



HAL
open science

Max-Entropy Sampling for Deterministic Timed Automata Under Linear Duration Constraints

Benoît Barbot, Nicolas Basset

► **To cite this version:**

Benoît Barbot, Nicolas Basset. Max-Entropy Sampling for Deterministic Timed Automata Under Linear Duration Constraints. Qest23, Sep 2023, Antwerp, Belgium, Belgium. pp.188-204, <10.1007/978-3-031-43835-6_14>. <hal-04239175>

HAL Id: hal-04239175

<https://hal.science/hal-04239175v1>

Submitted on 12 Oct 2023

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.



HAL Authorization

Max-entropy sampling for deterministic timed automata under linear duration constraints^{*}

Benoît Barbot¹ and Nicolas Basset²

¹ Univ Paris Est Creteil, LACL, F-94010 Creteil, France

² Univ. Grenoble Alpes, CNRS, Grenoble INP, VERIMAG,
38000 Grenoble, France



Abstract. Adding probabilities to timed automata enables one to carry random simulation of their behaviors and provide answers with statistical guarantees to problems otherwise untractable. Thus, when just a timed language is given, the following natural question arises: *What probability should we consider if we have no a priori knowledge except the given language and the considered length (i.e. number of events) of timed words?* The maximal entropy principle tells us to take the probability measure that maximises the entropy which is the uniform measure on the language restricted to timed word of the given length (with such a uniform measure every timed word has the same chance of being sampled). The uniform sampling method developed in the last decade provides no control on the duration of sampled timed words.

In the present article we consider the problem of finding a probability measure on a timed language maximising the entropy under general linear constraints on duration and for timed words of a given length. The solution we provide generalizes to timed languages a well-known result on probability measure over the real line maximising the Shannon continuous entropy under linear constraints. After giving our general theorem for general linear constraints and for general timed languages, we concentrate to the case when only the mean duration is prescribed (and again when the length is fixed) for timed languages recognised by deterministic timed automata. For this latter case, we provide an efficient sampling algorithm we have implemented and illustrated on several examples.

1 Introduction

Since their introduction in the early 90's, Timed Automata (TA) are extensively used to model and verify the behaviors of real-timed systems and thoroughly explored from a theoretical standpoint. Several lines of research have been developed to add probabilities to these models. The two main motivations for this are (i) modelling systems that both exhibit real-time and probabilistic aspects, for example network protocols where delays are chosen at random to resolve conflicts, as in CSMA-CD (see *e.g.* [16,9]); (ii) statistical model checking of TA with the claim of replacing a prohibitively expensive exhaustive verification of the

^{*} This work was financed by the ANR MAVeriQ (ANR-20-CE25-0012)

system by a thorough random simulation of the TA with statistical guarantees (see *e.g.* [11] and reference therein).

For statistical model checking (or any other approach based on statistics) to make sense, the probability distribution defined on the runs of the TA has to be clearly given, which is not always the case as pointed in [10,2]. In the present article we follow a line of research initiated in [6,2] focusing on Deterministic Timed Automata (DTA), where the aim is to give a constructive answer to the following question inspired by Jayne’s maximal entropy principle [14]: *What probability should we specify on the runs of a given DTA without any a priori knowledge?* Another aim of this approach, which appears equivalent, is to get a sampler as uniform as possible of runs of the DTA, that is, a sampler that gives to runs of the same length the same (density of) probability. The article [6] was a first theoretical one that dealt with infinite timed words while [2] gave more pragmatic algorithms and their implementation to sample timed words of a given finite length. These two probability measures maximize Shannon continuous entropy adapted to probability measures on finite and infinite timed words respectively. The algorithms in [2] were later implemented in the tool WORDGEN [5] and applied in the context of validation of CPS by generating input signals satisfying a specification given as a DTA [4,3]. With this sampling method, one can choose the length of runs, that is, the number of events occurring in them, but not their duration.

In Section 3, we propose, as a main contribution, a maximal entropy theorem for general linear constraints on duration, that is, the expectations (over the timed words of a DTA of a given length) of some arbitrary constraint functions depending only on duration are prescribed. We were inspired by a classical maximal entropy theorem for real-to-real functions and classical Shannon continuous entropy (see the dedicated chapter of the textbook on information theory [12]). As a second main contribution we propose in Section 4, a procedure for sampling finite timed words of fixed length with random duration such that the *mean duration is prescribed*. Besides these two main contributions, we introduce, in Section 4, the key concept of Laplace Transform of Volumes (LTV) which generalizes the volume functions of [1,2] with an extra parameter. As we will discuss in the conclusion (Section 6), we think that replacing these previous volume functions by LTVs can enable us to extend several results by adding a focus on duration. Last but not least, we implemented our algorithms by extending the tool WORDGEN and in this article we provide several experiments in Section 5, one of which features a DTA with thousands of control states.

Before exposing our contributions, we recall preliminaries on timed languages and previous work on max-entropy sampling for DTA in Section 2. Due to space constraints, we move the proof of our results into an appendix.

2 Preliminaries

In this section, we recall definitions on timed languages, max-entropy probability measure on them and how to sample them.

2.1 General timed languages and their measure

Let Σ be a finite alphabet of *events*. A *timed letter* (t, a) is a couple of $\mathbb{R}^+ \times \Sigma$ where t represents the *delay* before the *event* a happens. A *timed word* \mathbf{w} is a sequence of timed letters. We denote by ε the empty timed word. For the sake of conciseness we often remove commas and parentheses and write $t_1 a_1 \cdots t_n a_n$ for $(t_1, a_1) \cdots (t_n, a_n)$. Such timed word is called of length n and of duration $\sum_{i=1}^n t_i$ denoted by $\theta(\mathbf{w})$. A *timed language* is a set of timed words. The universal language is the set of all timed words denoted by $(\mathbb{R}^+ \times \Sigma)^*$. We are mainly interested in timed languages recognized by timed automata described in Section 2.2 below but a large part of our theory can be stated for more general timed language.

For a timed language L and a timed letter ta , we define $(ta)^{-1}L$ the left derivative of L wrt ta as

$$(ta)^{-1}L = \{w \mid taw \in L\}.$$

Languages can be defined recursively via defining sub-languages $[L]_n$, $n \in \mathbb{N}$ which are the restriction to L to words of length n .

$$[L]_n = \bigcup_t \bigcup_a (ta).[(ta)^{-1}L]_{n-1} \text{ where the base case is } [L]_0 = \{\varepsilon\} \text{ or } [L]_0 = \emptyset. \quad (1)$$

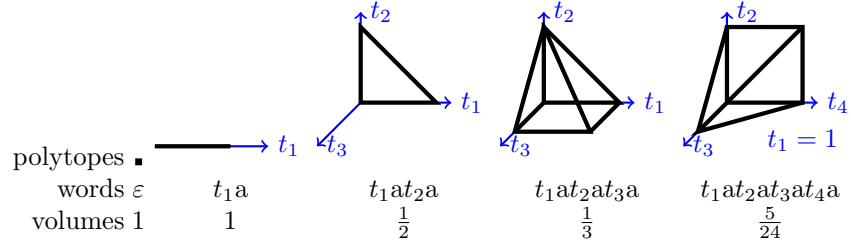
Example 1. Consider the timed language $L = \{t_1 a t_2 a \dots t_n a \mid \forall i \geq 1, t_i + t_{i+1} < 1\}$. This language satisfy $(ta)^{-1}L = \{t_2 a \dots t_n a \mid t_2 < 1 - t \wedge t_i + t_{i+1} < 1\}$ if $t < 1$ and is empty otherwise.

Timed language as a collection of sets of real-vectors Given a timed language L and $n \in \mathbb{N}$, every untimed-word $w = a_1 \cdots a_n \in \Sigma^n$ can be seen as the label for the set $\mathcal{P}_L(w) = \{(t_1, \dots, t_n) \in \mathbb{R}^n \mid t_1 a_1 \cdots t_n a_n \in L\}$ which is possibly empty.

In the following we only consider timed language for which every $\mathcal{P}_L(w)$ is a measurable set for the Lebesgue measure. In particular we mostly work with timed language for which every $\mathcal{P}_L(w)$ is a union of polytopes but our general theorem applies for the general case of measurable sets.

A polytope in dimension n is a subset of \mathbb{R}^n defined as the set of points whose coordinates satisfy a finite conjunction of linear inequalities, that is, of the form $\sum_{i=1}^n \alpha_i t_i \leq \beta$. Note we consider non-necessarily bounded polytope e.g. $\{t_1, t_2 \mid t_1 \leq 3 \wedge t_2 \geq 1 \wedge t_1 + t_2 \geq 3\}$.

