times \mathbb{Z}$ and a cell $i \in \mathbb{Z}$, if there exist time steps $0 = t_0 < t_1 < t_2$ such that the machine position is i at times t_0 and t_2 , and $i + N$ (resp., $i - N$) at time t_1 . We say that a machine is bounded-zigzag if the maximal width of the zigzags that it can make is finite.*

3.1 Complexity of S_T

While bounded cycle machines have a sofic shift S_T , the bounded-zigzag machines have a subshift recognized by a deterministic pushdown automata. The

words of S_T contain information about the tape symbols and the head state. From this data, it is possible to deduce the tape symbol of the visited cells and the relative position of the head at each time step. In order to recognize S_T , we can register this information and check its coherence at each time step. When the width of the cycles is bounded, we only need to register a finite part of the tape (bounded-cycle machines have a subshift that can be recognized by a finite state automaton).

When only one “wide” cycle can be done, we can register the information in a stack, from which it can be read exactly once (and is lost forever once read). This corresponds to the fact that the cells registered in the stack cannot be visited any more and zigzags cannot be allowed. The complete proof can be found in the appendix.

Theorem 3 *A machine T is bounded-zigzag if and only if S_T is recognized by some deterministic pushdown automaton.*

3.2 Complexity of S_H

If we now adopt a point of view fixed on the tape $-S_H-$ rather than the head, a cycle in the subshift corresponds to a waiting time during which cell 0 does not change. We can adapt the previously built DPDA so that it recognizes exactly these waiting words between two visits of the head. The key point here is that these languages are unary, and unary context-free languages are regular (see for example [6]), and thus they can be recognized with a finite automaton.

When the machine is bounded-zigzag, the head can make at most one long cycle by side. The rest of the time, the head is either moving closer to or farther from cell 0, or staying in some finite window around cell 0. All of these behaviors can be recognized by a finite automaton, thus the language of S_H is regular. Therefore, we obtain a surprising reduction in language complexity when changing the point of view: if S_T is recognized by some DPDA, then S_H is sofic. The complete proof can be found in the appendix. Note that, up to a rescaling of the tape alphabet, all factor subshifts can be reduced to the case of S_H .

Theorem 4 *For any bounded-zigzag machine, all the factor subshifts of T_H are sofic.*

The converse of this theorem is false: we can construct a machine with a tape with n levels, where the head vertically shifts down the content of each level while moving right. It rebounds when it finds a wall in the lowest level (which is erased in the same way), and does the same in the opposite direction. We can see that the machine can make arbitrarily wide n -zigzags, each of independent length, in such a way that the factor subshifts of T_H are sofic.

Nevertheless, we can prove that this kind of construction is possible only with a bounded n . Let us introduce this formally.

Definition 4 *We say that a machine makes an n -cycle of width $N \in \mathbb{N}$ over configuration $x \in A^{\mathbb{Z}} \times Q \times \mathbb{Z}$ and cell $i \in \mathbb{Z}$, if there exist $2n + 1$ time steps*

$0 = t_0 < t_1 < \dots < t_{2n}$ such that the head is in position i at time t_{2q} and outside $\llbracket -N, N \rrbracket$ at time t_{2q+1} , for each $q \in \llbracket 0, n \rrbracket$. We say that the machine is n -bounded-cycle if there is some N such that the head cannot make n -cycles of width larger than N .

When S_T is sofic, the machine is 1-bounded cycle. Considering some machine T , we denote $\psi_N(x) \in \mathbb{N} \sqcup \{+\infty\}$ the maximum n such that the machine can make an n -cycle of width N over configuration x . Clearly, T is n -bounded cycle if and only if for some $N \in \mathbb{N}$, ψ_N is bounded by $n - 1$.

Let us call $\Phi_i(x)$ the set of time steps for which the head has position $i \in \mathbb{Z}$ when computing over configuration x . This set is linked to cycles by the following intuitive observation.

Proposition 2 *If T is an n -bounded-cycle machine, then there exists $p \in \mathbb{N}$ such that for any cell $i \in \mathbb{Z}$ and any non-preperiodic configuration $x \in X$, $|\Phi_i(x)| \leq p$.*

Proof. Let $n, N \in \mathbb{N}$ be such that $\max\{\psi_N(x) \mid x \in X\} = n - 1$, and $x \in X$ such that $|\Phi_0(x)| > p = 2n|A|^{2N+1}$ – the case $i \neq 0$ can be obtained by shifting. Consider $\{t_0, \dots, t_p\} \subset \Phi_0(x)$ with $t_0 < t_1 < \dots < t_p$. If we consider an $(n - 1)$ -cycle over x in cell 0, we can see that there exist $t_{k_1} < t_{k_2} < \dots < t_{k_{n-1}}$ such that for any $i \in \llbracket 1, n - 1 \rrbracket$, the head goes beyond N or $-N$ between time steps t_{k_i} and $t_{k_{i+1}}$, but not between (possibly equal) times $t_{k_{i+1}}$ and $t_{k_{i+1}}$. This means that t_{k_i} is the last time that the head is in 0 before going beyond $\llbracket -N, N \rrbracket$. Let $k_0 = -1$ and $k_n = p$, in such a way that $\llbracket 0, p \rrbracket = \bigcup_{i=0}^n I_i$, where $I_i = \llbracket k_i + 1, k_{i+1} \rrbracket$ for $0 \leq i \leq n$. There are $n + 1$ such intervals, so one of them, say I_i , has at least $|A|^{2N+1}$ elements; this is all the more the case for $\llbracket t_{k_{i+1}}, t_{k_{i+1}} \rrbracket \supset \{t_{k_j} \mid k_i < j \leq k_{i+1}\}$. Hence, between time steps $t_{k_{i+1}}$ and $t_{k_{i+1}}$ there are at least $|A|^{2N+1}$ consecutive time steps in $\Phi_0(x)$ such that the head stays within the interval of cells $\llbracket -N, N \rrbracket$. As a result, there are $i, j \in \llbracket t_{k_{i+1}}, t_{k_{i+1}} \rrbracket$ with $i < j$ and $T^i(x) = T^j(x)$, which implies that x is preperiodic. \square

