



Thin and Thick Timed Regular Languages

Nicolas Basset, Eugene Asarin

► **To cite this version:**

Nicolas Basset, Eugene Asarin. Thin and Thick Timed Regular Languages. FORMATS, Sep 2011, Netherlands. pp.113-128, 10.1007/978-3-642-24310-3_9 . hal-00793583

HAL Id: hal-00793583

<https://hal-upec-upem.archives-ouvertes.fr/hal-00793583>

Submitted on 24 Jul 2014

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.

Thin and thick timed regular languages

Nicolas Basset¹ and Eugene Asarin²

¹ LIGM, Université Paris-Est and CNRS, nbasset@dptinfo.ens-cachan.fr

² LIAFA, Université Paris Diderot and CNRS, asarin@liafa.jussieu.fr

Abstract. In previous literature on timed automata, it was noticed that they are in several aspects too precise, which leads sometimes to strange artifacts, mathematical pathologies or unrealistic models. In particular, some timed automata are non-implementable, non-robust, behave badly under discretization, have many Zeno runs etc. In this paper, we propose a unifying approach to most of these issues for deterministic timed automata. We classify these automata either as **thin** or as **thick**. In thin automata, all the infinite trajectories are, in some weak sense, Zeno; the discretization of long trajectories is difficult, since it requires very small discretization step. In thick automata, most of trajectories are non-Zeno and behave well under discretization; such automata satisfy a sort of pumping lemma. Formally, the thin-thick alternative is based on the notion of entropy of timed regular languages introduced by E. Asarin and A. Degorre in [3, 4]. Thin languages have the entropy $= -\infty$ while thick have a larger one. An important application of thin-thick alternative is again the entropy theory of timed languages. We show that the entropy can be computed with a desired precision using discretization and thus it is computable, which closes a question left open in [3, 4].

1 Introduction

Timed automata [2] using exact continuous clocks, exact guards and resets are a beautiful mathematical object and a useful model of real-time systems. However, from the very beginning of the timed automata research, it was clear that they are in several aspects too precise, which leads sometimes to strange artifacts, mathematical pathologies or unrealistic models. Several lines of research have partially elucidated these issues.

Thus, the state space of a timed automaton being infinite, some long (or infinite) runs never revisit the same state. For this reason, as stated in [7], usual **pumping lemmata** do not hold, and should be replaced by rather involved analogues. In a run, infinitely many events can happen during a finite amount of time, or two events can happen again and again with the time interval between them tending to 0. Such a run reminds of Zeno's aporias and is often called a **Zeno run**, see [12] and reference therein. Pathological runs do not support well discretization of clocks, see [14, 6].

In order to rule out bad behaviors, restricted classes of timed automata and alternative semantics were considered by several authors. Thus, in [13, 15], a **tube language** semantics is introduced. In [19] a **robust semantics**, based

on small imprecisions is considered. It reappears in a different flavor as **implementability**, see [22, 21], and in another version in [1].

With the same objective to rule out bad behaviors, restrictions are often put on all the cycles in the automaton, by requiring that each cycle takes at least one time unit (strongly non-Zeno condition), or resets all the clocks (progress cycle condition), or even resets all the clocks at one and the same transition (regeneration or synchronization condition).

In this paper, we propose a unifying approach to most of these issues for deterministic timed automata. We classify each automaton either as *thin* or as *thick* (the classification is decidable).

In thin automata all the infinite trajectories, are, in some weak sense Zeno; the digitization of long trajectories is difficult, since it requires very small discretization step.

In thick automata most of trajectories are non-Zeno, behave well under digitization and satisfy a sort of pumping lemma.

The main technical tool used to characterize thin and thick timed languages is their (volumetric) entropy introduced in [3, 4]. Let us briefly recall this notion: a timed language L , for a given number of events n , can be seen as several polytopes in \mathbb{R}^n , their total volume is denoted by V_n . In most cases, for $n \rightarrow \infty$ this volume behaves exponentially: $V_n \approx 2^{n\mathcal{H}}$. The growth rate (i.e. \mathcal{H}) is referred to as *entropy*, and characterizes the size and the information contents of the timed language. With the notion of entropy, the definition of thin and thick languages is simple: a language is thin if its entropy equals $-\infty$ (that is the volume V_n decays faster than any exponent), and thick otherwise.

We identify a novel notion of a forgetful cycle, that is a cyclic path allowing forgetting the clock values. We state that any path in a timed automaton which is thick and long (in some precise sense), necessarily contains a forgetful cycle (Thm. 1), which can be seen as a weak version of pumping lemma. Based on this pumping lemma, we obtain our first main result (Thm. 2): thickness of a language is equivalent to many other nice properties briefly described above (good discretization, existence of forgetful cycle etc.).

The proof of Thms. 1-2 is rather technical, and uses together with “timed” techniques inspired by [19, 1], the monoid version of Ramsey theory, namely Simon factorization forests [20].

The thin-thick alternative leads to a more precise analysis of a timed automaton when applied to its strongly connected components. In general, a timed automaton can be decomposed into several strongly connected components (some of them are thin, others thick) and acyclic pathways between them. We show that most of the long enough runs spend most of the time in thick components, and only few pathological runs wander in thin components (Thm. 3).

Finally, we apply the thin-thick alternative to the analysis of entropy of timed languages. In [4] it was shown that the entropy of a timed language can be lower and upper bounded using entropies of two discrete languages L_ε^- and L_ε^+ , corresponding to a shrunk and a bloated discretizations of L :

$$h(L_\varepsilon^-) + \log(\varepsilon) \leq \mathcal{H} \leq h(L_\varepsilon^+) + \log(\varepsilon).$$

Here (Thm. 4) we strengthen this result and establish that on a thick component the entropy of a timed language can be approximated with a good precision using the entropy $h(L_\varepsilon)$ of the discretized language:

$$\mathcal{H} = h(L_\varepsilon) + \log(\varepsilon) + o(1).$$

As a corollary we obtain a converging algorithm allowing computation of \mathcal{H} for any timed regular language with any precision required. This answers the open question from [4]: \mathcal{H} is always a computable real number.

Paper organization. In Sect. 2, we recall some basic definitions, define thin and thick languages and give some motivating examples. In Sect. 3 we describe three more involved constructions: polytopes associated to paths in timed automata as in [18], region split automaton as in [3] and monoid of orbit graphs inspired by [19]. In the central Sect. 4 we state the thin-thick alternative for timed automata and a sort of pumping lemma for thick automata (Thm. 1). In Sect. 5 we apply these results to entropy of timed languages. We conclude with some perspectives in Sect. 6.

2 Preliminaries

2.1 Timed languages and their measures

A *timed word* $\alpha = (t_1, a_1) \dots (t_n, a_n)$ is a word on the alphabet $\mathbb{R}^+ \times \Sigma$ where Σ is a finite alphabet of *events*. Times t_i represent *delays* between events a_{i-1} and a_i . Throughout this paper, delays will be bounded³ by an integer constant M . We will sometimes write the same timed word α as (\mathbf{t}, w) with $\mathbf{t} \in [0, M]^n$ and $w \in \Sigma^n$. A timed language L is a set of timed words. We will denote by L_n the language L restricted to words of length n .

Volume and volumetric entropy. Let L be a timed language. For each word of events $w \in \Sigma^n$, let $L(w)$ be the set $\{\mathbf{t} \in [0, M]^n \mid (\mathbf{t}, w) \in L\}$. This subset of \mathbb{R}^n has a volume⁴ (Lebesgue's measure) denoted by $\mathbf{Vol}(L(w))$. The *volume* of L_n is $\mathbf{Vol}(L_n) = \sum_{w \in \Sigma^n} \mathbf{Vol}(L(w))$. Bounding the delays by M leads to a volume bounded by $(|\Sigma|M)^n$.

In this paper, we will work with factor closed languages⁵, i.e. such that if $(t_1, a_1) \dots (t_n, a_n) \in L$ then for all $1 \leq i \leq j \leq n$ $(t_i, a_i) \dots (t_j, a_j) \in L$. For each couple of words $w_1, w_2 \in \Sigma^*$ the language inclusion $L(w_1 \cdot w_2) \subseteq L(w_1)L(w_2)$ holds, and then $\mathbf{Vol}(L(w_1 \cdot w_2)) \leq \mathbf{Vol}(L(w_1))\mathbf{Vol}(L(w_2))$. The volumetric entropy of a timed language is a number in $[-\infty, +\infty)$ defined as

$$\mathcal{H}(L) = \lim_{n \rightarrow +\infty} \frac{1}{n} \log_2(\mathbf{Vol}(L_n)).$$

³ Our approach to timed languages is based on volumes, and does not apply, in its present form, to unbounded delays which lead to infinite volumes.

⁴ under the condition that the set is measurable; timed languages considered in this paper are all measurable as unions of polytopes.

⁵ This roughly corresponds to automata where all states are both initial and final.

The limit (finite or $-\infty$) exists due to subadditivity of $\log_2 \text{Vol}(L_n)$ wrt n and is upper bounded by $\log_2(|\Sigma| M)$. A timed language L is called *thin* if $\mathcal{H}(L) = -\infty$ and *thick* otherwise.

To deal with finite objects, one can discretize all the previous continuous languages. Given an $\varepsilon = \frac{1}{N}$, ($N \in \mathbb{N}$) called *discretization step*, let us define $L_{\varepsilon,n} = L_n \cap (\varepsilon\mathbb{N} \times \Sigma)^n$, i.e. the set of words in L_n with all delays multiple of ε , $L_\varepsilon = \bigcup_{n \in \mathbb{N}} L_{\varepsilon,n}$, $L_\varepsilon(w) = L(w) \cap (\varepsilon\mathbb{N})^n$. The ε -entropy h_ε is defined as

$$h_\varepsilon(L) = \lim_{n \rightarrow \infty} \frac{1}{n} \log_2(|L_{\varepsilon,n}|).$$

We will relate the discrete entropy h_ε to the continuous one \mathcal{H} in Sect. 5.

Open, closed and punctual languages. A language is said to be *open* if for all n , the set L_n is an open subset (for the product topology) of $\mathbb{R}^n \times \Sigma^n$. In other words, L_n should be a finite union of $O \times \{w\}$ where w is a word of events and O an open subset of \mathbb{R}^n . Interior and closure of L , denoted $\text{int}(L)$ and \overline{L} are defined in a natural way.

Taking closure or interior of a finite union of polytopes of \mathbb{R}^n has no effect on its volume. Nevertheless the number of discrete points can drastically change if punctuality is allowed (see [5]). We call a language to be *punctual* if $\text{int}(\overline{L}) \neq \overline{L}$.