Example 2. The polytopes corresponding to the language of Example 1 are depicted below with their respective volumes. Polytopes up to dimension 4 are displayed. For dimension 4 the projection of the polytope with $t_1 = 1$ is displayed.



Integrating a function over a language, volume and entropy Given a timed language L , $n \in \mathbb{N}$ and $f : L_n \rightarrow \mathbb{R}$, a real-valued function defined on L_n , we use the following notation for the integral of f over L_n .

$$\int_{L_n} f(\mathbf{w}) d\mathbf{w} = \sum_{w=a_1 \dots a_n \in \Sigma^n} \int_{(t_1, \dots, t_n) \in \mathcal{P}_L(w)} f(t_1 a_1 \dots t_n a_n) dt_1 \cdot dt_n.$$

When f is non-negative this value is either a non-negative real or $+\infty$. We say that two functions are equal almost everywhere if the set where they differ is of null measure, that is, $\int_{L_n} 1_{f(\mathbf{w}) \neq g(\mathbf{w})} d\mathbf{w} = 0$. In the rest of the paper we will just see as equals functions that are almost everywhere equals. Similarly when we say that a function f is a unique solution of a maximisation problem, it means that all functions almost-everywhere equal to f , and only them, are also a solution of the problem. These conventions will also apply to functions from \mathbb{R} to \mathbb{R} .

Volumes For a given n , L_n is the formal union of subsets $\mathcal{P}_L(w)$ of \mathbb{R}^n . The volume of L_n denoted $\text{Vol}(L_n)$ is the sum of volumes of these sets. When one is infinite then so is $\text{Vol}(L_n)$. Equivalently it is just the integral of 1 over L_n : $\text{Vol}(L_n) = \int_{L_n} 1 d\mathbf{w}$. Volumes can also be characterised recursively by

$$\text{Vol}(\emptyset) = 0, \text{Vol}(\{\varepsilon\}) = 1, \quad \text{Vol}(L_n) = \sum_{a \in \Sigma} \int_0^\infty \text{Vol}([(ta)^{-1}L]_{n-1}) dt. \quad (2)$$

PDF on timed languages A *probability density function* (PDF) over L_n is a function f non-negative for which $\int_{L_n} f(\mathbf{w}) d\mathbf{w} = 1$. Given a PDF f , and a function g defined on L_n , the expected value of g is denoted by $E_f(g)$ and defined by $\int_{L_n} g(\mathbf{w}) f(\mathbf{w}) d\mathbf{w}$. We will be mostly interested with the expected duration of timed words of a given language L and PDF f : $E_f(\theta) = \int_{L_n} \theta(\mathbf{w}) f(\mathbf{w}) d\mathbf{w}$.

Entropy of a PDF The *entropy* of a PDF f on L_n is³

$$H(f) = E_f(-\log_2(f)) = - \int_{L_n} f(\mathbf{w}) \log_2 f(\mathbf{w}) d\mathbf{w}, \quad (3)$$

This definition is the natural generalisation to timed languages of the Shannon continuous entropy (aka. Shannon differential entropy) for functions from \mathbb{R} to \mathbb{R} . The following max-entropy theorem, tells us that the uniform PDF (used in [2] for uniform random sampling) is the unique PDF that maximises the entropy.

³ In this definition the usual convention that $0 \log_2 0 = 1$ applies.

Theorem 1. *Given a timed language L and $n \in \mathbb{N}$ such that $\text{Vol}(L_n) < +\infty$, the maximal value of the entropy of a PDF on L_n is $\log_2(\text{Vol}(L_n))$. It is reached by a unique PDF on L_n which is the uniform PDF on L_n : the constant function $\mathbf{w} \mapsto 1/\text{Vol}(L_n)$.*

This theorem will be a particular case of our main theorem (Theorem 2 of Section 3) where no constraint is given on duration.

Maximal entropy measure and uniform sampling for general timed languages Uniform sampling can be defined over general timed languages based on Eq. (2). Exact sampling in L of a word of length n is performed iteratively. When n is equal to 0 the sampling stops otherwise the next letter $a \in \Sigma$ is chosen with probability $\frac{\int_0^\infty \text{Vol}([(ta)^{-1}L]_{n-1})dt}{\text{Vol}(L_n)}$. Next a delay t is sampled⁴ from the continuous distribution defined by the Probability Density function $t \mapsto \frac{\text{Vol}([(ta)^{-1}L]_{n-1})}{\int_0^\infty \text{Vol}([(ta)^{-1}L]_{n-1})}$. Then a timed word of length $n - 1$ is sampled from the language $(ta)^{-1}L$. This scheme needs a practical way of computing PDF which will be the case for timed languages recognized by deterministic timed automata.

2.2 Languages recognized by DTA and their measures

In this part we restrict our attention to languages recognised by a DTA. For these languages, we can describe computation of the volume of the language with recursive equations on the structure of the DTA recognising the language.

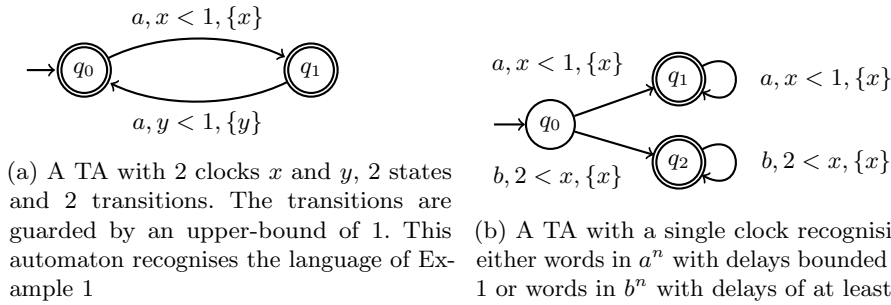


Fig. 1: Examples of Deterministic Timed Automata

Timed automata A *timed automaton* (TA) \mathcal{A} is defined as a tuple $(\Sigma, X, Q, q_0, \mathcal{F}, \Delta)$ where Σ is a finite set of events; X is a finite set of *clocks*; Q is a finite set of *locations*; q_0 is the initial location; $\mathcal{F} \subseteq Q$ is a set of final locations; and Δ is a finite set of *transitions*. The finite set of clocks X is a finite set of non-negative

⁴ One could also sample the delay before the action, this would lead at the end to the same probability distribution on timed words.

real-valued variables. A *guard* is a finite conjunction of clock constraints. For a clock vector $\mathbf{x} \in [0, \infty[^X$ and a non-negative real t , we denote by $\mathbf{x} + t$ the vector $\mathbf{x} + (t, \dots, t)$. A transition $\delta = (\delta^-, a_\delta, \mathbf{g}_\delta, \mathbf{r}_\delta, \delta^+) \in \Delta$ has an *origin* $\delta^- \in Q$, a *destination* $\delta^+ \in Q$, a label $a_\delta \in \Sigma$, a *guard* \mathbf{g}_δ and a *reset* function \mathbf{r}_δ determined by a subset of clocks $B \subseteq X$. To simplify notation \mathbf{r} are overloaded to be both a subset of the clocks set and the function assigning 0 to clocks in the subset. Fig. 1a depicts a TA with two clocks x and y , two locations q_0, q_1 which are final and two transitions. The transition from q_0 to q_1 is labelled by a is guarded by $x < 1$ and reset x while the transition from q_1 to q_0 is labelled by a is guarded by $y < 1$ and reset y . Other examples of TAs are given in Fig. 1b and Fig. 4.

A *state* $s = (q, \mathbf{x}) \in Q \times [0, \infty[^X$ is a pair of location and a clock vector. The initial state of \mathcal{A} is $(q_0, \mathbf{0})$. A *timed transition* is a pair (t, δ) of a time *delay* $t \in [0, \infty[$ followed by a discrete transition $\delta \in \Delta$. The delay t represents the time before firing the transition δ .

A run is an alternating sequence $(q_0, \mathbf{x}_0) \xrightarrow{t_1, \delta_1} (q_1, \mathbf{x}_1) \dots \xrightarrow{t_n, \delta_n} (q_n, \mathbf{x}_n)$ of states satisfying that q_i is the successor of q_{i-1} by δ_i , the vector $\mathbf{x}_{i-1} + t$ satisfies the guard \mathbf{g}_δ and $\mathbf{x}_i = \mathbf{r}_\delta(\mathbf{x}_{i-1} + t)$, (q_0, \mathbf{x}_0) is the initial state and q_n is a final location. This run is *labelled* by the *timed word* $(t_1, a_1) \dots (t_n, a_n)$ where for every $i \leq n$, a_i is the label of δ_i . The set of timed words that label all the runs is called the *timed language* of \mathcal{A} . The TA of Fig. 1a recognises the language of Ex. 1.