Theorem 5 *If S_H is sofic, then T is n -bounded-cycle for some n .*

Proof. Assume that $S_H = \tau_H(X_H)$ is recognized by some finite automaton with N states, and that there exists some configuration $x \in X$ on which the machine makes some N -cycle of width N . Let t_0, \dots, t_{2N} be as in the definition of N -cycles, and $u = \tau_H(x)_{\llbracket 0, t_{2N} \rrbracket}$. Let $o_0 \dots o_{t_{2N}+1}$ be the corresponding path of the finite automaton. We can see that there are $i < j < N$ such that $o_{t_{2i}} = o_{t_{2j}}$, hence there is some periodic infinite word $z \in \tau_H(X_H)$ corresponding to the path w that repeats the cycle $(o_{t_{2i}} \dots o_{t_{2j}})$. From Proposition 1, z is isolated. As a consequence, w is the only path to start from $o_{t_{2i}}$. Therefore, its vertices are all different, and $t_{2j} - t_{2i} \leq N$, but in this case the head does not have the time to go beyond $\llbracket -N, N \rrbracket$ between these two iterations, which is a contradiction. We have proved that T is N -bounded-cycle. \square

Here, too, the converse is false, since it is easy to build a machine doing a given number of arbitrarily wide rebounds on specific wall characters before stopping. The language of such a machine cannot be regular because the time intervals between two rebounds are not independent.

3.3 Almost equicontinuity

We have already seen that in sofic machines, almost all configurations of X_H are equicontinuous. It is still so when allowing n -cycles, though in this case there are some configurations with head which are not equicontinuous – recall that Theorem 2 is an equivalence.

Theorem 6 *If T is an n -bounded-cycle machine for some n , then T_H is almost equicontinuous.*

Proof. By compactness of the space, it is enough to prove that for any cylinder $[u]$ and any $k \in \mathbb{N}$, there exist some $x \in [u]$ and some $m \in \mathbb{N}$ such that for any $y \in [x_{\llbracket -m, m \rrbracket}]$ and any $t \in \mathbb{N}$, $T_H^t(y) \in [T_H^t(x)_{\llbracket -k, k \rrbracket}]$. Let $N \in \mathbb{N}$ be as in the definition of n -bounded-cycle machine, $[u]$ a cylinder of X_H and $k \in \mathbb{N}$. If $[u]$ contains some preperiodic configuration with the head, then we can easily find m thanks to Remark 1. Otherwise, let us consider some configuration $x \in [u]$ (with the head) maximizing $|\Phi_{-k}(x) \sqcup \Phi_k(x)|$, which is finite thanks to Proposition 2. Let $m \in \mathbb{Z}$ be such that $m \geq k$ and the interval $\llbracket -m, m \rrbracket$ contains all the cells visited, when computing from x , up to time step $t = \max(\Phi_{-k}(x) \sqcup \Phi_k(x))$. Then we can see that any configuration $y \in [x_{\llbracket -m, m \rrbracket}]$ has the same evolution as x until this time step, and that after that, its head cannot visit cell $-k$ nor k , otherwise it would contradict the maximality of x . We can deduce that the head of x (then also y) is outside $\llbracket -k, k \rrbracket$ after iteration t , otherwise it would be trapped between $-k$ and k and would become periodic. We observe, then, that the cells of $\llbracket -k, k \rrbracket$ evolve exactly in the same way for configurations x and y . \square

The converse is untrue: imagine a machine whose head rebounds between two walls, each time shifting them to the left. Every configuration where the head starts enclosed between two walls is equicontinuous. Any finite pattern can be extended by adding walls to enclose the head, therefore equicontinuous points are dense, but the head can make an arbitrary number of arbitrarily wide cycles.

Conclusion

The complexity of the Turing machine will always be very hard to understand. In our attempt to treat this issue through the theories of topological and symbolic dynamics, we have found interesting relations between:

- the head movements that can be observed during the computation;
- the density of equicontinuous points;
- the language complexity of the associated subshifts S_T and S_H .

These relations introduce a new point of view on how computation is performed. In addition to generalizing them to more machines, the next step would be to study Turing machines as computing model by introducing a halting state, and to link all of these considerations to the result itself of the computation, and eventually the temporal or spatial complexity of the computation.