2.2 Timed automata and their languages

Clocks, zones and regions. Let X be a finite set of variables called *clocks*. Clocks have non negative values bounded by a constant M_c . A rectangular constraint is a formula of the form $x \sim c$ where $x \in X$, $c \in \mathbb{N}$, $\sim \in \{\leq, <, =, >, \geq\}$. A diagonal constraint is a formula of the form $x - y \sim c$ where $x, y \in X$. A guard is a finite conjunction of rectangular constraints. We denote by G the set of all guards. A *zone* is a set of clock vectors $\mathbf{x} \in [0, M_c]^X$ satisfying a finite conjunction of rectangular and diagonal constraints. A *region* is an inclusion-minimal zone. A region (which is always a simplex) is uniquely defined by a point with integer coordinates $\lfloor \mathbf{x} \rfloor \in \{0, \dots, M_c\}^X$ giving integer part of clocks and an order on the fractional part of clocks $0 \sim_0 \{x_{i_1}\} \sim_1 \{x_{i_2}\} \sim_2 \dots \sim_{|X|} 1$ where $\sim_0, \dots, \sim_{|X|} \in \{<, =\}$. The closure of a region (abusively called closed region) can be obtained by replacing $<$ by \leq and $>$ by \geq in its definition.

Timed automata. A *timed automaton* is a tuple $(Q, X, \Sigma, E, \mathcal{L}, I, F)$ with Q a finite set of *locations*; X a set of bounded clocks; $E \subseteq Q \times G \times 2^X \times Q$ a finite set of edges; $\mathcal{L} : E \rightarrow \Sigma$ a labeling function on edges; $I \subseteq Q \times [0, M_c]^X$ the set of initial states; $F \subseteq Q \times [0, M_c]^X$ the set of final states.

By default all the *states* (elements of $Q \times [0, M_c]^X$) are initial and final, otherwise they are given by union of zones.

The clocks grow with the same (unit) speed and some of them are reset to 0 when passing through an edge. More formally, there is a *transition* $(q, \mathbf{x}) \xrightarrow{(t,a)} (q', \mathbf{x}')$ if there is an edge $e = (q, \mathbf{g}, R, q') \in E$ with $\mathcal{L}(e) = a$

such that $\mathbf{x} + (t, \dots, t)$ satisfies the guard \mathbf{g} and for each clock $x \in X$, its new value $x' = 0$ iff $x \in R$, or $x' = x + t$ otherwise. A *run* on the timed word $\alpha = (t_1, a_1) \dots (t_n, a_n)$ is a sequence of consecutive transitions $(q_0, \mathbf{x}_0) \xrightarrow{(t_1, a_1)} (q_1, \mathbf{x}_1) \dots \xrightarrow{(t_n, a_n)} (q_n, \mathbf{x}_n)$, where $\mathbf{x}_0, \dots, \mathbf{x}_n \in [0, M]^X$, $q_0, \dots, q_n \in Q$. A timed word is recognized by the automaton if there exists a run on it from an initial state to a final state. The timed language $L(\mathcal{A})$ consists of all the recognized words. We will be interested in its entropy $\mathcal{H}(L(\mathcal{A}))$, that will be abusively denoted $\mathcal{H}(\mathcal{A})$.

We call a TA *right resolving* if any two edges leaving the same location and having the same label have disjoint guards. Adding the condition that there is only one initial state gives the usual definition of determinism. In the rest of the paper, we work with right resolving TA.

Paths and reachability relation. We call a *path* in an automaton any sequence of edges. The “useful” paths are sequences of consecutive edges (such that the starting location of the $(i+1)^{th}$ edge is the ending one of the i^{th}), but we allow arbitrary words of E^* and all objects associated to a non-consecutive sequence will be empty.

Given two clock vectors \mathbf{x}, \mathbf{x}' , a path $\pi \in E^n$ and a sequence of delays of the same length $\mathbf{t} = (t_1, \dots, t_n)$, we write that $\mathbf{x} \xrightarrow{\mathbf{t}, \pi} \mathbf{x}'$ whenever exists a run in the automaton of the form $(q_0, \mathbf{x}) \xrightarrow{(t_1, a_1)} (q_1, \mathbf{x}_1) \dots \xrightarrow{(t_n, a_n)} (q_n, \mathbf{x}')$ following the sequence of edges π .

Several objects are naturally associated with a path. Given a path and two clock vectors, a language (a polytope of all the timings of the path) can be defined: $L(\pi, \mathbf{x}, \mathbf{x}') = \{\mathbf{t} \mid \mathbf{x} \xrightarrow{\mathbf{t}, \pi} \mathbf{x}'\}$. If we are not interested in clock values, we get a polytope depending only on the path: $L(\pi) = \{\mathbf{t} \mid \exists \mathbf{x}, \mathbf{x}', \mathbf{x} \xrightarrow{\mathbf{t}, \pi} \mathbf{x}'\}$. The other way around, if we do not care about timing, we get the reachability predicate: $\text{Reach}(\pi) = \{(\mathbf{x}, \mathbf{x}') \mid \exists \mathbf{t}, \mathbf{x} \xrightarrow{\mathbf{t}, \pi} \mathbf{x}'\}$.

A path $\pi \in E^n$ is said to be *punctual* if $L(\pi)$ is not empty but has dimension less than n . The closure (resp. interior) of a path π denoted by $\bar{\pi}$ (resp. $\text{int}(\pi)$) is the path obtained from π by taking closure (resp. interior) of all guards of edges in π (i.e. changing strict inequalities in non-strict ones (resp. vice versa)). There is a *limit cycle* (resp. *strong limit cycle*) along π if there exists a clock vector \mathbf{x} and time sequence \mathbf{t} such that $\mathbf{x} \xrightarrow{\mathbf{t}, \bar{\pi}} \mathbf{x}$ (resp. $\mathbf{x} \xrightarrow{\mathbf{t}, \text{int}(\pi)} \mathbf{x}$). Given $\varepsilon > 0$, in ε -discrete limit cycles all the components of \mathbf{x} and \mathbf{t} should be multiple of ε .

2.3 Thinness, simplices and examples

Our analysis of thin languages will start with a simple observation that the volume of k -dimensional simplices tends to 0 faster than any exponent:

Lemma 1. *The volume of a simplex of “type 1” described by inequalities $0 \leq t_1 + \dots + t_k \leq 1$, $t_i \geq 0$ or of a simplex of “type 2” described by inequalities $0 \leq t_1 \leq \dots \leq t_k \leq 1$ is $\frac{1}{k!}$.*

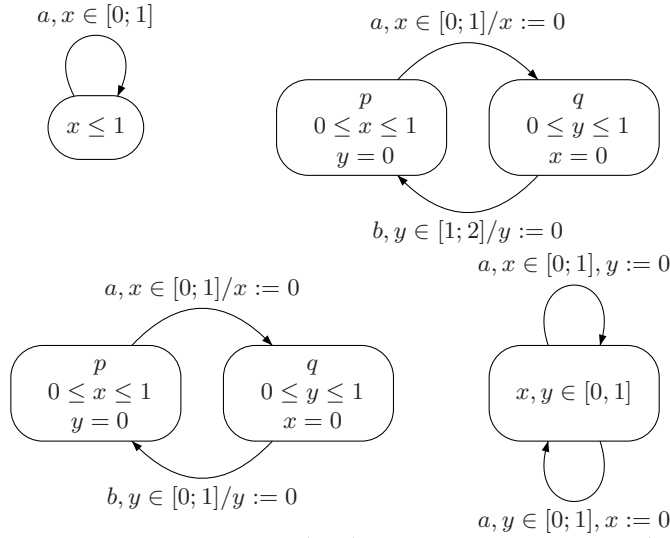


Fig. 1. First row: thin automata $\mathcal{A}_1, \mathcal{A}_2$. Second row: thick ones $\mathcal{A}_3, \mathcal{A}_4$. Initial states are given by conditions in nodes

By change of coordinates the lemma can be extended to more general polytopes:

Corollary 1. *Let P be a subset of $\{t_1, \dots, t_n \mid 0 \leq t_i \leq M\}$. If there exists a subsequence of indices $s(1), \dots, s(k)$ of $1, \dots, n$ and new coordinates $u_{s(i)}$ functions of $t_{s(1)}, \dots, t_{s(i)}$ with $0 \leq u_{s(1)} \leq u_{s(2)} \leq \dots \leq u_{s(k)} \leq 1$ and $\left| \frac{\partial u_{s(i)}}{\partial t_{s(i)}} \right| \geq 1$ then $\text{Vol}(P) \leq \frac{M^{n-k}}{k!}$.*

The automata on Fig. 1 illustrate the concepts of thin and thick. $L_n(\mathcal{A}_1) = \{t_1, \dots, t_n \mid \sum_{i \leq n} t_i \leq 1\}$ is a simplex of type 1, and thus $L(\mathcal{A}_1)$ is thin. $L_n(\mathcal{A}_2, q) = \{t_1, \dots, t_n \mid \forall i, t_{2i} + t_{2i+1} \leq 1 \wedge t_{2i+1} + t_{2i+2} \geq 1\}$, by interchanging even and odd indices we obtain $L_n(\mathcal{A}_2, p)$. Posing $u_{2i+1} = 1 - t_{2i+1}$ and $u_{2i} = t_{2i}$ yields a simplex $0 \leq u_1 \leq \dots \leq u_n \leq 1$. This change of coordinates preserves volume and so $\text{Vol}(L_n(\mathcal{A}_2, q)) = \frac{1}{n!}$. This is an example of automaton satisfying the *progress cycle condition* (i.e. resetting all clocks along each cycle) and nevertheless thin.

Third and fourth examples are thick, their entropies can be computed symbolically because they are $1\frac{1}{2}$ clock (see [3]), they are respectively $\log_2 \frac{2}{\pi}$ and $\log_2 \log_2(e)$. Note that \mathcal{A}_4 does not satisfy the progress cycle condition.

3 More on paths and cycles

3.1 Region graph and state split automata

Timed variants of the region graph [2] are extensively used in the literature. Here we use so-called region-split automaton given in [3], add several new conditions and modify those concerning initial states. A timed automaton is in region-split form if

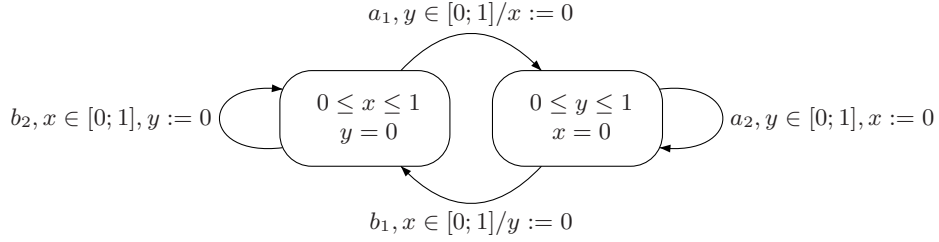


Fig. 2. The closed region-split version of \mathcal{A}_4

- B1. For every location $q \in Q$ a unique region \mathbf{r}_q (called its *entry region*) exists, such that the set of clock values with which q is entered is exactly \mathbf{r}_q .
- B2. The guard \mathbf{g} of every transition $\delta = (q, \mathbf{g}, R, q') \in E$ is just one region.
- B3. All the states of entry regions (and only these states) are both initial and final.
- B4. For any location there exists a path leading to some cycle and a path coming from some cycle.
- B5. For every transition δ its guard \mathbf{g} has no constraints of the form $x = c$ in its definition.
- B6. The labeling function on edges is identity (and so every two distinct edges have different labels).