A *deterministic* timed automaton (DTA) is a TA where for every location q , transitions starting from q have pairwise disjoint guards or different labels. All TA shown in this paper are DTA.

Max-entropy sampling for DTA In [2,3,4,5] algorithms were developed to compute the volume for languages of DTA with some restrictions. Given a DTA, (2) can be instantiated over its zone graph. The first step is to compute the forward-reachability zone graph. The second operation splits the zone graph such that, for any vertex of the graph, for each available transition δ , there exist two functions \mathbf{lb}_δ and \mathbf{ub}_δ of the clock vector \mathbf{x} such that, for all $t \in \mathbb{R}^+$, $\mathbf{x} + t \models \mathbf{g}_\delta$ is equivalent to $t \in [\mathbf{lb}_\delta(\mathbf{x}), \mathbf{ub}_\delta(\mathbf{x})]$. Moreover the bounds $\mathbf{lb}_\delta(\mathbf{x})$ and $\mathbf{ub}_\delta(\mathbf{x})$ are of the form $c - x$ where $c \in \mathbb{N}$ and x is a clock or zero. Details on the computation of the split operation can be found in [2]. In the split zone graph equation (2) becomes:

$$v_0(l, \mathbf{x}) = 1_{l \in F} \text{ and } v_n(l, \mathbf{x}) = \sum_{\delta^- = l} \int_{\mathbf{lb}_\delta(\mathbf{x})}^{\mathbf{ub}_\delta(\mathbf{x})} v_{n-1}(\delta^+, \mathbf{r}_\delta(\mathbf{x} + t)) dt \quad (4)$$

Given n , $v_n(l, \mathbf{x})$ can be computed in polynomial time with respect to n and the split zone graph size which in the worst case can be exponentially large in the number of clocks. Given n and location l the functions $\mathbf{x} \mapsto v_n(l, \mathbf{x})$ is a polynomial in \mathbf{x} . This can be seen by a straightforward recursion since v_0 is a constant and the bound of the integrals to define v_{n+1} from v_n are polynomials.

Since closed-form formulae for v_n are efficiently computable, this provides an effective way of sampling using the procedure presented in Section 2.1.

More precisely the k^{th} letter is randomly chosen in state (q, \mathbf{x}) by choosing the transition $\delta = (q, a, \mathbf{g}_\delta, \mathbf{r}_\delta, q')$ labelling a with the discrete probability distribution

$$p_k(\delta | q, \mathbf{x}) = \frac{\int_{\mathbf{1b}_\delta(\mathbf{x})}^{\mathbf{ub}_\delta(\mathbf{x})} v_{n-k-1}(q', \mathbf{r}_\delta(\mathbf{x} + t)) dt}{v_{n-k}(q, \mathbf{x})} \quad (5)$$

and the delay with the PDF:

$$p_k(t | a, q, \mathbf{x}) = \frac{\mathbf{1}_{t \in (\mathbf{1b}_\delta, \mathbf{ub}_\delta)} v_{n-k-1}(q', \mathbf{r}_\delta(\mathbf{x} + t))}{\int_{\mathbf{1b}_\delta(\mathbf{x})}^{\mathbf{ub}_\delta(\mathbf{x})} v_{n-k-1}(q', \mathbf{r}_\delta(\mathbf{x} + t)) dt} \quad (6)$$

Approximate uniform sampling method are presented in [2] to sample words of large length. One of such method is the *receding sampling* where to avoid computing the p_k in (5) and (6) when k is close to a too large sampling length m , one use instead $p_{\min(n,k)}$ for a n large enough but small compared to m (see [2] for the details and formal statements).

3 Max-Entropy Theorem Under Linear Duration Constraints

In previous work on uniform sampling, constraint on duration could not be expressed. Here we are interested to express for instance that the mean and the variance of duration are equals to μ and ν , we write $\int_{L_n} \theta(\mathbf{w}) p(\mathbf{w}) d\mathbf{w} = \mu$ and $\int_{L_n} \theta^2(\mathbf{w}) p(\mathbf{w}) d\mathbf{w} = \nu + \mu^2$.

More generally we are given m measurable functions $f_i : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$ and constants $a_i \in \mathbb{R}$, $i = 1..m$, and we address the following max-entropy problem:

$$\text{Maximise } H(p) \text{ under constraints } \int_{L_n} f_i(\theta(\mathbf{w})) p(\mathbf{w}) d\mathbf{w} = a_i \text{ for } i \in \{0, \dots, m\}, \quad (7)$$

where f_0 is the constant 1 and $a_0 = 1$, to add the constraints that p is a PDF.

Our solution given in the max-entropy theorem below (Theorem 2) is based on expressing this max-entropy problem on L_n as a max-entropy problem for functions of the reals (that take the duration as argument). Before stating this theorem, we first need few definitions and Lemmas.

We say that a function f defined on L_n *depends only on duration* if $f = \tilde{f} \circ \theta$ for some functions on the reals \tilde{f} , that is, for every $\mathbf{w} \in L_n$, $f(\mathbf{w}) = \tilde{f}(\theta(\mathbf{w}))$. For a PDF f , "depending only on duration" can be rephrased as "being uniform at fixed duration": for every T , all the timed words of duration T have the same density of probability.

With Lemma 1 we show that integrating a function that depends only on duration, i.e. of the form $\tilde{f} \circ \theta$, can be expressed as integrating $\tilde{f}(T)$ with the duration T as the variable of integration multiplied by a weight denoted $V_n^L(T)$.

This weight $V_n^L(T)$ is the $n - 1$ -dimensional⁵ volume of timed-word of length n restricted to duration T defined by

$$V_n^L(T) = \sum_{w_1 \dots w_n \in \Sigma^n} \int_{(\mathbb{R}_{\geq 0})^{n-1}} \mathbf{1}_{t_1 w_1 \dots t_{n-1} w_{n-1} (T - t_1 - \dots - t_{n-1}) w_n \in L} dt_1 \dots dt_{n-1}. \quad (8)$$

We will denote $V_n^L(T)$ by $V_n(T)$ when L is clear from the context. We will also use the following notation: $L_{n,T} = \{\mathbf{w} \in L_n \mid \theta(\mathbf{w}) = T\}$ and consider $(n - 1)$ -dimensional integral on it:

$$\int_{L_{n,T}} f(\mathbf{w}) d\mathbf{w} = \sum_{w_1 \dots w_n \in \Sigma^n} \int_{(\mathbb{R}_{\geq 0})^{n-1}} \mathbf{1}_{\mathbf{w} \in L_{n,T}} f(\mathbf{w}) dt_1 \dots dt_{n-1}.$$

In particular $V_n^L(T) = \int_{L_{n,T}} 1 d\mathbf{w}$ and hence $\mathbf{w} \mapsto 1/V_n^L(T)$ is the uniform PDF on $L_{n,T}$

Lemma 1. *If $f = \tilde{f} \circ \theta$ then $\int_{L_n} f(\mathbf{w}) d\mathbf{w} = \int_0^{+\infty} \tilde{f}(T) V_n^L(T) dT$*

Remark 1. We give few cases of interest for this Lemma.

1. If $f(\mathbf{w}) = 1$, then we get $\text{Vol}(L_n) = \int_0^{+\infty} V_n(T) dT$
2. If $f(\mathbf{w}) = \frac{1}{\text{Vol}(L_n)}$, then $T \mapsto \frac{V_n(T)}{\text{Vol}(L_n)}$ is a PDF on reals.
3. If $f(\mathbf{w}) = \frac{\theta(\mathbf{w})}{\text{Vol}(L_n)}$ then $\int_{L_n} \frac{\theta(\mathbf{w})}{\text{Vol}(L_n)} d\mathbf{w} = \int_0^{+\infty} T \frac{V_n(T)}{\text{Vol}(L_n)} dT$ is the mean duration for the uniform distribution on L_n .

A straightforward consequence of Lemma 1, called Lemma 2 below, is that the max-entropy problem is equivalent to a max-entropy problem for functions defined on the reals. In this Lemma $p \ll V_n$ means that $\forall T \in \mathbb{R}_{\geq 0}$, $V_n(T) = 0 \Rightarrow p(T) = 0$, and

$$H_{V_n}(q) = - \int_0^{+\infty} q(T) \log_2 \frac{q(T)}{V_n(T)} dT.$$

Lemma 2. *For every PDF of the form $\tilde{p} \circ \theta$, the function $p^{\mathbb{R}} : T \mapsto V_n(T) \tilde{p}(T)$ is a PDF such that $H_{V_n}(p^{\mathbb{R}}) = H(p \circ \theta)$. Moreover $\tilde{p} \circ \theta$ is a solution of (7) iff $p^{\mathbb{R}}$ is a solution of*

$$\text{Maximise } H_{V_n}(p) \text{ s.t. } \int_0^{+\infty} f_i(T) p(T) dT = a_i \text{ for } i \in \{0, \dots, m\} \text{ and } p \ll V_n. \quad (9)$$

With Lemma 3 we show that given any PDF, one can always find another PDF with higher or equal entropy that depends only on duration.