References

1. Kůrka, P.: On topological dynamics of Turing machines. *Theoret. Comput. Sci.* **174**(1-2) (1997) 203–216
2. Blondel, V.D., Cassaigne, J., Nichitiu, C.: On the presence of periodic configurations in Turing machines and in counter machines. *Theoret. Comput. Sci.* **289** (2002) 573–590
3. Oprocha, P.: On entropy and Turing machine with moving tape dynamical model. *Nonlinearity* **19** (October 2006) 2475–2487
4. Gajardo, A., Mazoyer, J.: One head machines from a symbolic approach. *Theoret. Comput. Sci.* **370** (2007) 34–47
5. Gajardo, A.: Sofic one head machines. In Durand, B., ed.: *Journées Automates Cellulaires*. (2008) 54–64
6. Ginsburg, S., Rice, H.: Two families of languages related to algol. *J. ACM* **9**(3) (1962) 350–371
7. Ogden, W.F.: A helpful result for proving inherent ambiguity. *Mathematical Systems Theory* **2**(3) (1968) 191–194

Proofs

The Ogden Lemma [7] is a well-know generalization to the case of pushdown automata of the pumping lemma on finite automata. It can be expressed on paths of the graph as follows.

Lemma 1 *Consider a DPDA $(A, \Omega, \Gamma, \perp, \lambda, o_0, F)$, and $(o_0, \mu^0) \dots (o_n, \mu^n)$ some path of its graph and $I \subset \llbracket 0, n \rrbracket$ a subset of distinguished positions of size $|I| > q = 2^{|\Omega|^2|\Gamma|^2+1}$. Then there exist four positions $0 \leq l_1 \leq l_2 < l_3 \leq l_4 \leq n$ and such that:*

1. $(o_{l_1}, \mu_0^{l_1}) = (o_{l_2}, \mu_0^{l_2})$;
2. $(o_{l_3}, \mu_0^{l_3}) = (o_{l_4}, \mu_0^{l_4})$;
3. $\forall i \in \llbracket l_1, l_4 \rrbracket, |\mu^i| \geq |\mu^{l_1}|$;
4. $\forall i \in \llbracket l_2, l_3 \rrbracket, |\mu^i| \geq |\mu^{l_2}|$;
5. $(o_0, \mu^0) \dots (o_{l_1}, \mu^{l_1})(o_{l_2+1}, \tilde{\mu}^{l_2+1}) \dots (o_{l_3}, \tilde{\mu}^{l_3})(o_{l_4+1}, \mu^{l_4+1}) \dots (o_n, \mu^n)$ is also a valid path of the graph, where $\tilde{\mu}^t = \mu^{l_1} \mu_0^{t-l_1}$;
6. $I \cap \llbracket l_2, l_3 \rrbracket \neq \emptyset$;
7. $|I \cap \llbracket l_1, l_4 \rrbracket| \leq q$;
8. Either $I \cap \llbracket 0, l_1 \rrbracket \neq \emptyset \neq I \cap \llbracket l_1, l_2 \rrbracket$ or $I \cap \llbracket l_3, l_4 \rrbracket \neq \emptyset \neq I \cap \llbracket l_4, n \rrbracket$.

If T is a machine with rule $\delta : A \times Q \rightarrow A \times Q \times \{-1, 0, 1\}$ and $\alpha, q \in A \times Q$, then we note $\delta_A(\alpha, q) = \beta$, $\delta_Q(\alpha, q) = p$ and $\delta_D(\alpha, q) = d$ if $\delta(\alpha, q) = (\beta, p, d)$. If $u = (\bar{u}, q, n) \in A^k \times Q \times \mathbb{Z}$, then we can define the corresponding cylinder in space X :

$$[u]_i = \{y, p, j \in A^{\mathbb{Z}} \times Q \times \mathbb{Z} \mid y \in [\bar{u}]_i \text{ and } p = q \text{ and } (n \in \llbracket i, i + |u| \rrbracket \Rightarrow j = n)\} .$$

Let ε denote the empty word.

Theorem 3 comes from the following lemmas.

Lemma 2 *Let N be a fixed natural number and T a Turing machine. Given two partial configurations $u = (\bar{u}, p, 0), v = (\bar{v}, q, k) \in A^{2N+1} \times Q \times \llbracket -N, N \rrbracket$, there exists a DFA $C_{u,v}$ that recognizes the language $\mathcal{C}_{u,v}$ of the words $(\tau_T \Psi(x))_{j=0}^t$ for $t \in \mathbb{N}$, $x \in [u]$ such that $T^t(x) \in [v]$ and for any $j \in \llbracket 0, t \rrbracket$ the head position of $T^j(x)$ is in $\llbracket -N, N \rrbracket$.*

Moreover, if x satisfies the conditions of $\mathcal{C}_{u,v}$, then every $y \in [u]$ also does, with the same time t .

The language $\mathcal{C}_{u,v}$ can be either empty, a singleton or, when v is periodic for T , infinite. The automaton $C_{u,v}$ simply simulates T by loading u on its memory, and making the partial configuration over cells $\llbracket -N, N \rrbracket$ evolve simply by applying the machine rule. The next lemma corresponds to similar and more evolved proof.