Proposition 1. *Given a right resolving TA \mathcal{A} with bounded clocks and all states initial and final, one can construct an automaton $\text{RS}(\mathcal{A})$ called the region-split automaton of \mathcal{A} which satisfies B1 – B6 and such that $\mathcal{H}(\text{RS}(\mathcal{A})) = \mathcal{H}(\mathcal{A})$.*

In the following, we replace w.l.o.g. for the computing of \mathcal{H} , \mathcal{A} by $\overline{\text{RS}(\mathcal{A})}$ obtained from $\text{RS}(\mathcal{A})$ by taking non-strict inequalities instead of strict ones.

Proposition 2. *For region split automata:*

- words, paths, and region paths are in natural bijection;
- volume of any path is less or equal to 1;
- every path of consecutive edges has a non empty and non punctual language.

As for our running examples, $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ are already region split automata. $\overline{\text{RS}(\mathcal{A}_4)}$ is depicted in Fig. 2.

3.2 Paths and polytopes

Let us describe languages associated with paths as polytopes in \mathbb{R}^n , following [18].

Contiguous polytopes. Let $T_n = \{t_1, \dots, t_n\}$ be an ordered set of real variables bounded by M . A sum $S_{j..k} = \sum_{i=j}^k t_i$ is called a *contiguous sum* (of length $k - j + 1$). A *temporal inequality* is a constraint of the form $S_{j..k} \in [A, B]$ where $A, B \in \mathbb{N}$, $A \geq 0$ and $B \leq M$. A *contiguous polytope* is a bounded subset of

\mathbb{R}^n which is composed by all the points satisfying a conjunction of temporal inequalities. We say that the polytope is *d-contiguous* if the length of all sums in the inequalities is bounded by d .

Proposition 3. *For each path $\pi \in E^*$, $L(\bar{\pi})$ is a contiguous polytope.*

The inequality $\sum_{i=j}^k t_i \in [A, B]$ comes from testing the guard $x \in [A, B]$ during the k^{th} transition, provided that the last reset of x took place in the j^{th} transition. The subclass of *d-contiguous* polytopes corresponds to automata with progress cycle condition, where the number of transitions between two resets is bounded by d .

Given a polytope \mathcal{P} , we denote by $N\mathcal{P}$ its N -fold dilated copy, i.e. $\{N\mathbf{t} \mid \mathbf{t} \in \mathcal{P}\}$ and by $E(\mathcal{P}) = \mathcal{P} \cap \mathbb{Z}^n$ the set of points with integer coordinates in \mathcal{P} . A contiguous polytope is said to be *N-fat* if there exists an integer point in the interior of $N\mathcal{P}$ (called an *internal point*): $E(\text{int}(N\mathcal{P})) \neq \emptyset$.

3.3 Point to point reachability: algebraic characterization

In this section, we characterize the relation $\text{Reach}(\bar{\pi})$ in terms of an algebraic object: monoid of orbit graphs. Our analysis is less detailed than those in [10, 11, 16] and follows the lines of [19].

For a closed region $\bar{\mathbf{r}}$, let us denote by $V(\mathbf{r}) = \{S_1, \dots, S_p\}$ its vertices. Any point \mathbf{x} in the region is uniquely described by its barycentric coordinates $\lambda_1, \dots, \lambda_p$, i.e. nonnegative numbers such that $\sum_{i=1}^p \lambda_i = 1$; $\mathbf{x} = \sum_{i=1}^p \lambda_i S_i$.

Given two regions $\bar{\mathbf{r}}$ and $\bar{\mathbf{r}'}$, we call *orbit graph* any graph G with vertices $V(\mathbf{r}) \uplus V(\mathbf{r}')$ if \mathbf{r} and \mathbf{r}' are different and $V(\mathbf{r})$ otherwise, and with edges going from $V(\mathbf{r})$ to $V(\mathbf{r}')$. Informally, an edge from S to S' means that the clock vector at the vertex S can reach the clock vector at S' along some transition or path.

Orbit graphs compose in the natural way: for G_1 on regions $\bar{\mathbf{r}}_1$ and $\bar{\mathbf{r}}'_1$, and G_2 on regions $\bar{\mathbf{r}}_2$ and $\bar{\mathbf{r}}'_2$, their product $G = G_1 \cdot G_2$ is defined if $\bar{\mathbf{r}}'_1 = \bar{\mathbf{r}}_2$. In this case, G is an orbit graph on $\bar{\mathbf{r}}_1$ and $\bar{\mathbf{r}}'_2$. There is an edge from S to S'' in G if and only if there exists S' such that (S, S') and (S', S'') are edges of G_1 and G_2 . Whenever $\bar{\mathbf{r}}'_1 \neq \bar{\mathbf{r}}_2$, we put $G_1 \cdot G_2$ equal to some special (absorbing) element $\mathbf{0}$. The set \mathcal{G} of orbit graphs, augmented with $\mathbf{0}$ and a neutral element $\mathbf{1}$ has a structure of finite monoid.

An orbit graph G can be represented by its adjacency matrix M of size $|V(\mathbf{r})| \times |V(\mathbf{r}')|$. Products in the monoid of orbit graphs are easy to compute using matrices: $M(G_1 G_2) = M(G_1) \otimes M(G_2)$ where the “product” \otimes is defined by

$$(A \otimes B)_{ij} = \max_k \min(A_{ik}, B_{kj}).$$

There exists a natural morphism $\gamma : E^* \rightarrow \mathcal{G}$ from paths to orbit graphs defined as follows. For a transition e between $\bar{\mathbf{r}}$ and $\bar{\mathbf{r}'}$, we define the orbit graph $\gamma(e)$ on \mathbf{r} and \mathbf{r}' with edges $\{(S, S') \in V(\mathbf{r}) \times V(\mathbf{r}') \mid \exists t, S \xrightarrow{(e,t)} S'\}$. For a path $\pi = e_1 \dots e_n$, we define $\gamma(\pi) = \gamma(e_1) \dots \gamma(e_n)$ (it will be called the orbit graph of the path π). For the empty path we have $\gamma(\varepsilon) = \mathbf{1}$, and for any non-consecutive path $\gamma(\pi) = \mathbf{0}$.

For example, the orbit graphs of cycles ab and ba of \mathcal{A}_3 and \mathcal{A}_4 are complete, the orbit graphs of the other running examples are given in Fig. 3.

The orbit graph is crucial for reachability analysis.

Proposition 4. *The orbit graph of a path $\gamma(\pi)$ determines its reachability relation $\text{Reach}(\bar{\pi})$. In particular, $\gamma(\pi)$ is complete iff $\text{Reach}(\bar{\pi}) = \bar{\mathbf{r}} \times \bar{\mathbf{r}}'$, or equivalently iff $\text{Reach}(\text{int}(\pi)) = \mathbf{r} \times \mathbf{r}'$.*

The proof of the first criterion is based on the following remarkable characterization of $\text{Reach}(\bar{\pi})$ in terms of the orbit graph due to Puri [19].

Lemma 2. ⁶ *Let \mathbf{x} and \mathbf{x}' be two clock vectors with barycentric coordinates $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}'$. Then $(\mathbf{x}, \mathbf{x}') \in \text{Reach}(\bar{\pi})$ iff there exists a stochastic matrix $P \preceq M(\gamma(\pi))$, such that $\boldsymbol{\lambda}P = \boldsymbol{\lambda}'$.*

Here matrix “inequality” $A \preceq B$ means that $B_{ij} = 0 \Rightarrow A_{ij} = 0$ for all i, j .

Adding clock resets. For future use, we must enrich the monoid of orbit graphs by adding information on clock resets. Elements of the monoid \mathcal{M} are couples (orbit graph, subset of clocks) (and also, as before, two special elements $\mathbf{0}, \mathbf{1}$), the product rule is:

$$(G_1, X) \cdot (G_2, Y) = \begin{cases} (G_1 \cdot G_2, X \cap Y), & \text{if } G_1 \cdot G_2 \neq \mathbf{0} \\ \mathbf{0}, & \text{otherwise.} \end{cases}$$

For each $\pi \in E^*$ we denote by $\nu(\pi)$ the set of clocks not reset along the path π . We define a morphism $\mu : E^* \rightarrow \mathcal{M}$ as follows: $\mu(\pi) = (\gamma(\pi), \nu(\pi))$.

Idempotents. An idempotent of a monoid is an element m such that $m \cdot m = m$. Every finite monoid contains an idempotent. In our case, an idempotent orbit graph is always associated to a cyclic path, it is a graph G equal to its transitive closure $G^+ = \cup_{n \in \mathbb{N}^+} G^n$.

4 The thin-thick alternative and its consequences

In this central section, we characterize thin and thick paths and languages, based on a new notion of a forgetful cycle.

4.1 Forgetful cycles, and the others

After reading a timed path $\pi \times \mathbf{t}$ from a state s_0 , the reached state s depends only on s_0 and on the delays \mathbf{t} . We will say that π is *forgetful* if s and s_0 are independent, i.e. all the following equivalent conditions hold: $\text{Reach}(\text{int}(\pi)) = \mathbf{r} \times \mathbf{r}'$, $\text{Reach}(\bar{\pi}) = \bar{\mathbf{r}} \times \bar{\mathbf{r}}'$, $\gamma(\pi)$ is complete.

If a cycle is non-forgetful, and moreover its orbit graph is not strongly connected, then it is possible to find a linear Lyapunov function:

⁶ An intuition behind this lemma could be as follows. A clock vector with barycentric coordinates $\boldsymbol{\lambda}$ in a region can be seen as a probabilistic distribution over vertices of this region (with probabilities $\boldsymbol{\lambda}$). The lemma says that this distribution, at each cycle, evolves exactly as in some Markov chain.