⁵ The timed words of duration T have their timed vector (t_1, \dots, t_n) belonging to the hyperplane $t_1 + \dots + t_n = T$ which have a null volume. That is why we do not integrate over the last delay which is fixed and equal to $T - t_1 - \dots - t_{n-1}$

Lemma 3. Let p be a PDF on L_n and⁶ $\tilde{p} : T \mapsto \frac{1}{V_n(T)} \int_{L_{n,T}} p(\mathbf{w}) d\mathbf{w}$ then $H(p) \leq H(\tilde{p} \circ \theta)$ with equality if and only if $p = \tilde{p} \circ \theta$.

Theorem 2 (Maximal entropy theorem on L_n with integral constraints on duration). Given the max-entropy problem (7) with its constraints functions f_i and constants a_i . If one find constants $\lambda_i, (0 \leq i \leq m)$ such that the function $p^*(\mathbf{w}) = \exp(\lambda_0 + \sum_{i=1}^m \lambda_i f_i(\theta(\mathbf{w})))$ is a PDF that satisfies the constraints, then the problem admits a unique solution which is p^* .

Proof. Let p^* and λ_i as above so that p^* satisfies the constraints, then we have to prove that $H(p^*)$ is maximal and unique for this property. First, Lemma 3 implies that we only have to consider PDF q of the form $q = \tilde{q} \circ \theta$. We show that such PDF q satisfies $H(q) \leq H(p^*)$ with equality iff $q = p^*$. This is equivalent to show that $H_{V_n}(q^{\mathbb{R}}) \leq H_{V_n}(p^{\mathbb{R}})$ with equality iff $q^{\mathbb{R}} = p^{\mathbb{R}}$, with $p^{\mathbb{R}} = \tilde{p} \cdot V_n$ and $q^{\mathbb{R}} = \tilde{q} \cdot V_n$ since by Lemma 2, $H_{V_n}(p^{\mathbb{R}}) = H(p^*)$ and $H_{V_n}(q^{\mathbb{R}}) = H(q)$.

We adapt a classical proof based on Kulback-Lebler divergence that can be found in [12]. The Kullback–Leibler divergence from $p^{\mathbb{R}}$ to $q^{\mathbb{R}}$ is the quantity $D(q^{\mathbb{R}}||p^{\mathbb{R}}) = \int_0^{+\infty} q^{\mathbb{R}}(T) \log \frac{q^{\mathbb{R}}(T)}{p^{\mathbb{R}}(T)} dT$ which is always non-negative, and null iff $p^{\mathbb{R}} = q^{\mathbb{R}}$. We will show that $D(q^{\mathbb{R}}||p^{\mathbb{R}}) = H_{V_n}(p^{\mathbb{R}}) - H_{V_n}(q^{\mathbb{R}})$ and thus $H_{V_n}(p^{\mathbb{R}}) = H_{V_n}(q^{\mathbb{R}})$ will be equivalent to $p^{\mathbb{R}} = q^{\mathbb{R}}$.

We introduce $V_n(T)$ using the fact that $\log_2(q^{\mathbb{R}}/p^{\mathbb{R}}) = \log_2 \frac{q^{\mathbb{R}}(T)}{V_n(T)} - \log_2 \frac{p^{\mathbb{R}}(T)}{V_n(T)}$:

$$D(q^{\mathbb{R}}||p^{\mathbb{R}}) = \int_0^{+\infty} q^{\mathbb{R}}(T) \log_2 \frac{q^{\mathbb{R}}(T)}{V_n(T)} dT - \int_0^{+\infty} q^{\mathbb{R}}(T) \log_2 \frac{p^{\mathbb{R}}(T)}{V_n(T)} dT$$

The first integral is $-H_{V_n}(q^{\mathbb{R}})$, hence it remains to prove that $H_{V_n}(p^{\mathbb{R}}) = -\int q^{\mathbb{R}}(T) \log_2 \left(\frac{p^{\mathbb{R}}(T)}{V_n(T)} \right)$. We recall that $\frac{p^{\mathbb{R}}(T)}{V_n(T)} = \tilde{p}(T) = \exp(\sum_{i=0}^m \lambda_i f_i(T))$, thus

$$\begin{aligned} \int_0^{+\infty} q^{\mathbb{R}}(T) \log_2 \left(\frac{p^{\mathbb{R}}(T)}{V_n(T)} \right) &= \frac{1}{\ln 2} \int_0^{+\infty} q^{\mathbb{R}}(T) \left(\sum_{i=0}^m \lambda_i f_i(T) \right) dT \\ &= \frac{1}{\ln 2} \sum_{i=0}^m \lambda_i \int_0^{+\infty} q^{\mathbb{R}}(T) f_i(T) dT. \end{aligned}$$

Since $q^{\mathbb{R}}$ satisfies the constraints this quantity is equal to $\frac{1}{\ln 2} \sum_{i=0}^m \lambda_i a_i$. The same reasoning holds for p , so we can conclude the proof with the equality:

$$H_{V_n}(p^{\mathbb{R}}) = -\frac{1}{\ln(2)} \sum_{i=0}^m \lambda_i a_i = -\int_0^{+\infty} q^{\mathbb{R}}(T) \log_2 \left(\frac{p^{\mathbb{R}}(T)}{V_n(T)} \right) dT.$$

Theorem 1 is a specialisation of Theorem 2 where there is no constraint on duration. Indeed the function has to be searched as a constant e^{λ_0} with the unique constraint that $\int_{L_n} e^{\lambda_0} d\mathbf{w} = 1$, that is, $e^{\lambda_0} = 1/\text{Vol}(L_n)$. As there is no constraint on duration the volume must be finite for this theorem to apply.

⁶ Here \tilde{p} generalises the previous concept it was used for: when p depends only on duration then $p = \tilde{p} \circ \theta$.

4 Max-entropy theorem with prescribed mean duration

In this section we describe and see how to compute the max-entropy PDF when the mean duration is prescribed. So we will use Theorem 2 with f_1 being the identity function and explain the normalising constant (link between λ_0 and λ_1).

Theorem 3 (Maximal entropy theorem for measures on L_n with prescribed mean duration). *Given a timed language L , length n , duration T_{mean} . If there exist probability measures p that satisfy the constraint $E_p(\theta(w)) = T_{mean}$, then there is a unique one p^* satisfying this constraint that maximizes the entropy. It is given by*

$$p^*(w) = \frac{e^{-s\theta(w)}}{v_{n,s}^L}$$

where the normalising constant $v_{n,s}^L$ is defined by

$$v_{n,s}^L = \int_{L_n} e^{-s\theta(\mathbf{w})} d\mathbf{w} = \int_0^{+\infty} e^{-sT} V_n^L(T) dT \quad (10)$$

with s the unique real such that $\frac{1}{v_{n,s}^L} \frac{\partial v_{n,s}^L}{\partial s} = T_{mean}$.

The normalising constant $v_{n,s}^L$ defined in equation (10) can be interpreted as a Laplace transform of the function V_n and hence we call it the *Laplace Transform of Volumes* (LTV) and s is called the *Laplace parameter*. In the next section we will propose efficient computation of it which will be the base of a sampling algorithm. As shown above by tuning the Laplace parameter s we can control the mean duration which is defined wrt. the LTV and its derivative. Controlling the variance could be done by adding an extra parameter. Efficient sampling for this latter case is left for future work, and we focus on fixing only the mean duration so working only with the LTV. Once this done, we can characterise the variance (without prescribing it) with the LTV and its derivatives.

Proposition 1 (Characterization of the variance wrt. LTV and its derivatives). *Let p^* be the PDF given in Theorem 3 and let $Var_{p^*}(\theta(\mathbf{w})) = E_{p^*}(\theta(\mathbf{w})^2) - E_{p^*}(\theta(\mathbf{w}))^2$ the variance of the duration of timed words sampled with this probability. Then*

$$Var_{p^*}(\theta(\mathbf{w})) = \frac{\partial^2 v_{n,s}^L}{\partial s^2} - \left(\frac{1}{v_{n,s}^L} \frac{\partial v_{n,s}^L}{\partial s} \right)^2.$$

Remark 2. Theorem 1 is again a special case. It suffices to take $s = 0$. We get in addition, a characterisation of the mean of the duration and with Proposition 1 its variance.