Lemma 3 *Let N be a fixed natural number and let T be a Turing machine that cannot do 1-zigzags of width N . If we have three partial configurations $u =$*

$(\bar{u}, p, N), v = (\bar{v}, q, k), u' = (\bar{u}', p', 0) \in A^{2N+1} \times Q \times \mathbb{N}$ such that $u_{\llbracket -N, 0 \rrbracket} = v_{\llbracket -N, 0 \rrbracket}$ and $\mathcal{C}_{u', u} \neq \emptyset$, then there exists a DPDA $R_{u, v}$ that recognizes the language $\mathcal{R}_{u, v}$ of the words $(\tau_T \Psi(x))_{j=0}^t$ for $t \in \mathbb{N}$, $x \in [u]$ such that $T^t(x) \in [v]$ and for any $j \in \llbracket 0, t \rrbracket$, the head position of $T^j(x)$ is strictly positive.

Moreover, if x satisfies the conditions of $\mathcal{R}_{u, v}$, then every y such that $y_{\llbracket 0, \infty \rrbracket} = x_{\llbracket 0, \infty \rrbracket}$ also does, with the same time t .

Symmetrically, if $u = (\bar{u}, p, -N), v = (\bar{v}, q, k), u' = (\bar{u}', p', 0) \in A^{2N+1} \times Q \times \mathbb{Z}_-$ such that $u_{\llbracket 0, N \rrbracket} = v_{\llbracket 0, N \rrbracket}$ and $\mathcal{C}_{u', u} \neq \emptyset$, then there exists a DPDA $L_{u, v}$ that recognizes the language $\mathcal{L}_{u, v}$ of the words $(\tau_T \Psi(x))_{j=0}^t$ for $t \in \mathbb{N}$, $x \in [u]$ such that $T^t(x) \in [v]$ and for any $j \in \llbracket 0, t \rrbracket$, the head position of $T^j(x)$ is strictly negative.

Moreover, if x satisfies the conditions of $\mathcal{L}_{u, v}$, then every y such that $y_{\llbracket -\infty, 0 \rrbracket} = x_{\llbracket -\infty, 0 \rrbracket}$ also does, with the same time t .

Proof. We will do the proof only for $\mathcal{R}_{u, v}$. The automaton registers the states of the tape and updates them at each step. The states of the cells at the right of the head will be registered in the internal state of the automaton, while the states of the cells at the left will be stocked in the stack. The position of the head is given by the stack depth; in this way the head is always reading the symbol w_0 .

We define actually an automaton in a slightly different model than previously defined. The initial and terminal states actually involve the content of the stack: we initially push a given finite word into the stack, and to accept a word, we verify if both the terminal state and the stack content are in some given finite sets. It is easy to see, by considering some complex encoding in the stack alphabet Γ , that this model can be simulated by the previous one. The automaton $R_{u, v}$ has input alphabet $A \times Q$, states set $\Omega = (A^{\leq N} \times Q) \sqcup \{REJECT\}$, stack alphabet $A^{\leq N}$; its initial state is $o = (\bar{u}_N, p)$ and initial stack content the mirror of $\bar{u}_{\llbracket 1, N \rrbracket}$; it terminates when the pair composed of the internal state and the stack content is in $F = \{((w, q), \mu) \mid v = (\mu w, q, |\mu| + 1)\}$; its transition function λ is defined by:

$$\lambda((\alpha, p), (w, q), \beta) = \begin{cases} ((w_{\llbracket 1, N \rrbracket}, \delta_Q(\alpha, q)), \delta_A(\alpha, p)\beta) & \text{if } \begin{cases} p = q \text{ and } \delta_D(\alpha, p) = 1 \\ w = \varepsilon \text{ or } w_0 = \alpha \end{cases} \\ ((\beta \delta_A(\alpha, p) w_{\llbracket 1, N-2 \rrbracket}, q'), \varepsilon) & \text{if } \begin{cases} p = q \text{ and } \delta_D(\alpha, p) = -1 \\ w = \varepsilon \text{ or } w_0 = \alpha \end{cases} \\ REJECT & \text{in any other case.} \end{cases}$$

Let us denote by $\mu^j \in A^*$ and $(w^j, q^j) \in A^{\leq N} \times Q$ the respectively stack content and internal state at iteration $j \in \mathbb{N}$.

- We will prove by induction on $j \in \mathbb{N}$, that if x satisfies the conditions of $\mathcal{R}_{u, v}$, then $T^j(x) \in [(\mu^j w^j, q^j, |\mu^j| + 1)]$.

For $j = 0$ it is clear, because $x \in [u] \subset [(\mu^0 w^0, q^0, N)]$. Let us suppose that it is true for a given $j \in \mathbb{N}$, and let us prove it for $j + 1$.

If $w^j \neq \varepsilon$, the head is reading the symbol w_0^j and is in state q^j , hence the only input accepted is (w_0^j, q^j) . In this case, the head will pass to state $\delta_Q(w_0^j, q^j)$

and will move to $\delta_D(w_0^j, q^j)$. If $\delta_D(w_0^j, q^j) = 1$ the automaton must push $\delta_A(w_0^j, q^j)$ and “erase” w_0^j . If $\delta_D(w_0^j, q^j) = -1$ the automaton must replace w_0^j by $\delta_A(w_0^j, q^j)$, pop a symbol and concatenate it to w^j .