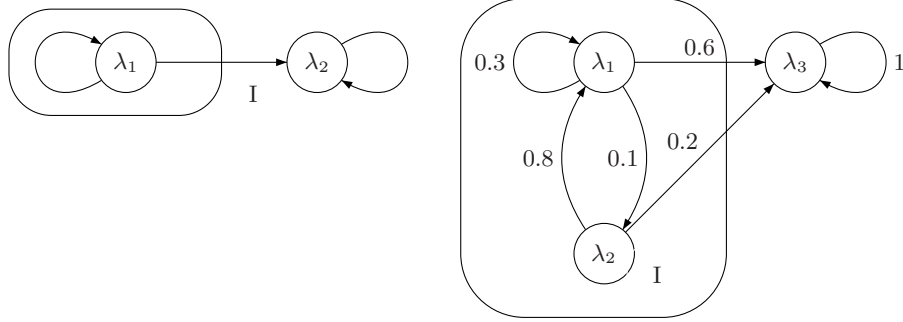


Fig. 3. Two non strongly connected orbit graphs, the first one is the orbit graph of the cycle of \mathcal{A}_1 , of the cycle ab of \mathcal{A}_2 and of the cycles a and b of \mathcal{A}_4 . States move from initial SCC I to final one. By choosing the convex combination of paths given by the Markov chain on the second orbit graph we pass from state $(\lambda_1 = 0.2, \lambda_2 = 0.5, \lambda_3 = 0.3)$ to state $(\lambda'_1 = 0.46, \lambda'_2 = 0.02, \lambda'_3 = 0.52)$. The sum $\lambda_1 + \lambda_2$ can only decrease.

Lemma 3. For a cycle π , if $\gamma(\pi)$ is not strongly connected then there exists a non empty $I \subsetneq \{1, \dots, p\}$ such that $\sum_{i \in I} \lambda'_i \leq \sum_{i \in I} \lambda_i$ whenever $(\mathbf{x}, \mathbf{x}') \in \text{Reach}(\bar{\pi})$, where $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}'$ stand for barycentric coordinates of \mathbf{x} and \mathbf{x}' .

In this lemma, as before, $\{1, \dots, p\}$ are indices of the vertices of the region where π starts (and ends).

In fact I corresponds to an initial strongly connected component (SCC) of the orbit graph, i.e. an SCC without incoming edges from other SCC. According to the lemma, the state moves from the facet spanned by I towards other vertices of the region and cannot come back.

Comparing to other types of cycles. Two other kinds of cycles are often considered in the literature: in a *progress cycle* each clock should be reset at some edge; in a *synchronizing cycle* all the clocks are reset along one and the same edge of the cycle.

Proposition 5. *progress cycles* \supseteq *forgetful cycles* \supseteq *synchronizing cycles*.

A remark is in order, in most works using progress or synchronizing cycles, **all the cycles are required** to be like that. In our work, **existence** of a forgetful cycle **appears** naturally in “non degenerate” (i.e. thick) automaton.

The condition of progress cycle can be seen as a weaker kind of forgetting: the state after such a cycle is exactly determined by the delays (see following lemma). Nevertheless the orbit graph of a progress cycle can be not strongly connected (e.g. cycle ab of \mathcal{A}_2 depicted in Fig. 3); in that case starting states and ending states are still dependent.

Lemma 4. If all clocks have been reset during reading of $\pi \times (t_1, \dots, t_m)$ then for all non empty $I \subsetneq \{1, \dots, p\}$, there exists $\alpha_1 \dots \alpha_m \in \{-p, \dots, p\}$ and an integer constant C such that $\sum_{i \in I} \lambda_i = C + \sum_{j=1}^m \alpha_j t_j$. Moreover one of the α_j is not zero.

4.2 Pumping lemma for long thick paths

For a given real $\eta > 0$, we say that a path π is η -*thick* if $\text{Vol}(L(\pi)) \geq \eta^{|\pi|}$. The following “pumping lemma” will play the key role in characterization of thick languages below and can be interesting by itself.

Theorem 1. *For every timed automaton \mathcal{A} and every $\eta > 0$, there exists N_η such that any η -thick path longer than N_η contains a forgetful cycle.*

The rest of this section is devoted to the proof of this result.

Elements of the monoid \mathcal{M} associated to forgetful cycles will be referred to as *forgetful*, they are idempotent. We will first see how repeating a non forgetful idempotent induces a subexponential volume (like the simplex example), then we will use Simon’s theorem on factorization forests to factorize paths and find some repeated idempotent. Absence of forgetful cycles in a path will then imply thinness.

Proposition 6. *Let π_1, \dots, π_k be k cycles of E^* such that $\mu(\pi_1), \dots, \mu(\pi_k)$ are all equal to a same non forgetful idempotent of \mathcal{M} , then $\text{Vol}(L(\pi_1 \dots \pi_k)) \leq \frac{M^{n-k}}{k!}$ where $n = |\pi_1| + \dots + |\pi_k|$.*

Proof. If G is an idempotent orbit graph (thus equal to its transitive closure), G is complete if and only if G is strongly connected. Thus we will distinguish two disjoint kinds of non forgetful idempotents, those associated to non progress cycles and those associated to progress cycles with non strongly connected orbit graphs. In the former case a clock is not reset all along the path $\pi_1 \dots \pi_k$, thus $L(\pi_1 \dots \pi_k)$ is in a simplex of type 1 and the volume satisfies the inequality to prove. In the latter case, we use Lem. 3,4, and Cor. 1. \square

A *factorization forest* of a word π is an unranked labeled tree with leaves labeled by the letters of π , with root labeled by π and with two types of internal nodes:

- binary node labeled by a word $\pi_1 \cdot \pi_2$ with two children labeled by the words π_1 and π_2 ;
- idempotent node labeled by a word $\pi_1 \dots \pi_k$ with all $\mu(\pi_i)$ equal to a same idempotent and with children labeled by the words π_1, \dots, π_k .

Theorem (Simon [20]). *If μ is a morphism from E^* to a finite monoid \mathcal{M} , then every word admits a factorization forest of height at most $h(\mathcal{M}) = 9|\mathcal{M}|$.*

We suppose that there are no forgetful cycles on a long path π and consider its factorization forest of height at most $h(\mathcal{M})$. When its length n grows up, the number of leaves also grows and since the height is bounded, branching of nodes must get larger and larger. These hugely branched nodes are idempotent and satisfy hypotheses of Lem. 6, thus their volume is very small, which implies that $\text{Vol}(L(\pi))$ is also small. The Prop. 7 below quantifies this “smallness” of $\text{Vol}(L(\pi))$ as function of the length of π and height of its factorization forest, and Thm. 1 follows immediately from this proposition.

Let LVol be the function defined on paths by $\text{LVol}(\pi) = \log_2 \text{Vol}(L(\pi))$. This function is subadditive and non-positive, i.e. $\text{LVol}(\pi_1 \cdot \pi_2) \leq \text{LVol}(\pi_1) + \text{LVol}(\pi_2) \leq 0$. Let $L(n, h)$ be the maximum of $\text{LVol}(\pi)$ over paths π of length n that do not contain forgetful idempotents and admit a factorization forest of height at most h .

Proposition 7. *For any height h , for any $C < 0$, there exists $N_{h,C} \in \mathbb{N}$ such that for all $n > N_{h,C}$ the inequality $L(n, h) \leq Cn$ holds.*

Proof. We will define $N_{h,C}$ by induction on the height h . Let a be a factorization forest of height h with n leaves. We consider all the children of the root and their subtrees (all these subtrees have heights $\leq h - 1$), and distinguish two disjoint cases:

1. There are more than $k = \frac{n}{2N_{h-1,2C}}$ subtrees having less than $N_{h-1,2C}$ leaves.
2. There are less than $k = \frac{n}{2N_{h-1,2C}}$ subtrees with less than $N_{h-1,2C}$ leaves. Here the juicy part (sons with enough leaves to satisfy induction hypothesis) has more than $\frac{n}{2}$ leaves.

In the first case: root is an idempotent node and we can apply Lem. 6:

$$\text{LVol}(\pi) \leq (n - k) \log_2(M) - \log_2(k!) \leq nC \text{ for } n \text{ large enough.}$$

In the second case: $\text{LVol}(\pi) \leq \sum_{i=1}^k L(n_i, h_i) \leq \sum_{n_i \geq N_{h-1,2C}} L(n_i, h_i)$. We apply the induction hypothesis:

$$\text{LVol}(\pi) \leq 2C \sum_{n_i \geq N_{h-1,2C}} n_i \leq 2C \frac{n}{2} \leq nC \text{ (recall that } C \text{ is negative). } \square$$

To conclude the proof of Thm. 1, given $\eta > 0$, let $C = \log_2 \eta$ and $h = h(\mathcal{M})$ the bound on height of factorization forest. Using Prop. 7, we obtain that a path longer than $N_{h,C}$ without forgetful idempotents cannot be η -thick. \square

4.3 Characterizing thick languages

We are ready to describe thick languages now.

Theorem 2. *For a right resolving timed automaton in region split form the following conditions are equivalent and define thick languages:*

1. $\mathcal{H} > -\infty$;
2. there exists a forgetful cycle;
3. there exists a strong limit cycle;
4. there exists an ε -discrete strong limit cycle with $\varepsilon > 0$.

Equivalence between 3 and 4 can be found in [16]. $2 \Rightarrow 3$ is straightforward.

Proof of 4 \Rightarrow 1. There exist $q_0, \dots, q_{d-1}, \mathbf{x}_0, \dots, \mathbf{x}_{d-1}, \pi \in E^d, u_1, \dots, u_d \in \{\varepsilon, 2\varepsilon, \dots, M - \varepsilon\}$ such that $(q_0, \mathbf{x}_0) \xrightarrow{(u_1, w_1)} (q_1, \mathbf{x}_1) \dots \xrightarrow{(u_d, w_d)} (q_0, \mathbf{x}_0)$ along π and all the \mathbf{x}_i are not on the frontier of regions and have discrete coordinates. First we can see that all clocks have been reset at least once because any non-reset clock would augment during the run, which contradicts its cyclicity. Then for each $n \in \mathbb{N}^*$ the language $L(\pi^n)$ is a d -contiguous polytope with equation of the form $A \leq \sum_{i=1}^d t_i \leq B$. Extending u periodically permits to have a word in $L(\pi^n)$ such that $A + \varepsilon \leq \sum_{i=1}^d u_i \leq B - \varepsilon$. Taking $t_i \in [u_i - \frac{\varepsilon}{d}, u_i + \frac{\varepsilon}{d}]$ defines a hypercube included in $L(\pi^n)$ whose volume is therefore greater than $(\frac{2\varepsilon}{d})^{nd}$. Then $\mathcal{H}(\mathcal{A}) \geq \log_2 \frac{2\varepsilon}{d} > -\infty$. \square

Proof of 1 \Rightarrow 2. We notice first that a thick language contains long thick paths.

Lemma 5. *If $\mathcal{H} > -\infty$, there exists $\eta > 0$ such that for all n big enough, there exists an η -thick path of length n .*

Proof. Let $\beta = 2^{\mathcal{H}-1}$. For n large enough $\text{Vol}(L_n) \geq \beta^n$. Let $\pi_{n,\max}$ be the path of E^n of maximal volume, then $\text{Vol}(L_n) \leq \text{Vol}(L(\pi_{n,\max}))|E|^n$ and so if we pose $\eta = \frac{\beta}{|E|}$ we have $\text{Vol}(L(\pi_{n,\max})) \geq \eta^n$. \square

Combining Lem. 5 with Thm. 1 we find a required forgetful cycle. \square

4.4 Thin and thick SCC

The theory developed above can be refined using a decomposition of \mathcal{A} into strongly connected components (SCC) $\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_k$.