The LTV v_n can be characterized via a recursive definition:

Proposition 2 (Recursive definition of v_n).

$$v_{n,s}^L = \sum_{a \in \Sigma} \left(\int_0^{+\infty} e^{-st} v_{n-1,s}^{(ta)^{-1}L} dt \right) \quad (11)$$

This recursive definition is a step towards dynamic programming computation that will be fully possible now when we focus on DTA.

4.1 Computing LTV and max-entropy sampling for Timed Automata

Theorem 3 tells us the form of the maximal entropy PDF when a mean duration constraint is imposed. In this section we focus on how to sample this PDF for language recognized by a DTA using the LTVs for languages starting on states of this DTA. This PDF is then used to sample its language.

Computation of LTV as exponential Polynomials for DTA In this section, we describe how LTV can be computed effectively from a timed automaton. This is done by computing LTV recursively and by showing that all computations are performed over exponential polynomials (Def. 1) for which the computation of integral is effective. Here we find a suitable generalisation of the volumes functions used in [1,6,2] and recall in Section 2. In a nutshell, we multiply by e^{-st} the function which is integrated over the next transition with the delay as integration variable.

In the following we assume that a DTA in split form is given and for every length n , parameter s and state (l, \mathbf{x}) we denote by $v_{n,s}(l, \mathbf{x})$ the LTV of the language starting from (l, \mathbf{x}) that is $v_{n,s}(q, \mathbf{x}) = v_{n,s}^{L(l, \mathbf{x})}$.

Definition 1 (Exponential Polynomials). *Given a variable s and a sequence of variable $\mathbf{X} = (X_i)_{i=1}^n$, we call exponential polynomials, expressions $EP(s, \mathbf{X})$ such that : there exists a finite set $S \subset \{0, 1, \dots, n\} \times \mathbb{N}$ and a polynomial $P_{i,k}$ for each element of S with $P_{i,k}$ a polynomials over $\mathbf{X} \cup \{\frac{1}{s}\}$ with the the convention that $X_0 = 0$; and $EP(s, \mathbf{X})$ is written in the form*

$$EP(s, \mathbf{X}) = \sum_{(i,k) \in S} P_{i,k} \left(\frac{1}{s}, \mathbf{X} \right) e^{s(X_i - k)}$$

Proposition 3. *For every location q and length n , $\mathbf{x} \mapsto v_{n,s}(q, \mathbf{x})$ is an exponential polynomials. It can be computed in polynomial time using dynamic programming from the following recursive equations:*

$$v_{0,s}(l, \mathbf{x}) = 1_{l \in F} \text{ and } v_{n,s}(l, \mathbf{x}) = \sum_{\delta = -l} \int_{\text{lb}_\delta(\mathbf{x})}^{\text{ub}_\delta(\mathbf{x})} v_{n-1,s}(\delta^+, \mathbf{r}_\delta(\mathbf{x} + t)) e^{-st} dt. \quad (12)$$

Random sampling using the LTV

The sampling follows the same line as in for the unconstrained case (Section 2.1) where the probability distributions for discrete and continuous choice (5) and (6) are replaced by

$$p_{k,s}(a \mid q, \mathbf{x}) = \frac{\int_{\mathbf{1b}_\delta(\mathbf{x})}^{\mathbf{ub}_\delta(\mathbf{x})} v_{n-k-1,s}(q', \mathbf{r}_\delta(\mathbf{x} + t)) e^{-st} dt}{v_{n-k,s}(q, \mathbf{x})}; \text{ and} \quad (13)$$

$$p_{k,s}(t \mid a, q, \mathbf{x}) = \frac{1_{(\mathbf{1b}_\delta, \mathbf{ub}_\delta)} v_{n-k-1,s}(q', \mathbf{r}_\delta(\mathbf{x} + t)) e^{-st}}{\int_{\mathbf{1b}_\delta(\mathbf{x})}^{\mathbf{ub}_\delta(\mathbf{x})} v_{n-k-1,s}(q', \mathbf{r}_\delta(\mathbf{x} + t)) e^{-st} dt}. \quad (14)$$

5 Experiments

We have implemented our approach in the tool WORDGEN [5], which required developing a data structure for exponential polynomials, implementing the LTV computation and the estimation of parameters s . These developments are freely available with the GPLv3 licence. This section contains experiments on small timed automata (Example 3 and 4) and a case study on a larger automaton (Example 5).

Example 3. We compute the LTVs of the language defined in Example 1 recognized by the timed automaton depicted in Figure 1a. After computing the split zone graph, the number of locations is still 2. We show the LTVs computed for small n in both of these locations and in the initial one in the following table.

$(q, \mathbf{x}) \setminus n$	0	1	2
$(q_0, (x, y))$	1	$\frac{1 - e^{-s(1-x)}}{s}$	$\frac{1 - e^{-s(1-x)}}{s^2} + \frac{(x-1)}{s} e^{-s}$
$(q_1, (x, y))$	1	$\frac{1 - e^{-s(1-y)}}{s}$	$\frac{1 - e^{-s(1-y)}}{s^2} + \frac{(y-1)}{s} e^{-s}$

If we restrict to timed words of length 2, sampling with the parameter $s = 0$ (using Taylor expansion) provides a uniform distribution in the triangle. The expected duration is $\frac{1}{v_{2,s}(q_0, (0,0))} \frac{\partial v_{2,s}}{\partial s}(q_0, (0,0))$. For $s \neq 0$ it is $\frac{2+2s+s^2-2e^s}{s+s^2-se^s}$. Using Taylor expansions in $s = 0$ of both the numerator and the denominator one can show that the limit when $s \rightarrow 0$ is $\frac{2}{3}$.

Figure 2 shows plots of the sampling from this language with different mean durations. We observe a shift of the concentration of points along the axis $y = x$. Note that the time duration of 0.9 is obtained with a negative value of s . This is allowed when all clocks are bounded and thus $V_n(T)$.

Example 4. In this example, we are interested in the language recognized by the automaton of Figure 1b. We want to generate words of length 3 ($n = 3$) with a mean time duration of 5.5. There are only two possible untimed words which are aaa and bbb . One can see that their corresponding polytopes are $[0, 1]^3$ and $(2, +\infty)^3$. In particular, timed words are either of duration < 3 or > 6 , so there are no timed words of the target duration 5.5. However, a mean duration of 5.5 can be reached by computing and taking the appropriate value for the parameter s , that is, 0.52. By sampling 100 000 words with this parameter, we obtain the histogram depicted in Figure 3 with an average duration of 5.50116.

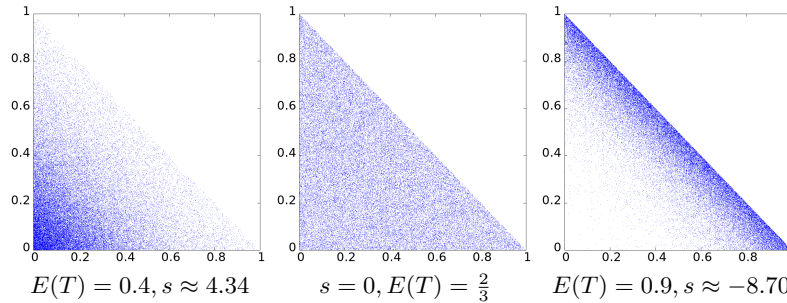


Fig. 2: Sample 50000 timed words t_1at_2a from the language of Example 1 (the DTA is in Figure 1a) with t_1 in the abscissa and t_2 in the ordinate.

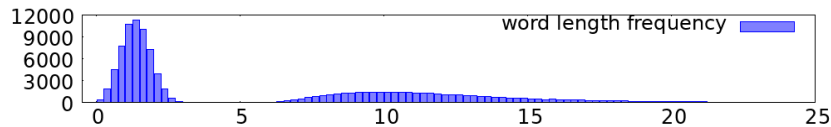


Fig. 3: Histogram of duration of 100000 words sampled from the automaton in Figure 1b.

Example 5 (Train Gate). We illustrate the scalability of our method using the well-known Train Gate example described extensively in the Uppaal Tutorial [8]. This example does not involve a single automaton but a network of timed automata synchronized by their transitions. The semantics of such a network is a single timed automaton whose state space is the cross product of each automaton. This synchronisation semantics is explained in [8] and implemented in the tool Uppaal, thus we will use the same notation as in this tool.