If $w^j = \varepsilon$, the automaton will accept (α, p) only if $p = q^j$; in this case it will work, assuming that $w^j = \alpha$.

- Now we need to prove that every word recognized by $R_{u,v}$ is in fact in $\mathcal{R}_{u,v}$. We use recurrence to define the configuration x that certifies this. The first condition is that $x \in [(\bar{u}_N, p, N)]$, it follows from the first verification: $(\alpha, p) = (w_0^0, q^0) = (\bar{u}_N, p)$. Let us suppose that we have defined $x = (\bar{x}, p, N)$ such that $T^s(x) \in [(\mu^s w^s, q^s, |\mu^s| + 1)]$ for every $s \leq j$ and that the set of cells visited by the head is $\llbracket r, i \rrbracket$ for some $r < N$ and $i \geq |\mu^j w^j| + 1$. Let us prove that the same is true for $j+1$ for a suitable x' . We can note that the condition $T^j(y) \in [(\mu^j w^j, q^j, |\mu^j| + 1)]$ holds for any y satisfying $y_{\llbracket 0, i \rrbracket} = x_{\llbracket 0, i \rrbracket}$.

If $w^j = \varepsilon$, then the automaton will accept any pair (α, p) with $p = q^j$ if cell $k = |\mu^r| + 1$ has already been visited; the value of \bar{x}_k is important and cannot be defined to be α . But if k was visited, its value was registered in w^s for some s , and it has been erased because the head has moved to $k - N$ in some moment (then $k > N$). The existence of $u' = (\bar{u}', p', 0)$ such that $\mathcal{C}_{u',u} \neq \emptyset$ insures that the head has moved from 0 to N , which means that, the head has made a 1-zigzag to the right between cells $k - N$ and k , with is forbidden by hypothesis. Hence k has not been visited before ($i < k$) and we can define $x'_k = \alpha$.

When $w^j \neq \varepsilon$, we know that the value of cell k in $T^j(x)$ is w_0^j . The automaton will only accept the pair $(\alpha, p) = (w_0^j, q^j)$. This and the former construction insure that $T^{j+1}(x) \in [(\mu^{j+1} w^{j+1}, q^{j+1}, |\mu^{j+1}| + 1)]$. \square

Proof (of Theorem 3).

(\Rightarrow) Since S_T does not regards the head position, we can suppose that the head starts at 0. Let x be a configuration.

- If the head does not exit the interval $\llbracket -N, N \rrbracket$ during the whole evolution, then only M is needed to recognize $\tau_T \Psi(x)$, we conclude that $\tau_T \Psi(x)_{\llbracket 0, k \rrbracket} \in \mathcal{C}_{x_{\llbracket -N, N \rrbracket}, T^k(x)_{\llbracket -N, N \rrbracket}}$, for every $k \in \mathbb{N}$.
- If the head exits $\llbracket -N, N \rrbracket$ for the first time at iteration t_0 , by the right side, and never comes back to cell 0 after that, then $\tau_T \Psi(x)_{\llbracket 0, k \rrbracket} \in \mathcal{C}_{x_{\llbracket -N, N \rrbracket}, T^{t_0}(x)_{\llbracket -N, N \rrbracket}} \mathcal{R}_{T^{t_0}(x)_{\llbracket -N, N \rrbracket}, T^k(x)_{\llbracket -N, N \rrbracket}}$, for any k .
- If the head exits $\llbracket -N, N \rrbracket$ for the first time at iteration t_0 , by the right side, comes back to 0 at iteration t_1 , and never exit $\llbracket -N, N \rrbracket$ again, then $\tau_T \Psi(x)_{\llbracket 0, k \rrbracket} \in \mathcal{C}_{x_{\llbracket -N, N \rrbracket}, T^{t_0}(x)_{\llbracket -N, N \rrbracket}} \mathcal{R}_{T^{t_0}(x)_{\llbracket -N, N \rrbracket}, T^{t_1}(x)_{\llbracket -N, N \rrbracket}} \mathcal{C}_{T^{t_1}(x)_{\llbracket -N, N \rrbracket}, T^k(x)_{\llbracket -N, N \rrbracket}}$, for any k .
- If the head exits $\llbracket -N, N \rrbracket$ for the first time at iteration t_0 , by the right side, comes back to 0 at iteration t_1 , and exits $\llbracket -N, N \rrbracket$ again at t_2 and does not ever come back to 0, then $\tau_T \Psi(x)_{\llbracket 0, k \rrbracket}$ is in the concatenation of the languages $\mathcal{C}_{x_{\llbracket -N, N \rrbracket}, T^{t_0}(x)_{\llbracket -N, N \rrbracket}}$, $\mathcal{R}_{T^{t_0}(x)_{\llbracket -N, N \rrbracket}, T^{t_1}(x)_{\llbracket -N, N \rrbracket}}$, $\mathcal{C}_{T^{t_1}(x)_{\llbracket -N, N \rrbracket}, T^{t_2}(x)_{\llbracket -N, N \rrbracket}}$ and $\mathcal{L}_{T^{t_2}(x)_{\llbracket -N, N \rrbracket}, T^k(x)_{\llbracket -N, N \rrbracket}}$, for any k .