Proposition 8. *Volumetric entropy of \mathcal{A} equals the maximal volumetric entropy of its SCC. In particular, \mathcal{A} is thin iff so are all the subautomata \mathcal{A}_i .*

It is easy to see that long and thick paths spend most of the time in thick SCC.

Theorem 3. *For every timed automaton \mathcal{A} and every $\eta, \alpha > 0$, there exists $N_{\eta,\alpha}$ such that for any η -thick path of length $n > N_{\eta,\alpha}$ at most $n\alpha$ states belong to thin SCC.*

5 Entropies of thick languages

In this section, we apply the results of the previous section to show that in thick automata, volumes and entropies can be computed with a good precision using discretization.

Theorem 4. *For a thick strongly connected automaton \mathcal{A} in region split form, the discrete and the volumetric entropies are related as follows⁷:*

$$h_\varepsilon = \log_2 \frac{1}{\varepsilon} + \mathcal{H} + o(1).$$

⁷ It can be proved that $o(1)$ is in fact $O\left(\varepsilon^{1/3} (\log_2 \frac{1}{\varepsilon})^{2/3}\right)$.

Proof (\geq direction). To bound the volume of L_n by the number of discrete points, we will use a beautiful theorem on counting points in polytopes:

Theorem (Ehrhart, see [8]). *For integer N and an integer polytope $\mathcal{P} \subset \mathbb{R}^n$ (i.e. whose vertices have integer coordinates), the number of integer points $|E(N\mathcal{P})|$ is a polynomial in N with non negative coefficients of degree n and whose coefficient of the highest degree is the volume.*

We deduce directly from this theorem that for each path π of length n and $\varepsilon = \frac{1}{N}$ the following holds: $\mathbf{Vol}(L(\pi))N^n \leq |E(NL(\pi))| = |L_\varepsilon(\pi)|$. Summing over all words of length n and taking $\lim_{n \rightarrow \infty} \frac{1}{n} \log_2$, we obtain that $\mathcal{H} + \log_2 \frac{1}{\varepsilon} \leq h_\varepsilon$. \square

Proof (\leq direction of Thm. 4).

Upper bounding h_ε by $\mathcal{H} + \log_2 \frac{1}{\varepsilon} + o(1)$ is more involved, and we give only a sketch of proof. We fix several integer parameters: a, b, c, d, e (they have to be adjusted in order to obtain the required estimate). Let π be a path of a length $n > a$. At every b transitions, we insert in π a forgetful cycle of length c (it exists by virtue of Thm. 2, and can be made of the same fixed length everywhere for an appropriate choice of c). Thus we obtain a slightly longer path π' (its length is $n' \approx n(1 + c/b)$), satisfying two additional conditions:

- every clock is reset at least every $2d$ transitions (and thus $L(\pi')$ is $2d$ -contiguous polytope);
- the polytope $L(\pi')$ is e -fat.

We have three inequalities:

1. The first one:

$$|L_\varepsilon(\pi)| \leq |L_\varepsilon(\pi')|$$

can be proved by constructing an injection from the left-hand side discrete language to the right-hand side one.

2. We choose ε' slightly smaller than ε (another parameter to adjust) and consider the polytope L^- obtained from $L(\pi')$ by pushing all its facets inside by the amount $\delta = \varepsilon' d$ ⁸. Using fatness of $L(\pi')$, it is possible to build an injection from its ε -discrete points to ε' -discrete points of L^- (the latter is a bit smaller but its discrete points are slightly denser).

$$|L_\varepsilon(\pi')| \leq |L_{\varepsilon'}^-|.$$

3. Taking an ε' -cube at every ε' -discrete point of L^- , we get a set included in $L(\pi')$ (this requires $2d$ -contiguity of $L(\pi')$). Passing to volumes we conclude that

$$\varepsilon'^{n'} |L_{\varepsilon'}^-| \leq \mathbf{Vol}(L(\pi')).$$

⁸ i.e. by replacing each constraint $S_{j..k} \in [A, B]$ in the definition of $L(\pi')$ as a contiguous polytope by $S_{j..k} \in [A + \delta, B - \delta]$ (see [4]).

Combining the three inequalities we get:

$$|L_\varepsilon(\pi)| \leq \left(\varepsilon'^{-n'} \mathbf{vol}(L(\pi')) \right),$$

and with an appropriate choice of parameters, ε' and n' can be made very close to ε and n . Summing up over π and taking $\lim_{n \rightarrow \infty} \frac{1}{n} \log_2$ in the previous inequality, we obtain the required result. \square

Corollary 2. *For right resolving TA with bounded clocks, $\mathcal{H}(\mathcal{A})$ is computable as function of \mathcal{A} . Consequently, $\mathcal{H}(\mathcal{A})$ is a computable real (i.e. one can compute its approximation with any wanted precision).*

Proof. First compute $\text{RS}(\mathcal{A})$. Then compute by fixpoint method the submonoid of orbit graphs $\gamma(E^*) \subset \mathcal{G}$ and see whether there is a complete graph. If there is none, the automaton is thin and $\mathcal{H} = -\infty$. Otherwise, the automaton is thick and it just remains to compute the discrete entropy of $L_\varepsilon(\text{RS}(\mathcal{A}))$ for the wanted precision (similarly to [4]). \square

6 Conclusion and future work

We have identified the class of thick timed automata (those with non-vanishing language volume). Most runs in such automata are thick and exhibit a nice behavior, they spend most of the time in thick strongly connected components (Thm. 3) and visit from time to time forgetful cycles (Thm. 1). Thick runs are captured (both qualitatively and quantitatively) by ε -discretized automata.

We believe that the notions of thick languages and forgetful cycles will be useful in the operator approach to volume and entropy of [3] and will imply some good properties of operators associated to these forgetful cycle. Similarly, we believe that thickness hypothesis is exactly what is needed for the analysis of probabilistic timed systems in the spirit of [9] but for an unbounded time horizon. Another direction of future work is to extend the thin-thick dichotomy to the case of punctual paths and to find when the two size measures of [5] are defined. We hope also to relate thinness with the notion of mean topological dimension [17]. In the verification context, we believe that when analyzing a thick timed automaton, it suffices to check that the thick paths satisfy the specification, while thin ones can violate it.

Acknowledgements. The authors are thankful to Thomas Colcombet for a key advise: to use Simon factorization forests, and to Dominique Perrin and Aldric Degorre for inspiring discussions.

References

1. Abdulla, P.A., Krcál, P., Yi, W.: Sampled semantics of timed automata. *Logical Methods in Computer Science* 6(3) (2010)
2. Alur, R., Dill, D.L.: A theory of timed automata. *Theor. Comput. Sci.* 126, 183–235 (1994)
3. Asarin, E., Degorre, A.: Volume and entropy of regular timed languages: Analytic approach. In: *FORMATS. LNCS*, vol. 5813, pp. 13–27. Springer (2009)
4. Asarin, E., Degorre, A.: Volume and entropy of regular timed languages: Discretization approach. In: *CONCUR. LNCS*, vol. 5710, pp. 69–83. Springer (2009)
5. Asarin, E., Degorre, A.: Two size measures for timed languages. In: *FSTTCS. LIPIcs*, vol. 8, pp. 376–387. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2010)
6. Asarin, E., Maler, O., Pnueli, A.: On discretization of delays in timed automata and digital circuits. In: *CONCUR. LNCS*, vol. 1466, pp. 470–484. Springer (1998)
7. Beauquier, D.: Pumping lemmas for timed automata. In: *FoSSaCS. LNCS*, vol. 1378, pp. 81–94. Springer (1998)
8. Beck, M., Robins, S.: Computing the continuous discretely: Integer-point enumeration in polyhedra. Springer (2007)
9. Carnevali, L., Grassi, L., Vicario, E.: State-density functions over DBM domains in the analysis of non-Markovian models. *IEEE Trans. Software Eng.* 35(2), 178–194 (2009)
10. Comon, H., Jurski, Y.: Timed automata and the theory of real numbers. In: *CONCUR. LNCS*, vol. 1664, pp. 242–257. Springer (1999)
11. Dima, C.: Computing reachability relations in timed automata. In: *LICS*. pp. 177–. IEEE Computer Society (2002)
12. Gómez, R., Bowman, H.: Efficient detection of Zeno runs in timed automata. In: *FORMATS. LNCS*, vol. 4763, pp. 195–210. Springer (2007)
13. Gupta, V., Henzinger, T.A., Jagadeesan, R.: Robust timed automata. In: *HART. LNCS*, vol. 1201, pp. 331–345. Springer (1997)
14. Henzinger, T.A., Manna, Z., Pnueli, A.: What good are digital clocks? In: *ICALP. LNCS*, vol. 623, pp. 545–558. Springer (1992)
15. Henzinger, T.A., Raskin, J.F.: Robust undecidability of timed and hybrid systems. In: *HSCC. LNCS*, vol. 1790, pp. 145–159. Springer (2000)
16. Krcál, P.: Infinite Structures in Timed Systems. Ph.D. thesis, University of Uppsala, Dept. of Information Technology (May 2009)
17. Lindenstrauss, E., Weiss, B.: Mean topological dimension. *Israel J. Math* 115, 1–24 (2000)
18. Maler, O., Pnueli, A.: On recognizable timed languages. In: *FoSSaCS. LNCS*, vol. 2987, pp. 348–362. Springer (2004)
19. Puri, A.: Dynamical properties of timed automata. *Discrete Event Dynamic Systems* 10(1-2), 87–113 (2000)
20. Simon, I.: Factorization forests of finite height. *Theor. Comput. Sci.* 72(1), 65–94 (1990)
21. Wulf, M.D., Doyen, L., Markey, N., Raskin, J.F.: Robust safety of timed automata. *Formal Methods in System Design* 33(1-3), 45–84 (2008)
22. Wulf, M.D., Doyen, L., Raskin, J.F.: Almost ASAP semantics: from timed models to timed implementations. *Formal Asp. Comput.* 17(3), 319–341 (2005)

A Proofs and minor results

A.1 Region split automaton

We assume w.l.o.g. that there is a special clock T in the automaton \mathcal{A} to be split with a trivial behavior: in every transition, guards are of the form $0 \leq T \leq M$ and T is reset at each transition.

The volume of a path in the region split automaton is less or equal to 1. In the region split automaton the special clock T is reset at every transition and verify guards of the form $T \in (A, A + 1)$. Thus times t_i vary in intervals of length 1. The language of each path is then included in a hypercube of side 1. Volume is therefore less or equal to 1.