In this model, N trains want to cross a one-way bridge repeatedly. Each train is modeled as a timed automaton with one clock. The bridge is modeled as a FIFO scheduler, which stops trains or clears them for crossing. The timed automaton for each train is depicted in Figure 4(left) where id is instantiated as an identifier for the train. This automaton is replicated N times with an independent copy of clock x . The clock x is reset on every edge (omitted in the figure). Note that the states Safe and Stop are unbounded. Figure 4(right) depicts the scheduler whose size depends on the number of trains. The FIFO queue is implemented by an array of identifiers of size N and four functions `pop`, `push`, `front` and `last` that modify it.

In Table 1 we show the performance of the tool WORDGEN in sampling a thousand runs of this automaton of expected duration 50. The number N of trains and parameter n are shown in the first two columns. As there is one clock per train, the number of zones grows exponentially with N , while the splitting only double the number of zones. The computation time is dominated by the computation of $v_{n,s}(l, \mathbf{x})$. The sampling time is proportional to the number of trajectories and requires the evaluation of large polynomials for the sampling of timed distributions. As expected, the computation of the zone graph and split

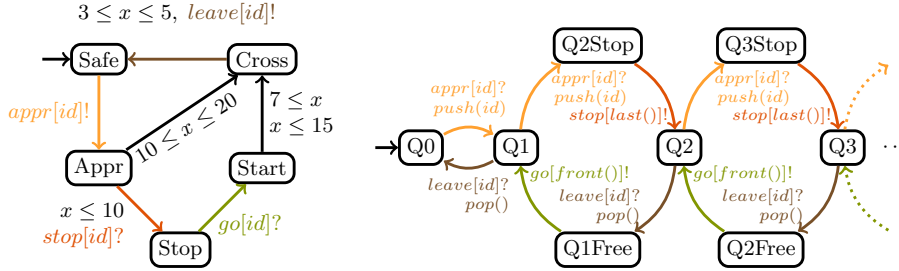


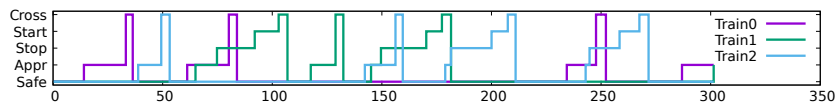
Fig. 4: The automaton on the left depicts the timed automaton for each train where the clock x is reset on every edge. The automaton on the right depicts the scheduler.

zone graph is small, compared to the other computations. The memory appears to be a computational bottleneck as the number of terms in each $v_{n,s}(l, \mathbf{x})$ grows exponentially with the number of clocks and they need to be computed on each location.

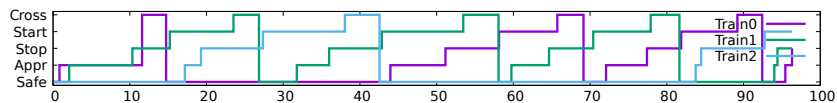
N	n	number of locations		time (sec)			memory
		reachability	split	reach+split	distribution	sampling	
1	10	9	9	0	0	0.07	6.2 MB
2	10	51	70	0.01	0	0.13	34.8 MB
3	10	1081	1826	0.12	9.04	2.6	1.7 GB
4	10	28353	53534	7.2	647	466	51.9 GB
5	10	88473	178414	35	Out of Memory > 64GB		
5	7	88473	178414	35	408	205	52.4 GB

Table 1: Each experiment samples 1000 timed words of length n and expected duration 50

By changing the expected duration of timed words, we observe very different behaviors of the system. Figure 5 depicts two timed words sampled from the Train Gate example with 3 trains and $n = 10$ and of length of length $m = 40$ using receding sampling. Figure 5(a) depicts a timed word sampled with expected duration 80 while Figure 5(b) depicts a timed word sampled with expected time duration 240. We can see that in Figure 5(b) the system is saturated since there is almost always a train in the locations Start or Cross which have a lower bound on their waiting time, and the synchronisation blocks other events. In Figure 5(a) we observe that the system stay in a state where all trains are in location Safe for some time where there are no constraints (no lower nor upper bound) thus trains evolve more independently.



(a) Timed words with an expected duration of 240 time units and 40 events ($s \approx 0.1098$).



(b) Timed words with an expected duration of 80 time units and 40 events ($s \approx 0.9070$).

Fig. 5: Timed words sampled from the Train Gate example. The state of trains are in ordinates.

6 Conclusion and Future work

In this article, we address the problem of characterizing probability measures of maximal entropy for timed words of a fixed length of timed languages under duration constraints. We focus our attention on constraint on mean duration, which leads us to define and propose a method for efficiently computing Laplace Transform of Volumes, a key theoretical tool to sample timed word wrt. the maximal entropy measure. Several experiments are provided to illustrate our sampling algorithm.

Ongoing and future work The LTV we compute are Laplace transforms of functions $V_n(T)$ which can be called the volume of timed words with fixed duration T . These functions are crucial in our theoretical development of Section 3 but are not computed in this current article. An ongoing work is to write recursive characterization of such functions that could be turned into a sampler of timed words of exact duration T . The computation of closed form formulae for $V_n(T)$ could be used to sample the PDF of the form $p^{\mathbb{R}}(T) = \tilde{p}(T)V_n(T)$, which, coupled with a uniform sampler of exact duration T , would provide a max-entropy sampler for the general linear constraints of Theorem 2. Another promising research direction is to revisit the results of [6,1] with the operator underlying the definition of the LTV. With this approach we aim at defining maximal entropy measure for infinite timed words with a prescribed frequency of event, *e.g.* 0.7 events per time unit. Finally, another possible extension is to consider max-entropy measures when the length is also random. For this we would like to adapt Boltzmann sampling algorithms ([13]) to our settings. A first adaptation of such a sampling for a very particular subclass of timed languages (used for random sampling of permutations) was proposed in [7]. This latter work though was not concerned at all with duration of timed words (nor entropy).

References

1. Asarin, E., Basset, N., Degorre, A.: Entropy of regular timed languages. *Information and Computation* **241**, 142–176 (2015)
2. Barbot, B., Basset, N., Beunardeau, M., Kwiatkowska, M.: Uniform sampling for timed automata with application to language inclusion measurement. In: QEST 2016, Quebec City, QC, Canada, August 23–25, 2016. pp. 175–190 (2016)
3. Barbot, B., Basset, N., Dang, T.: Generation of signals under temporal constraints for CPS testing. In: NASA Formal Methods - 11th International Symposium, NFM 2019. pp. 54–70 (2019)
4. Barbot, B., Basset, N., Dang, T., Donzé, A., Kapinski, J., Yamaguchi, T.: Falsification of cyber-physical systems with constrained signal spaces. In: NASA Formal Methods - 12th International Symposium, NFM, 2020. pp. 420–439 (2020)
5. Barbot, B., Basset, N., Donze, A.: Wordgen : A timed word generation tool. In: Proceedings of the 26th ACM International Conference on Hybrid Systems: Computation and Control. HSCC '23, Association for Computing Machinery, New York, NY, USA (2023). <https://doi.org/10.1145/3575870.3587116>, <https://doi.org/10.1145/3575870.3587116>
6. Basset, N.: A maximal entropy stochastic process for a timed automaton. *Inf. Comput.* **243**, 50–74 (2015). <https://doi.org/10.1016/j.ic.2014.12.006>, <https://doi.org/10.1016/j.ic.2014.12.006>
7. Basset, N.: Counting and generating permutations in regular classes. *Algorithmica* **76**(4), 989–1034 (2016). <https://doi.org/10.1007/s00453-016-0136-9>, <https://doi.org/10.1007/s00453-016-0136-9>
8. Behrmann, G., David, A., Larsen, K.G.: A tutorial on UPPAAL. In: Bernardo, M., Corradini, F. (eds.) *Formal Methods for the Design of Real-Time Systems: 4th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM-RT 2004*. pp. 200–236. No. 3185 in LNCS, Springer-Verlag (September 2004)
9. Bertrand, N., Bouyer, P., Brihaye, T., Menet, Q., Baier, C., Größer, M., Jurdzinski, M.: Stochastic timed automata. *Log. Methods Comput. Sci.* **10**(4) (2014). [https://doi.org/10.2168/LMCS-10\(4:6\)2014](https://doi.org/10.2168/LMCS-10(4:6)2014), [https://doi.org/10.2168/LMCS-10\(4:6\)2014](https://doi.org/10.2168/LMCS-10(4:6)2014)
10. Bohlender, D., Bruintjes, H., Junges, S., Katelaan, J., Nguyen, V.Y., Noll, T.: A review of statistical model checking pitfalls on real-time stochastic models. In: ISOLA. pp. 177–192. Springer (2014)
11. Budde, C.E., D’Argenio, P.R., Hartmanns, A., Sedwards, S.: A statistical model checker for nondeterminism and rare events. In: *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*. pp. 340–358. Springer (2018)
12. Cover, T.M., Thomas, J.A.: Information theory and statistics. *Elements of information theory* **1**(1), 279–335 (1991)
13. Duchon, P., Flajolet, P., Louchard, G., Schaeffer, G.: Boltzmann samplers for the random generation of combinatorial structures. *Combinatorics, Probability and Computing* **13**(4), 577–625 (2004). <https://doi.org/10.1017/S0963548304006315>, publisher: Cambridge University Press
14. Jaynes, E.T.: Information Theory and Statistical Mechanics. II. *Physical Review Online Archive (Prola)* **108**(2), 171–190 (Oct 1957)
15. Moll, V.H.: *Special Integrals of Gradshteyn and Ryzhik: the Proofs-Volume I, vol. 1*. CRC Press (2015)

16. Norman, G., Parker, D., Sproston, J.: Model checking for probabilistic timed automata. *Formal Methods in System Design* **43**(2), 164–190 (2013)

A Appendix

In this appendix, we give the omitted proofs of the results presented above after recalling their statement.