- If the head exits $\llbracket -N, N \rrbracket$ for the first time at iteration t_0 , by the right side, comes back to 0 at iteration t_1 , exits $\llbracket -N, N \rrbracket$ again at t_2 , and comes back to 0 at t_3 , then $\tau_T \Psi(x)_{\llbracket 0, k \rrbracket}$ is in the concatenation of $\mathcal{C}_{x_{\llbracket -N, N \rrbracket}, T^{t_0}(x)_{\llbracket -N, N \rrbracket}}$, $\mathcal{R}_{T^{t_0}(x)_{\llbracket -N, N \rrbracket}, T^{t_1}(x)_{\llbracket -N, N \rrbracket}}$, $\mathcal{C}_{T^{t_1}(x)_{\llbracket -N, N \rrbracket}, T^{t_2}(x)_{\llbracket -N, N \rrbracket}}$, $\mathcal{L}_{T^{t_2}(x)_{\llbracket -N, N \rrbracket}, T^{t_3}(x)_{\llbracket -N, N \rrbracket}}$ and $\mathcal{C}_{T^{t_3}(x)_{\llbracket -N, N \rrbracket}, T^k(x)_{\llbracket -N, N \rrbracket}}$, for any k .

The analogous case when the head first exits $\llbracket -N, N \rrbracket$ through cell $-N - 1$ can be treated in a similar way. We conclude that for any x and any k , the word $\tau_T \Psi(x)_{\llbracket 0, k \rrbracket}$ is in the language

$$\begin{aligned} & \bigcup_{u^0, w} \mathcal{C}_{u^0, w} \bigcup_{u^0, v^0, w} \mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, w} \bigcup_{u^0, v^0, u^1, w} \mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, u^1} \mathcal{C}_{u^1, w} \bigcup \\ & \bigcup_{u^0, v^0, u^1, b^1, w} \mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, u^1} \mathcal{C}_{u^1, b^1} \mathcal{L}_{b^1, w} \bigcup_{u^0, v^0, u^1, b^1, u^2, w} \mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, u^1} \mathcal{C}_{u^1, b^1} \mathcal{L}_{b^1, u^2} \mathcal{C}_{u^2, w} \bigcup \\ & \bigcup_{u^0, b^0, w} \mathcal{C}_{u^0, b^0} \mathcal{L}_{b^0, w} \bigcup_{u^0, b^0, u^1, w} \mathcal{C}_{u^0, b^0} \mathcal{L}_{b^0, u^1} \mathcal{C}_{u^1, w} \bigcup \\ & \bigcup_{u^0, b^0, u^1, v^1, w} \mathcal{C}_{u^0, b^0} \mathcal{L}_{b^0, u^1} \mathcal{C}_{u^1, v^1} \mathcal{R}_{v^1, w} \bigcup_{u^0, b^0, u^1, v^1, u^2, w} \mathcal{C}_{u^0, b^0} \mathcal{L}_{b^0, u^1} \mathcal{C}_{u^1, v^1} \mathcal{R}_{v^1, u^2} \mathcal{C}_{u^2, w} , \end{aligned}$$

where $u^i \in A^{2N+1} \times Q \times \{0\}$, $v^i \in A^{2N+1} \times Q \times \{N\}$, $b^i \in A^{2N+1} \times Q \times \{-N\}$, and $w \in A^{2N+1} \times Q \times \llbracket -N, N \rrbracket$. This language is recognizable by a DPDA since it is a concatenation and union of languages which are recognizable by DPDAs, thanks to Lemmas 2 and 3.

We have to prove now that this union of languages contains only words of $\mathcal{L}(S_T)$. The proof is similar for each of the listed languages; we will develop it only for $\mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, u^1} \mathcal{C}_{u^1, b^1} \mathcal{L}_{b^1, w}$.

From Lemma 2, we know that if $\mathcal{C}_{u^0, v^0} \neq \emptyset$, then any $x \in [u^0]$ will satisfy $\tau_T \Psi(x)_{\llbracket 0, t_0 \rrbracket} \in \mathcal{C}_{u^0, v^0}$ if the head position at time t_0 is N . From Lemma 3, if $\mathcal{R}_{v^0, u^1} \neq \emptyset$, then there exists $y \in [v^0]$ and t_1 such that $T^{t_1}(y) \in [u^1]$ and $\tau_T \Psi(x)_{\llbracket t_0, t_1 \rrbracket} \in \mathcal{R}_{v^0, u^1}$. We define $x_{\llbracket N, \infty \rrbracket} = y_{\llbracket N, \infty \rrbracket}$, which will satisfy $\tau_T \Psi(x)_{\llbracket 0, t_1 \rrbracket} \in \mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, u^1}$. From the same lemmas, we know that the values of x on $\llbracket -\infty, -N \rrbracket$ are still “free” and $T^{t_1}(x) \in [u^1]$ gives $\tau_T \Psi(x)_{\llbracket t_1, t_2 \rrbracket} \in \mathcal{C}_{u^1, b^1}$, where t_2 is the instant in which the head reaches the cell $-N$ for the first time.