Proof of proposition 1. We will construct step by step automata \mathcal{A}^i such that B_1 to B_i is true for \mathcal{A}^i and such that $\mathcal{H}(\mathcal{A}) = \mathcal{H}(\mathcal{A}^i)$. To construct \mathcal{A}^2 we refer to the procedure of [3] without the two last points (because we want to keep all states initial). \mathcal{A}^4 can be constructed from \mathcal{A}^2 by removing locations that do not satisfy B_4 and by setting the initial and final states to be the entry regions. We must show that $\mathcal{H}(\mathcal{A}^4) = \mathcal{H}(\mathcal{A}^2)$. First we have that $\mathcal{H}(\mathcal{A}^4) \leq \mathcal{H}(\mathcal{A}^2)$ as $L(\mathcal{A}^4) \subseteq L(\mathcal{A}^2)$. We call internal locations which satisfy B_4 and external the others. Let k be the maximal length of a consecutive path composed exclusively with edges involving external locations (we speak of external path). We factorize every path of \mathcal{A}^2 with length greater than $2k$ as $\pi_{(1)}\pi_{(2)}\pi_{(3)}$ with $\pi_{(1)}, \pi_{(3)} \in E^k$. By definition of k , $\pi_{(2)}$ is a path of \mathcal{A}^4 . Moreover we have $\text{Vol}(L(\pi)) \leq \text{Vol}(L(\pi_{(1)}))\text{Vol}(L(\pi_{(2)}))\text{Vol}(L(\pi_{(3)})) \leq M^{2k}\text{Vol}(L(\pi_{(2)}))$. Then $\text{Vol}(L_{n+2k}(\mathcal{A}^2)) \leq \text{Vol}(L_n(\mathcal{A}^4))M^{2k}$ and we are done by taking $\lim_{n \rightarrow \infty} \frac{1}{n} \log_2$.

\mathcal{A}^5 is constructed from \mathcal{A}^4 by deleting edges which do not satisfy B_5 (one has also to delete locations which no more satisfy B_3, B_4). By doing this we delete punctual paths, their volumes are null and do not contribute to \mathcal{H} .

\mathcal{A}^6 is constructed from \mathcal{A}^5 by taking identity as labeling function.

For each w we have $L(w) = \cup_{\pi \in \mathcal{L}^{-1}(w)} L(\pi)$ and then

$$\text{Vol}(L_n(\mathcal{A}^5)) = \sum_{w \in \Sigma^n} \text{Vol}(L(w)) \leq \sum_{w \in \Sigma^n} \sum_{\pi \in \mathcal{L}^{-1}(w)} \text{Vol}(L(\pi)) = \text{Vol}(L_n(\mathcal{A}^6)).$$

For the converse inequality, everything would be very simple if the above union were disjoint but this is not the case due to freedom on the initial state. However from one state s the languages $L(\pi_{(2)}, s)$, $L(\pi'_{(2)}, s)$ starting from s and associated with two distinct paths $\pi_{(2)} \neq \pi'_{(2)}$ are disjoint (as guards and regions of $\pi_{(2)}$ and $\pi'_{(2)}$ are disjoint at a moment). Then, if all clocks have been reset during a path $\pi_{(1)}$, for every two distinct paths $\pi_{(2)}, \pi'_{(2)}$ we have $L(\pi_{(1)}\pi_{(2)}) \cap L(\pi_{(1)}\pi'_{(2)}) = \emptyset$. We divide in two groups the set of paths of a given length l (l is a parameter that we will tune later):

- the set $R(l)$ of paths which have reset all its clocks;
- the set of other paths which have a volume $\text{Vol}(L(\pi)) \leq \frac{1}{l!}$

$$\begin{aligned} \sum_{\pi \in E^n} \text{Vol}(L(\pi)) &= \sum_{\pi_{(1)} \in R(l)} \sum_{\pi_{(2)} \in E^{n-l}} \text{Vol}(L(\pi_{(1)}\pi_{(2)})) + \\ &\quad \sum_{\pi_{(1)} \notin R(l)} \sum_{\pi_{(2)} \in E^{n-l}} \text{Vol}(L(\pi_{(1)}\pi_{(2)})). \end{aligned}$$

We will denote by S_1 and S_2 the two sums above. S_2 is upper bounded by $\frac{|E|^n}{l!}$. For each $w \in \Sigma^n$ we have

$$L(w) = \bigcup_{\pi_{(1)} \in R(l)} \biguplus_{\substack{\pi_{(2)} \in E^{n-l} \\ \pi_{(1)}\pi_{(2)} \in \mathcal{L}^{-1}(w)}} L(\pi_{(1)}\pi_{(2)})$$

and then

$$\begin{aligned} \text{Vol}(L(w)) &\geq \max_{\pi_{(1)} \in R(l)} \sum_{\substack{\pi_{(2)} \in E^{n-l} \\ \pi_{(1)}\pi_{(2)} \in \mathcal{L}^{-1}(w)}} \text{Vol}(L(\pi_{(1)}\pi_{(2)})) \geq \\ &\quad \frac{1}{|E|^l} \sum_{\pi_{(1)} \in R(l)} \sum_{\substack{\pi_{(2)} \in E^{n-l} \\ \pi_{(1)}\pi_{(2)} \in \mathcal{L}^{-1}(w)}} \text{Vol}(L(\pi_{(1)}\pi_{(2)})) \end{aligned}$$

We sum over all w and deduce $S_1 \leq |E|^l \text{Vol}(L_n(\mathcal{A}^5))$. Then $\text{Vol}(L_n(\mathcal{A}^6)) \leq \text{Vol}(L_n(\mathcal{A}^5))(|E|^l + \frac{|E|^n}{l! \text{Vol}(L_n(\mathcal{A}^5))})$. We can find l such that $l = o(n)$ and $\frac{|E|^n}{l! \text{Vol}(L_n(\mathcal{A}^5))} = O(1)$ (e.g. $n = \sqrt{\log_2 l!} \sim \sqrt{l \log_2 l}$). With such an l the quantity $\frac{1}{n} \log_2(|E|^l + \frac{|E|^n}{l! \text{Vol}(L_n(\mathcal{A}^5))})$ tends to 0 and then $\mathcal{H}(\mathcal{A}^5) \geq \mathcal{H}(\mathcal{A}^6)$.

Proof of proposition ??. A part of the previous proof is easy to adapt to show that $h_\varepsilon(\mathcal{A}) = h_\varepsilon(\mathcal{A}^4)$.

The following lemma says that $L(\mathcal{A}^5)$ is the non punctual part of $L(\mathcal{A}^4)$ and thus $h_\varepsilon(\overline{L(\mathcal{A}^4)}) = h_\varepsilon(\overline{L(\mathcal{A}^5)})$ if and only if $L(\mathcal{A}^4)$ is not punctual i.e. $\overline{L(\mathcal{A}^4)} = \text{int}(L(\mathcal{A}^4))$.

Lemma 6. $\overline{L(\mathcal{A}^5)} = \overline{\text{int}(L(\mathcal{A}^4))}$

Proof. Let $\text{NPin}(w)$ (resp. $\text{Pin}(w)$) be the set of non punctual (resp. punctual) paths π such that $\mathcal{L}(\pi) = w$. $\text{NPin}(w)$ is exactly the set of paths of $L(\mathcal{A}^5)$ labeling w . Thus we have to show that for each w , $\text{int}(L(w)) \subset \overline{\cup_{\pi \in \text{NPin}(w)} L(\pi)}$ (the converse inclusion is straightforward as $\cup_{\pi \in \text{NPin}(w)} L(\pi)$ is an open subset of $L(w)$). Let x be a point in $\text{int}(L(w))$. Suppose by contradiction that $x \notin \overline{\cup_{\pi \in \text{NPin}(w)} L(\pi)}$, then for ε small enough x is ε far from $\cup_{\pi \in \text{NPin}(w)} L(\pi)$. x is less than $\frac{\varepsilon}{2}$ far from a point y in $\text{int}(L(w))$. y is then in the interior of $\cup_{\pi \in \text{Pin}(w)} L(\pi)$ which is absurd as this set has empty interior. \square

As in the proof of the previous proposition, for each w we have $\overline{L(w)} = \cup_{\pi \in \mathcal{L}^{-1}(w)} \overline{L(\pi)}$ and then $|L_{n,\varepsilon}(\overline{\mathcal{A}^5})| = \sum_{w \in \Sigma^n} |L_{n,\varepsilon}(w)| \leq \sum_{w \in \Sigma^n} \sum_{\pi \in \mathcal{L}^{-1}(w)} |L_{n,\varepsilon}(\pi)| = |L_{n,\varepsilon}(\overline{\mathcal{A}^6})|$. We deduce that $h_\varepsilon(\overline{\mathcal{A}^5}) \leq h_\varepsilon(\overline{\mathcal{A}^6})$

Lemma 7. $h_\varepsilon(\mathcal{A}^6) \leq h_\varepsilon(\mathcal{A}^5)$

Proof. For a path π and a vector of delays \mathbf{t} , one can show that the set of states from which $\pi \times \mathbf{t}$ can be read is a polytope of $\mathbb{R}^{|\mathcal{X}|}$. Let us denote this set by $\text{Start}(\pi, \mathbf{t}) = \{\mathbf{x} \mid \exists \mathbf{x}', \mathbf{x} \xrightarrow{(t_1, \pi_1)(t_2, \pi_2) \dots (t_n, \pi_n)} \mathbf{x}'\}$. We denote by S_ε the set of $\frac{\varepsilon}{|\mathcal{X}|+1}$ -discrete and internal states. By lemma 2 of [6] if \mathbf{t} is ε -discrete, one can find a state in $\text{Start}(\pi, \mathbf{t}) \cap S_\varepsilon$. We deduce that $|L_{n,\varepsilon}(\mathcal{A}^5)| \leq |S_\varepsilon| \max_{\mathbf{x} \in S_\varepsilon} |L(\pi, \mathbf{x})| \leq |S_\varepsilon| |L_{n,\varepsilon}(\mathcal{A}^6)|$. As $|S_\varepsilon| \leq |Q| \frac{(|\mathcal{X}|+1)M_\varepsilon}{\varepsilon}$ which is a constant in n , by taking $\lim_{n \rightarrow \infty} \frac{1}{n} \log_2$ we conclude the proof. \square

We have the chain

$$h_\varepsilon(\mathcal{A}^6) \leq h_\varepsilon(\mathcal{A}^5) \leq h_\varepsilon(\mathcal{A}^4) \leq h_\varepsilon(\overline{L(\mathcal{A}^4)}) = h_\varepsilon(\overline{L(\mathcal{A}^5)}) \leq h_\varepsilon(\overline{\mathcal{A}^6}).$$

The remaining inequalities are straightforward due to language inclusion.

We close the cycle with the following lemma which is shown using the theory of section 5 (it should be read before the next proof).