Lemma 1. *If $f = \tilde{f} \circ \theta$ then $\int_{L_n} f(\mathbf{w}) d\mathbf{w} = \int_0^{+\infty} \tilde{f}(T) V_n^L(T) dT$*

Proof.

$$\int_{L_n} f(\mathbf{w}) d\mathbf{w} = \sum_{w \in \Sigma^n} \int_0^{+\infty} \cdots \int_0^{+\infty} 1_{t_1 a_1 \cdots t_n a_n \in L_n} \tilde{f}(t_1 + \cdots + t_n) dt_1 \cdots dt_n.$$

We do the change of variables that replace t_n with $T - t_1 - \cdots - t_{n-1}$ and leave unchanged the others t_i . This change of variables has a Jacobian equals to 1 and so no multiplicative constant appears.

$$\begin{aligned} \int_{L_n} f(\mathbf{w}) d\mathbf{w} &= \sum_{w \in \Sigma^n} \int_0^{+\infty} \cdots \int_0^{+\infty} 1_{t_1 a_1 \cdots t_{n-1} a_{n-1} (T - t_1 - \cdots - t_{n-1}) a_n \in L_n} \tilde{f}(T) dt_1 \cdots dt_{n-1} dT \\ &= \int_0^{+\infty} V_n^L(T) \tilde{f}(T) dT \end{aligned}$$

□

Lemma 2. *Let p be a PDF on L_n and⁷ $\tilde{p} : T \mapsto \frac{1}{V_n(T)} \int_{L_{n,T}} p(\mathbf{w}) d\mathbf{w}$ then $H(p) \leq H(\tilde{p} \circ \theta)$ with equality if and only if $p = \tilde{p} \circ \theta$.*

Proof. We first regroup timed word with respect to duration in the definition of $H(p)$:

$$H(p) = - \int_{L_n} p(\mathbf{w}) \log_2 p(\mathbf{w}) d\mathbf{w} = - \int_0^{+\infty} \int_{L_{n,T}} p(\mathbf{w}) \log p(\mathbf{w}) d\mathbf{w} dT$$

We use Jensen's inequality with the strictly convex function $\varphi : x \mapsto x \log_2 x$ the uniform PDF on $L_{n,T} : \mathbf{w} \mapsto \frac{1}{V_n(T)}$ and the function p :

$$\begin{aligned} \int_{L_{n,T}} p(\mathbf{w}) \log_2 p(\mathbf{w}) \frac{d\mathbf{w}}{V_n(T)} &\geq \left(\int_{L_{n,T}} p(\mathbf{w}) \frac{d\mathbf{w}}{V_n(T)} \right) \log_2 \left(\int_{L_{n,T}} p(\mathbf{w}) \frac{d\mathbf{w}}{V_n(T)} \right) \\ &= \tilde{p}(T) \log_2 \tilde{p}(T) \end{aligned}$$

Multiplying by $-V_n(T)$ and integrating wrt. T we get

$$H(p) \leq - \int_0^{+\infty} V_n(T) \tilde{p}(T) \log_2 \tilde{p}(T) dT = H(\tilde{p} \circ \theta)$$

The last equality being an application of Lemma 1. To have equality $H(p) = H(\tilde{p} \circ \theta)$ we must have equality in the Jensen's inequality which is only possible if $p(\mathbf{w})$ is constant on $L_{n,T}$. This means that p depends only on duration which is equivalent to $p = \tilde{p} \circ \theta$. □

⁷ Here \tilde{p} generalises the previous concept it was used for: when p depends only on duration then $p = \tilde{p} \circ \theta$.

Theorem 2 (Maximal entropy theorem on L_n with integral constraints on duration). *Given the max-entropy problem (7) with its constraints functions f_i and constants a_i . If one find constants $\lambda_i, (0 \leq i \leq m)$ such that the function $p^*(\mathbf{w}) = \exp(\lambda_0 + \sum_{i=1}^m \lambda_i f_i(\theta(\mathbf{w})))$ is a PDF that satisfies the constraints, then the problem admits a unique solution which is p^* .*

Proof. Let p^* and λ_i as above so that p^* satisfies the constraints, then we have to prove that $H(p^*)$ is maximal and unique for this property. First, Lemma 3 implies that we only have to consider PDF q of the form $q = \tilde{q} \circ \theta$. We show that such PDF q satisfies $H(q) \leq H(p^*)$ with equality iff $q = p^*$. This is equivalent to show that $H_{V_n}(q^{\mathbb{R}}) \leq H_{V_n}(p^{\mathbb{R}})$ with equality iff $q^{\mathbb{R}} = p^{\mathbb{R}}$, with $p^{\mathbb{R}} = \tilde{p} \cdot V_n$ and $q^{\mathbb{R}} = \tilde{q} \cdot V_n$ since by Lemma 2, $H_{V_n}(p^{\mathbb{R}}) = H(p^*)$ and $H_{V_n}(q^{\mathbb{R}}) = H(q)$.

We adapt a classical proof based on Kulback-Lebler divergence that can be found in [12]. The Kullback–Leibler divergence from $p^{\mathbb{R}}$ to $q^{\mathbb{R}}$ is the quantity $D(q^{\mathbb{R}}||p^{\mathbb{R}}) = \int_0^{+\infty} q^{\mathbb{R}}(T) \log \frac{q^{\mathbb{R}}(T)}{p^{\mathbb{R}}(T)} dT$ which is always non-negative, and null iff $p^{\mathbb{R}} = q^{\mathbb{R}}$. We will show that $D(q^{\mathbb{R}}||p^{\mathbb{R}}) = H_{V_n}(p^{\mathbb{R}}) - H_{V_n}(q^{\mathbb{R}})$ and thus $H_{V_n}(p^{\mathbb{R}}) = H_{V_n}(q^{\mathbb{R}})$ will be equivalent to $p^{\mathbb{R}} = q^{\mathbb{R}}$.

We introduce $V_n(T)$ using the fact that $\log_2(q^{\mathbb{R}}/p^{\mathbb{R}}) = \log_2 \frac{q^{\mathbb{R}}(T)}{V_n(T)} - \log_2 \frac{p^{\mathbb{R}}(T)}{V_n(T)}$:

$$D(q^{\mathbb{R}}||p^{\mathbb{R}}) = \int_0^{+\infty} q^{\mathbb{R}}(T) \log_2 \frac{q^{\mathbb{R}}(T)}{V_n(T)} dT - \int_0^{+\infty} q^{\mathbb{R}}(T) \log_2 \frac{p^{\mathbb{R}}(T)}{V_n(T)} dT$$

The first integral is $-H_{V_n}(q^{\mathbb{R}})$, hence it remains to prove that $H_{V_n}(p^{\mathbb{R}}) = -\int q^{\mathbb{R}}(T) \log_2 \left(\frac{p^{\mathbb{R}}(T)}{V_n(T)} \right)$. We recall that $\frac{p^{\mathbb{R}}(T)}{V_n(T)} = \tilde{p}(T) = \exp(\sum_{i=0}^m \lambda_i f_i(T))$, thus

$$\begin{aligned} \int_0^{+\infty} q^{\mathbb{R}}(T) \log_2 \left(\frac{p^{\mathbb{R}}(T)}{V_n(T)} \right) &= \frac{1}{\ln 2} \int_0^{+\infty} q^{\mathbb{R}}(T) \left(\sum_{i=0}^m \lambda_i f_i(T) \right) dT \\ &= \frac{1}{\ln 2} \sum_{i=0}^m \lambda_i \int_0^{+\infty} q^{\mathbb{R}}(T) f_i(T) dT. \end{aligned}$$