We can suppose that $\mathcal{L}_{b^1, w}$ is not empty – otherwise the result is trivial. Then there exists y' such that $T^k(y') \in [w]$ and $\tau_T \Psi(x)_{\llbracket t_2, k \rrbracket} \in \mathcal{L}_{b^1, w}$. The values of y' over $\llbracket -N, \infty \rrbracket$ are not important and we can fix them to those of $T^{t_2}(x)$, or in other words, to define $x_{\llbracket -\infty, -N \rrbracket} = y'_{\llbracket -\infty, -N \rrbracket}$. We obtain

$$\tau_T \Psi(x)_{\llbracket 0, k \rrbracket} \in \mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, u^1} \mathcal{C}_{u^1, b^1} \mathcal{L}_{b^1, w} .$$

This completes the proof.

(\Leftarrow) Let us assume that the language of S_T is recognized by some DPDA M , that p is as in Lemma 1, and that the machine can do a 1-zigzag of width $N = p + 3$; we can easily find some configuration x with time steps $0 < t_1 < t_2 < t_3$ such that the machine visits cell 1 at time 0, cell N at times t_1 and t_3 , and cell

0 at time t_2 . It can also be assumed that the zigzag is minimal, in the sense that no other configuration satisfies the condition with a lower t_3 . Moreover, we can assume that t_1 is the last time when cell N is visited before t_3 , and t_2 is the first time when cell 0 is visited. Note that $t_2 - t_1 \geq N$. Let $c = (o_0, \mu^0) \dots (o_{t_3}, \mu^{t_3})$ the corresponding path in the graph of M .

The key point of the proof is that, thanks to the determinism of the automaton, given $w \in \mathcal{L}(S_T)$, the i -th cell is visited by the head for the first time if and only if the corresponding vertex in the graph of M has out-degree more than 1. Since the out-degree of a vertex (q, u) of M depends only on (q, u_0) . Let V be the set of vertices with out-degree 1, and L be the subset of V corresponding to vertices whose unique out-neighbor is not in V and such that this unique transition corresponds to a left movement of the head. These vertices represent cells which are at the left extremity of some visited zone. For instance, note that the vertices $(o_{t_1+1}, \mu^{t_1+1}), \dots, (o_{t_2-1}, \mu^{t_2-1})$ are in V since the corresponding visited cells are between 1 and N , and (o_{t_2-1}, μ^{t_2-1}) is the first vertex of the path c to belong to L .

If we apply Lemma 1 with $I =]]t_1, t_2 - 1[[$, we obtain time steps $0 \leq l_1 \leq l_2 < l_3 \leq l_4 < t_3$ such that

$$\tilde{c} = (o_0, \mu^0) \dots (o_{l_1}, \mu^{l_1})(o_{l_2+1}, \tilde{\mu}^{l_2+1}) \dots (o_{l_3}, \tilde{\mu}^{l_3})(o_{l_4+1}, \mu^{l_4+1}) \dots (o_{t_3}, \mu^{t_3})$$

is a valid path in the graph of M , *i.e.* it can be obtained from some configuration \tilde{x} , which we can suppose to have the head in cell 1 without loss of generality.

- First, suppose $l_4 \geq t_2 - 1$. Since $|I \cap]]l_1, l_4[[| \leq p$, we must have $l_1 > t_1$. Moreover, the nonemptiness of $I \cap]]l_2, l_3[[$ gives $t_1 < l_1 \leq l_2 < t_2 - 1 \leq l_4 \leq t_3$. The vertices of $d = (o_{l_1}, \mu^{l_1}) \dots (o_{t_2-1}, \mu^{t_2-1})$ are in V , then d is the only subpath of this length starting at (o_{l_1}, μ^{l_1}) . Thus, $(o_{l_1}, \mu^{l_1})(o_{l_2+1}, \tilde{\mu}^{l_2+1}) \dots (o_{t_2-1-l_2+l_1}, \tilde{\mu}^{t_2-1-l_2+l_1}) = d$. In particular, $(o_{t_2}, \tilde{\mu}^{t_2}) = (o_{t_2-l_2+l_1}, \mu^{t_2-l_2+l_1})$. From the lemma, $(o_{t_2}, \tilde{\mu}^{t_2}) = (o_{t_2}, \mu_0^{t_2})$; the same automaton rule is applied in both vertices, and since (o_{t_2}, μ^{t_2}) is not in V , we conclude that $(o_{t_2-l_2+l_1}, \mu^{t_2-l_2+l_1}) \notin V$. This results in $l_1 = l_2$, and from the Ogden Lemma we get $l_3 < l_4 < t_2$, which is a contradiction.
- Now suppose that $l_4 < t_2$. As no vertex of \tilde{c} is in L before (o_{t_2-1}, μ^{t_2-1}) , we can see that, in this path too, the vertex (o_{t_2}, μ^{t_2}) corresponds, at time $t_2 - l_4 + l_3 - l_2 + l_1$, to the first visit of cell 0 – at the first time we go more to the left than the visited zone. Since both paths coincide after that, the head does the same movements, and we obtain that its position at the last vertex (o_{t_3}, μ^{t_3}) of path \tilde{c} is N . But, from path c , we know that $(o_{t_3}, \mu^{t_3}) \in V$, so on \tilde{x} too the machine had already visited cell N before arriving on this vertex. It could not be after time $t_2 - l_4 + l_3 - l_2 + l_1$, since from then on we have followed the same positions as in c , hence \tilde{c} represents a 1-zigzag; from the last point of the lemma, it is shorter than c , so \tilde{x} satisfies the construction hypotheses of x but contradicts its minimality. \square