Lemma 8. For non punctual automaton $h_\varepsilon(\text{RS}(\mathcal{A})) = h_\varepsilon(\overline{\text{RS}(\mathcal{A})}) + o(1)$

Proof. With the notation of the proof of theorem 4 we have :

$$|L_{n,\varepsilon}| \leq |L_{n',\varepsilon'}^-| \leq |\text{int}(L)_{n',\varepsilon'}| \leq |L_{n',\varepsilon'}|.$$

n' and ε' are designed in such a way that taking $\lim_{n \rightarrow \infty} \frac{1}{n} \log_2$ we obtain the result (see computations at the end of the proof of theorem 4).

A.2 Polytope of paths

Proof of proposition 3. We suppose that π_1, \dots, π_n are consecutive and then form a region path, otherwise $L(\pi)$ is empty and thus a contiguous polytope. We denote by $x^1, \dots, x^{|\mathcal{X}|}$ the clocks and by x_k^l the value of the clock x^l after the k^{th} transition. We call a mixed condition on t_1, \dots, t_n a condition of the form $\exists x_0^1, \dots, \exists x_0^{|\mathcal{X}|} (\bigwedge_i I_i(x_0^1, \dots, x_0^{|\mathcal{X}|}))$ where $I_i(x_0^1, \dots, x_0^{|\mathcal{X}|})$ are inequalities of three types:

- inequalities between two clocks without time: $x_0^l \leq B + x_0^p$;
- inequality between a clock and a contiguous sum starting from index 1:
 $x_0^l \leq B - S_{1..j}, A - S_{1..j} \leq x_0^l$;
- inequalities without clocks (which are temporal inequalities): $A \leq S_{j..k}, S_{j..k} \leq B$.

We will proceed in two steps: first we show that $L(\pi)$ verifies a mixed condition, then we show that any mixed condition describes a contiguous polytope by eliminating (projecting) clock variables with the Fourier-Motzkin procedure:

First step: Starting clock values satisfy the condition of the starting region: $x_0^l \leq B + x_0^p$, $A \leq x_0^l \leq B$. For all k between 1 and $n - 1$, the guard $x^l \in [A, B]$ of the k^{th} transition will be translated in $x_{k-1}^l + t_k \in [A, B]$. There are two cases, either x^l has not been reset and then $x_{k-1}^l + t_k = x_0^l + S_{1..k}$, we add the inequality $A - S_{1..k} \leq x_0^l \leq B - S_{1..k}$; or x^l has been reset for the last time in the j^{th} transition and then $x_{k-1}^l + t_k = S_{j..k}$, we add the inequality $A \leq S_{j..k} \leq B$.

Second step: Suppose by induction that we have found a mixed condition describing $L(\pi)$ without clocks x_0^1, \dots, x_0^{l-1} , we will show how to get rid of the clock x_0^l . Let us divide inequalities in five groups:

1. inequality without x_0^l ;
2. inequalities of type: $A - S_{1..i} \leq x_0^l$;
3. inequalities of type: $x_0^l \leq B - S_{1..j}$;
4. inequalities of type: $x_0^p \leq C + x_0^l$;
5. inequalities of type: $x_0^q \leq D + x_0^l$.

We keep inequalities of the first group, we replace others by a conjunction of equivalent inequalities:

- $A - S_{1..i} \leq B - S_{1..j}$ formed with groups 2 and 3;
- $A - S_{1..i} \leq D + x_0^q$ formed with groups 2 and 5;
- $x_0^p \leq C + B - S_{1..j}$ formed with groups 3 and 4;
- $x_0^p \leq C + D + x_0^q$ formed with groups 2 and 4.

Inequalities are of the expected form (first is equivalent to $A - B \leq S_{i+1..j}$ or $A \leq B$ or $S_{j+1..i} \leq B - A$ whether $i < j$, $i = j$ or $i > j$). At the end there is only a conjunction of temporal inequalities i.e. a contiguous polytope.

A.3 Point to point reachability

Proof of proposition 4. We use lemma 2 to show that $G(\pi)$ is complete iff $\text{Reach}(\bar{\pi}) = \bar{\mathbf{r}} \times \bar{\mathbf{r}'}$. Suppose that $G(\pi)$ is complete. For all \mathbf{x}, \mathbf{x}' , we denote by Λ, Λ' vectors with corresponding convex coordinates. We define P as the matrix with rows equal to Λ' , we have $\Lambda P = \Lambda'$, then $\mathbf{x}, \mathbf{x}' \in \text{Reach}(\bar{\pi})$. We conclude that $G(\pi) \Rightarrow \text{Reach}(\bar{\pi}) = \bar{\mathbf{r}} \times \bar{\mathbf{r}'}$. The reciprocal is trivial.

$\text{Reach}(\bar{\pi})$ and $\text{Reach}(\text{int}(\pi))$ are polytopes obtained by quantifier elimination (projection) in the formula $\exists t_1, \dots, t_n, \mathbf{x} \xrightarrow{(t_1, \pi_1)(t_2, \pi_2) \dots (t_n, \pi_n)} \mathbf{x}'$. The result is a finite conjunction of exactly the same equalities and inequalities in both cases except that inequalities in $\bar{\pi}$ are non-strict and strict in $\text{int}(\pi)$. By flipping the nature of inequalities one can change $\bar{\mathbf{r}} \times \bar{\mathbf{r}'}$ in $\mathbf{r} \times \mathbf{r}'$ and vice versa.

Proof of proposition 5. Synchronizing cycles are by definition particular forget cycles. We do the other inclusion by contrapositive : if a cycle is not progressing then there is a clock x which is not reset and thus in the orbit graph of the cycle there is no edge from vertices in the hyperplane $x = A + 1$ to vertices in the hyperplane $x = A$ where A is the integer part of x in the region where the cycle starts and ends. The graph is not complete and thus the cycle is not forgetting.

More on synchronization cycle The stringent condition of synchronization cycle seems too strong for us but is a simpler interesting subcase, in fact if πa is a synchronizing cycle with a the synchronizing edge then $\text{Vol}(L((\pi a)^n)) = (\text{Vol}(L(\pi a)))^n$ and so $\mathcal{H} \geq \frac{1}{|\pi a|} \text{Vol}(L(\pi a)) > -\infty$. This strong condition has been used in [9].

Proof of lemma 4. Modulo a reordering of the clocks and a fusion of equal clocks we can suppose that the ending p -dimensional region $\bar{\mathbf{r}}$ is

$$[\bar{x}] + \{(\{x_1\}, \dots, \{x_p\}) \mid 0 \leq \{x_1\} \leq \{x_2\} \leq \dots \leq 1\},$$

where $[\bar{x}] = (\lfloor x_1 \rfloor, \dots, \lfloor x_p \rfloor)$. We have the following lemma to pass from cartesian coordinates (x_1, \dots, x_p) to barycentric coordinates $(\lambda_1, \dots, \lambda_{p+1})$:

Lemma 9. for $i \geq 2$, $\lambda_i = \{x_{p+2-i}\} - \{x_{p+1-i}\}$.

Proof. Vertices of the region are $S_1 = \lfloor x \rfloor + (0, \dots, 0)$, $S_2 = \lfloor x \rfloor + (0, \dots, 1)$, $S_{p+1} = \lfloor x \rfloor + (1 \dots 1)$. Therefore $(x_1, \dots, x_p) = \sum_{i \in I} \lambda_i S_i = \lfloor \bar{x} \rfloor + (\lambda_{p+1}, \lambda_{p+1} + \lambda_p, \dots, \lambda_{p+1} + \lambda_p + \dots + \lambda_2)$ and then for $i \geq 2$ we have $\lambda_i + (\lfloor x_{p+2-i} \rfloor - \lfloor x_{p+1-i} \rfloor) = x_{p+2-i} - x_{p+1-i}$.

For each $i \in \{2, \dots, p\}$, $\lambda_i + (\lfloor x_{p+2-i} \rfloor - \lfloor x_{p+1-i} \rfloor) = x_{p+2-i} - x_{p+1-i}$ is the sum of all delays between reset of x_{p+2-i} and x_{p+1-i} with a negative coefficient if $x_{p+2-i} < x_{p+1-i}$. Therefore every λ_i ($i \in \{2, \dots, p\}$) is of the form $-C_i \pm \sum t_j$ with $C_i \in \mathbb{N}$. If $1 \notin I$ then $\sum_{i \in I} \lambda_i$ is of the expected form else as $\lambda_1 = 1 - \sum_{i \geq 2} \lambda_i$, there exists $J \subseteq \{2, \dots, p\}$ such that $\sum_{i \in I} \lambda_i = 1 + \sum_{i \in J} \pm \lambda_i$; the sum is also of the expected form. Moreover, there is one coefficient distinct from 0 because $\sum_{i \in I} \lambda_i$ is not constant (otherwise dimension of the region would be less than p).

Proof of lemma 3. Let P be the matrix such that $AP = A'$. There exists an SCC I of $G(\pi)$ without incoming edges from other SCCs. Modulo a change of indices, we can suppose that indices of I are minimal and then matrix P has the following form:

$$P = \begin{pmatrix} P_{I \rightarrow I} & P_{I \rightarrow \bar{I}} \\ 0 & P_{\bar{I} \rightarrow \bar{I}} \end{pmatrix}.$$

If we decompose A in $(A_I, A_{\bar{I}})$ and A' in $(A'_I, A'_{\bar{I}})$ we have $A_I P_{I \rightarrow I} = A'_I$.

$$\sum_{j \in I} \lambda'_j = \sum_{j \in I} \sum_{i \in I} \lambda_i P_{ij} = \sum_{i \in I} \lambda_i \sum_{j \in I} P_{ij} \leq \sum_{i \in I} \lambda_i.$$

A.4 Thin and thick SCCs

Proof of proposition 8. First, if all subautomata are thin, as cycle are included in SCC, there is no forgetful cycle in the whole automaton which is

therefore thin. Let us denote by \mathcal{H}_{\max} the maximal volumetric entropy obtained for a subautomaton $\mathcal{A}_{i_{\max}}$. As $L(\mathcal{A}_{i_{\max}})$ is a sublanguage of $L(\mathcal{A})$ then $\mathcal{H}_{\max} \leq \mathcal{H}(\mathcal{A})$. For the converse inequality we use a straightforward consequence of the definition of volumetric entropy: \mathcal{H} is the least real such that $\forall \sigma > 0, \exists A > 0, \forall n \geq 0, \text{Vol}(L_n) \leq A2^{n(\mathcal{H}+\sigma)}$. Therefore, in each subautomaton we have $\forall \sigma > 0, \exists A > 0, \forall n \geq 0, \text{Vol}(L_n) \leq A2^{n(\mathcal{H}_{\max}+\sigma)}$. One can decompose L_n in a finite disjoint union of languages of the form $L_{n_1}(\mathcal{A}_{i_1}) \cdot L^{i_1 \rightarrow i_2} \cdot L_{n_2}(\mathcal{A}_{i_2}) \cdots L_{n_m}(\mathcal{A}_{i_m})$ where $i_1 < \cdots < i_m$, $n - c \leq n_1 + \cdots + n_m \leq n$ (for some constant c) and $L^{i_j \rightarrow i_{j+1}}$ are finite languages going from \mathcal{A}_{i_j} to $\mathcal{A}_{i_{j+1}}$ and thus with bounded volume. Volume V of such a language verifies $\forall \sigma > 0, \exists A' > 0, \forall n \geq 0, V \leq A'2^{(n_1 + \dots + n_m)(\mathcal{H}_{\max}+\sigma)} \leq A'2^{n(\mathcal{H}_{\max}+\sigma)}$ where $A' = \max(2^{-c(\mathcal{H}_{\max}+\sigma)}, 1)$ (with some constant c). Summing over elements of the finite disjoint union of languages gives the expected result $\forall \sigma > 0, \exists A > 0, \forall n \geq 0, \text{Vol}(L_n) \leq A2^{n(\mathcal{H}_{\max}+\sigma)}$ and then $\mathcal{H}_{\max} \geq \mathcal{H}(\mathcal{A})$.