Since $q^{\mathbb{R}}$ satisfies the constraints this quantity is equal to $\frac{1}{\ln 2} \sum_{i=0}^m \lambda_i a_i$. The same reasoning holds for p , so we can conclude the proof with the equality:

$$\begin{aligned} H_{V_n}(p^{\mathbb{R}}) &= -\int_0^{+\infty} p^{\mathbb{R}}(T) \log_2 \left(\frac{p^{\mathbb{R}}(T)}{V_n(T)} \right) dT = -\frac{1}{\ln(2)} \sum_{i=0}^m \lambda_i a_i \\ &= -\int_0^{+\infty} q^{\mathbb{R}}(T) \log_2 \left(\frac{p^{\mathbb{R}}(T)}{V_n(T)} \right) dT \end{aligned}$$

□

Proposition 2 (Recursive definition of v_n).

$$v_{n,s}^L = \sum_{a \in \Sigma} \left(\int_0^{+\infty} e^{-st} v_{n-1,s}^{(ta)^{-1}L} dt \right) \quad (15)$$

Proof.

$$v_{n,s}^L = \sum_{a \in \Sigma} \cdots \sum_{a_n \in \Sigma} \int_0^{+\infty} 1_{ta \cdots t_n a_n \in L_n} e^{-s(t+t_2 \cdots + t_n)} dt dt_2 \cdots dt_n =$$

$$\sum_{a \in \Sigma} \left(\int_0^{+\infty} \left(\sum_{a_2 \in \Sigma} \cdots \sum_{a_n \in \Sigma} \int_0^{+\infty} \cdots \int_0^{+\infty} 1_{ta \cdots t_n a_n \in L_n} e^{-s(t_2 \cdots + t_n)} dt_2 \cdots dt_n \right) e^{-st} dt \right)$$

Then it suffices to remark that $ta \cdots t_n a_n \in L_n$ iff $t_2 a_2 \cdots t_n a_n \in ((ta)^{-1}L)_{n-1}$ \square

Theorem 3 (Maximal entropy theorem for measures on L_n with prescribed mean duration). *Given a timed language L , length n , duration⁸ T_{mean} , among the probability measures p that satisfy the constraint $E_p(\theta(w)) = T_{mean}$, there is a unique one p^* that maximizes the entropy. It is given by*

$$p^*(w) = \frac{e^{-s\theta(w)}}{v_{n,s}^L}$$

where the normalising constant $v_{n,s}^L$ is defined by

$$v_{n,s}^L = \int_{L_n} e^{-s\theta(\mathbf{w})} d\mathbf{w} = \int_0^{+\infty} e^{-sT} V_n^L(T) dT \quad (16)$$

with s the unique real such that $\frac{1}{v_{n,s}^L} \frac{\partial v_{n,s}^L}{\partial s} = T_{mean}$.

Proof. Applying the theorem tells us that $p^*(w) = e^{\lambda_0 + \lambda_1 \theta(w)}$ with $\int_{L_n} e^{\lambda_0 + \lambda_1 \theta(w)} dt = 1$ and $\int_{L_n} \theta(w) e^{\lambda_0 + \lambda_1 \theta(w)} dt = T_{mean}$. We let $s = -\lambda_1$ and let $v_{n,s}^L = e^{-\lambda_0} = \int_{L_n} e^{-s\theta(w)} dt$ which by Lemma 1 is also equal to $\int_0^{+\infty} e^{-sT} V_n^L(T) dT$.

The expected duration is by definition:

$$\int_{L_n} \theta(\mathbf{w}) \frac{e^{-s\theta(\mathbf{w})}}{v_{n,s}^L} d\mathbf{w}.$$

We can express this with the derivatives of $v_{n,s}^L = \int_{L_n} e^{-s\theta(\mathbf{w})} d\mathbf{w}$. Indeed,

$$\frac{\partial v_{n,s}^L}{\partial s} = \int_{L_n} \frac{\partial e^{-s\theta(\mathbf{w})}}{\partial s} d\mathbf{w} = - \int_{L_n} \theta(\mathbf{w}) e^{-s\theta(\mathbf{w})} d\mathbf{w} = -v_{n,s}^L T_{mean}.$$

The first equality holds because the integrated function $v_{n,s}^L$ is non-negative. \square

The following Lemma is useful in the proof of proposition 3.

⁸ satisfying the constraint to be in the bounds for L_n

Lemma 3 (Folk, see [15] page 82). *The primitive function of $e^{ax}x^n$ wrt. x is*

$$n!e^{ax} \sum_{k=0}^n \frac{(-1)^k x^{n-k}}{(n-k)!a^{k+1}}$$

Proposition 3. *For every location q and length n , $\mathbf{x} \mapsto v_{n,s}(q, \mathbf{x})$ is an exponential polynomial. It can be computed in polynomial time using dynamic programming from the following recursive equations:*

$$v_{0,s}(l, \mathbf{x}) = 1_{l \in F} \text{ and } v_{m,s}(l, \mathbf{x}) = \sum_{\delta=l} \int_{\mathbf{1b}_\delta(\mathbf{x})}^{\mathbf{ub}_\delta(\mathbf{x})} v_{m-1,s}(\delta^+, \mathbf{r}_\delta(\mathbf{x}+t)) e^{-st} dt. \quad (17)$$

Proof. (12) is a consequence of (11) where we use the language starting from states and their corresponding LTV. We reason by induction and assume that the results holds at some rank $m-1$ for some m (and every location q): and show it at rank m . The base case is straightforward since the function is $v_{0,s}(l, \mathbf{x})$ is the constant 1 or 0 depending on whether the location is final or not. We use the recursive characterisation (12). A sum of exponential polynomial is clearly an exponential polynomial so it suffices to show that every integral in (12) gives an exponential polynomial. There are two cases to compute the integral:

1. $x_i \in \mathbf{r}_\delta$:

$$\begin{aligned} \int_{\mathbf{1b}_\delta(\mathbf{x})}^{\mathbf{ub}_\delta(\mathbf{x})} P_{(i,k)} \left(\frac{1}{s}, \mathbf{r}_\delta(\mathbf{x}+t) \right) e^{s(\mathbf{r}_\delta(\mathbf{x}+t)_i - k)} e^{-st} dt \\ = e^{-sk} \int_{\mathbf{1b}_\delta(\mathbf{x})}^{\mathbf{ub}_\delta(\mathbf{x})} P_{(i,k)} \left(\frac{1}{s}, \mathbf{r}_\delta(\mathbf{x}+t) \right) e^{-st} dt \end{aligned}$$

as $t \mapsto P_{(i,k)} \left(\frac{1}{s}, \mathbf{r}_\delta(\mathbf{x}+t) \right)$ is a polynomial in t , applying classical results on Laplace Transform of polynomials (See for example [15] page 82, recalled in the appendix) we get:

$$e^{-sk} (e^{-s \mathbf{ub}_\delta(\mathbf{x})} \tilde{P}(1/s, \mathbf{x}) - e^{-s \mathbf{1b}_\delta(\mathbf{x})} \tilde{P}(1/s, \mathbf{x}))$$

where \tilde{P} is a polynomial. We recall that the bounds $\mathbf{1b}_\delta(\mathbf{x})$ and $\mathbf{ub}_\delta(\mathbf{x})$ are both of the form $c - x_i$ for some constant c and clock x_i , so the whole expression is an exponential polynomial.

2. $x_i \notin \mathbf{r}_\delta$:

$$\begin{aligned} \int_{\mathbf{1b}_\delta(\mathbf{x})}^{\mathbf{ub}_\delta(\mathbf{x})} P_{(i,k)} \left(\frac{1}{s}, \mathbf{r}_\delta(\mathbf{x}+t) \right) e^{s(\mathbf{r}_\delta(\mathbf{x}+t)_i - k)} e^{-st} dt \\ = e^{s(x_i - k)} \int_{\mathbf{1b}_\delta(\mathbf{x})}^{\mathbf{ub}_\delta(\mathbf{x})} P_{(i,k)}(\mathbf{r}_\delta(\mathbf{x}+t)) dt \end{aligned}$$

as $e^{s(x_i - k)}$ is a constant with respect to t . We assume that $P_{(i,k)}$ is non-null otherwise (i, k) can be removed from S . It cannot be the case that

$\mathbf{ub}_\delta(\mathbf{x}) = +\infty$ otherwise the integral would give an infinite value which is not possible because the LTV corresponding to the language starting from (q, \mathbf{x}) is well defined (at least for positive s) and satisfies (12). Hence \mathbf{ub}_δ is like \mathbf{lb}_δ a polynomial (of degree 1), and hence the integral is a polynomial as integral with polynomial bound of a polynomial.

□