Proof (of Theorem 4). Let us define the following languages.

$$\begin{aligned}\overline{\mathcal{R}_{u,v}} &= \{u_0^k \mid \exists w \in \mathcal{R}_{u,v}, |w| = k\} \\ \overline{\mathcal{L}_{u,v}} &= \{u_0^k \mid \exists w \in \mathcal{L}_{u,v}, |w| = k\}\end{aligned}$$

It is a context-free language since it is the transformation of a context-free language through a letter morphism. It is also a regular language because it uses a single symbol u_0 . If x and t satisfy the conditions of $\mathcal{R}_{u,v}$, then $\tau_H \Phi(x)_{\llbracket 0, t-1 \rrbracket} \in \overline{\mathcal{R}_{u,v}}$.

We also define the language $\mathcal{C}_{u,v}$ of the words $\tau_H \Phi(x)_{\llbracket 0, t \rrbracket}$ with $t \in \mathbb{N}$ and $x \in [u^0]$ such that $T^t(x) \in [u^1]$ and for any $j \in \llbracket 0, t \rrbracket$, the head of $T^j(x)$ is in $\llbracket -N, N \rrbracket$.

It is recognized by an automaton that simulates M and accepts a pair (α, p) if and only if the current head position is 0, and p and α match the simulation.

If the head starts at cell 0, the analogous concatenation and union of the $\overline{\mathcal{C}}$ s, $\overline{\mathcal{R}}$ s and $\overline{\mathcal{L}}$ s would represent $\mathcal{L}(S_H)$. But if the head does not start at 0, we need to consider, for $u = (\overline{u}, p, 0) \in A^{2N+1} \times Q \times \{0\}$, the language \mathcal{B}_u of the words \overline{u}_0^t for which there exists x with $T^t(x) \in [u]$ and for any $j < t$, the head of $T^j(x)$ is not in cell 0. \mathcal{B}_u represents the set of sequences of states observed at cell 0 until the head reaches it, when the partial configuration u is observed in $\llbracket -N, N \rrbracket$. \mathcal{B}_u is always a nonempty “interval”, *i.e.* $\mathcal{B}_u = \{u_0^t \mid 0 \leq t \leq n\}$ for some $n \in \mathbb{N}$ which may be 0 – if u is a “garden of Eden”.

Since \mathcal{B}_u is either finite or equal to $\{u_0^t \mid t \in \mathbb{N}\}$, it can be recognized with a DFA B_u . $\mathcal{L}(S_H)$ will be the concatenation and union of \mathcal{B} s and the other languages.

Globally, we obtain that $\tau_F \Phi(x)_{\llbracket 0, k \rrbracket}$ is in the following union:

$$\begin{aligned}& \{x_0^k\} \cup \bigcup_{u^0, w} \mathcal{B}_{u^0} \overline{\mathcal{C}_{u^0, w}} \cup \bigcup_{u^0, v^0, w} \mathcal{B}_{u^0} \overline{\mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, w}} \cup \bigcup_{u^0, v^0, u^1, w} \mathcal{B}_{u^0} \overline{\mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, u^1} \mathcal{C}_{u^1, w}} \cup \\ & \bigcup_{u^0, v^0, u^1, b^1, w} \mathcal{B}_{u^0} \overline{\mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, u^1} \mathcal{C}_{u^1, b^1} \mathcal{L}_{b^1, w}} \cup \bigcup_{u^0, v^0, u^1, b^1, u^2, w} \mathcal{B}_{u^0} \overline{\mathcal{C}_{u^0, v^0} \mathcal{R}_{v^0, u^1} \mathcal{C}_{u^1, b^1} \mathcal{L}_{b^1, u^2} \mathcal{C}_{u^2, w}} \cup \\ & \bigcup_{u^0, b^0, w} \mathcal{B}_{u^0} \overline{\mathcal{C}_{u^0, b^0} \mathcal{L}_{b^0, w}} \cup \bigcup_{u^0, b^0, u^1, w} \mathcal{B}_{u^0} \overline{\mathcal{C}_{u^0, b^0} \mathcal{L}_{b^0, u^1} \mathcal{C}_{u^1, w}} \cup \\ & \bigcup_{u^0, b^0, u^1, v^1, w} \mathcal{B}_{u^0} \overline{\mathcal{C}_{u^0, b^0} \mathcal{L}_{b^0, u^1} \mathcal{C}_{u^1, v^1} \mathcal{R}_{v^1, w}} \cup \bigcup_{u^0, b^0, u^1, v^1, u^2, w} \mathcal{B}_{u^0} \overline{\mathcal{C}_{u^0, b^0} \mathcal{L}_{b^0, u^1} \mathcal{C}_{u^1, v^1} \mathcal{R}_{v^1, u^2} \mathcal{C}_{u^2, w}} . \square\end{aligned}$$