Proof of theorem 3. Let $\lambda, \alpha > 0$ and π a λ -thick path of length n . Let r be the number of SCC in the automaton. We suppose by contradiction that π wanders more than αn transitions on thin SCCs and thus has a factor π' of length greater than $\frac{\alpha n}{r}$ in one thin SCC. In thin SCC, for every γ , for m large enough i.e. greater than a constant N_γ , every path π'' of length m is γ -thin : $\text{Vol}(\pi'') < \gamma^m$. If n is greater than $\frac{N_\gamma r}{\alpha}$ then $\text{Vol}(\pi) \leq \text{Vol}(\pi') < \gamma^{\frac{\alpha}{r} n}$ which contradict the λ -thickness when choosing γ such that $\gamma^{\frac{\alpha}{r}} = \lambda$.

Proposition 9. *In a thick SCC \mathcal{A}_i , there exists a constant c such that there is a forgetful cycle of length c on each location of \mathcal{A}_i .*

Proof. Let π be some forgetful cycle and \mathbf{r} be the region where π starts and ends. Let \mathbf{r}' be a region in the same SCC as \mathbf{r} . There is a path $\pi_{(1)}$ from \mathbf{r} to \mathbf{r}' and a path $\pi_{(2)}$ from \mathbf{r}' to \mathbf{r} . We will show that $\pi_{(1)}\pi_{(2)}$ is a forgetful cycle on \mathbf{r}' . As Puri stated in [19], in an orbit graph, there is an outgoing edge from each vertex of the starting region and an incoming edge to each vertex of the ending region. Let S, S' be two vertices of \mathbf{r}' , let S_1, S_2 be two vertices of \mathbf{r} such that (S, S_1) and (S_2, S') are respectively edges of $G(\pi_{(1)})$ and $G(\pi_{(2)})$. As $G(\pi)$ is complete there is an edge between S_1 and S_2 and so there is an edge between S and S' in $G(\pi_{(1)}\pi_{(2)})$. To sum up there is a forgetful cycle $\pi_{\mathbf{r}'}$ on each region \mathbf{r}' of the strongly connected subautomaton, the least common multiple of all length of $\pi_{\mathbf{r}'}$ gives an appropriate constant c .

A.5 Proof of theorem 4, more details for \leq direction

We begin by some auxiliary results used in the core of the proof.

Remark 1. An e -fat contiguous polytope is eM contiguous.

Proof. Let t_1, \dots, t_n an $\frac{1}{e}$ discrete and intern point of the polytope. Equations of the polytopes are of the form $A \leq S_{j..k} \leq B$ with $B \leq M$. As for every $i \in \{1, \dots, n\}$, $t_i > 0$ and t_i is multiple of $\frac{1}{e}$ then $t_i \geq \frac{1}{e}$ and $\frac{k-j+1}{e} \leq S_{j..k} \leq M$. We deduce the result: lengths $k - j + 1$ of contiguous sums are bounded by eM .

The following two discretization lemmata are from [6].

Lemma 10. *If \mathbf{x}, \mathbf{x}' are ε -discrete and $L(\overline{\pi}, \mathbf{x}, \mathbf{x}') \neq \emptyset$ then $L_\varepsilon(\overline{\pi}, \mathbf{x}, \mathbf{x}') \neq \emptyset$.*

Lemma 11. *Every contiguous polytope \mathcal{P} of dimension n has an $\frac{1}{m}$ -discrete internal point for all $m > n$.*

The core of the proof of 4 Let c be a constant such that there is a progress cycle of length c from everywhere in the automaton (see proposition 9 above). Let π be a path of length $n = mb$, we will insert in π at every b letters a forgetful cycle f_i such that polytopes associated to the created word are e -fat with $e = (b + 1)(c + 1)$ (and thus contain more discrete points).

Let $\pi = \pi_{(1)}\pi_{(2)} \dots \pi_{(m)}$ where $\pi_{(1)}, \dots, \pi_{(m)}$ are words of length b . For all $i \in \{1, \dots, m-1\}$, there exists a forgetful cycle f_i of length c on the region \mathbf{r}_i between $\pi_{(i)}$ and $\pi_{(i+1)}$. We define $\pi' = \phi(\pi)$ by $\phi(\pi) = \pi_{(1)}f_1\pi_{(2)} \dots \pi_{(m-1)}f_{m-1}\pi_{(m)}$. Function ϕ is an injection from $E^n = E^{mb}$ to $E^{mb+(m-1)c} = E^{n'}$ with n' defined as $n(1 + \frac{c}{b}) - c$. This injection can be extended to ε -discrete words because with lemma 10, for each couple of ε -discrete states of \mathbf{r}_i we can chose delays labeling f_i to join each other. Therefore we have a first inequality

$$|L_\varepsilon(\pi)| \leq |L_\varepsilon(\pi')|.$$

By lemma 11, for each i one can find a $\frac{1}{b+1}$ -discrete and internal run on $\pi_{(i)}$ starting from a state s_i and ending in a state s'_i . One can also find a $\frac{1}{e} = \frac{1}{(b+1)(c+1)}$ -discrete and internal run on f_i from s'_i to s_{i+1} . We have described a $\frac{1}{e}$ -discrete and internal run on π' which is thus e -fat.

Moreover an e -fat polytope is eM contiguous by remark 1. We pose $2d = eM$ and we have that π' is $2d$ contiguous.

Let $\varepsilon' = \frac{\varepsilon}{\alpha}$ where $\alpha = 1 + de\varepsilon$. $L^-(\pi')$ is defined by shrinking the inequalities defining $L(\pi')$ by the amount $\varepsilon'd$ i.e. changing $S_{j..k} \in [A, B]$ into $S_{j..k} \in [A + d\varepsilon', B - d\varepsilon']$.

Second inequality : $|L_\varepsilon(\pi')| \leq |L_{n', \varepsilon'}^-|$. To simplify we pose $N = \frac{1}{\varepsilon}$ and $N' = \frac{1}{\varepsilon'} = \alpha N$, We give an injection from $E(NL(\pi'))$ to $E(N'L^-(\pi'))$, these two sets are respectively in bijection with $L_\varepsilon(\pi)$ and $L_\varepsilon(\pi')$. As $L(\pi')$ is e -fat, there exists $(u_1, \dots, u_n) \in \mathbb{N}^n$ such that $eA + 1 \leq \sum_{i=j}^k u_i \leq eB - 1$ (where $S_{j..k} \in [A, B]$) is one of the equations defining $L(\pi')$. We multiply by d and use the fact that $de = (\alpha - 1)N$: $(\alpha - 1)NA + d \leq \sum_{i=j}^k du_i \leq (\alpha - 1)NB - d$. If (t_1, \dots, t_n) verifies $NA \leq \sum_{i=j}^k t_i \leq NB$ then $(t_1 + du_1, \dots, t_n + du_n)$ verifies $\alpha NA + d \leq \sum_{i=j}^k (t_i + du_i) \leq \alpha NB - d$ and thus it is in $E(N'L^-(\pi'))$. The translation of vector (du_1, \dots, du_n) is an injection from $E(NL(\pi'))$ to $E(N'L^-(\pi'))$ and then we are done.

Third inequality : $\varepsilon'^{n'} |L_{\varepsilon'}^-(\pi')| \leq \text{Vol}(L(\pi'))$. $\forall \mathbf{t} = (t_1, \dots, t_n) \in L_{\varepsilon'}^-(\pi'), \forall \mathbf{v} = (v_1, \dots, v_n) \in [-\frac{\varepsilon'}{2}, \frac{\varepsilon'}{2}]$, for all equations $S_{j..k} \in [A, B]$ defining $L(\pi')$, we have $A + d\varepsilon' \leq \sum_{i=j}^k (t_i + v_i) \leq B - d\varepsilon'$ and $|\sum_{i=j}^k v_i| \leq \frac{\varepsilon'}{2}(k - j + 1) \leq d$, thus $A \leq \sum_{i=j}^k (t_i + v_i) \leq B$ i.e. $\mathbf{t} + \mathbf{v} \in L(\pi')$. The set of cubes of side ε' centered in point of $L_{\varepsilon'}^-(\pi')$ has volume $\varepsilon'^{n'} |L_{\varepsilon'}^-(\pi')|$, it is contained in $L(\pi')$, this prove the third inequality.

End of the proof. Combining the three inequalities we get:

$$|L_{\varepsilon}(\pi)| \leq \left(\varepsilon'^{-n'}\right) \text{Vol}(L(\pi')).$$

Recall that $\phi : \pi \rightarrow \pi'$ is an injection and thus

$$|L_{n,\varepsilon}| = \sum_{\pi \in E^n} |L_{\varepsilon}(\pi)| \leq \left(\varepsilon'^{-n'}\right) \sum_{\pi' \in E^{n'}} \text{Vol}(L(\pi')) = \left(\varepsilon'^{-n'}\right) \text{Vol}(L_{n'}).$$

If we take $\lim_{n \rightarrow \infty} \frac{1}{n} \log_2$ in the previous inequality we obtain

$$h_{\varepsilon} \leq \left(1 + \frac{c}{b}\right) \left(\log_2 \left(\frac{1}{\varepsilon'}\right) + \mathcal{H}\right).$$

The right hand side is equal to $\log_2 \frac{1}{\varepsilon} + \mathcal{H} + \log_2 \alpha + \frac{c}{b} O(\log_2 \frac{1}{\varepsilon})$. Let $b = \varepsilon^{-\frac{1}{3}} (\log_2 \frac{1}{\varepsilon})^{\frac{1}{3}}$, then $\log_2 \alpha$ and $\frac{c}{b} \log_2 \frac{1}{\varepsilon}$ are $O\left(\varepsilon^{\frac{1}{3}} (\log_2 \frac{1}{\varepsilon})^{\frac{2}{3}}\right)$.