A Logical Characterization of Timed Pushdown Languages

Manfred Droste[⋆] and Vitaly Perevoshchikov[⋆⋆]

Universität Leipzig, Institut für Informatik,
04109 Leipzig, Germany
{droste,perev}@informatik.uni-leipzig.de

Abstract. Timed pushdown automata were introduced by Abdulla et al. in LICS 2012 to model the behavior of real-time recursive systems. In this paper, we introduce a quantitative logic on timed words which is expressively equivalent to timed pushdown automata. This logic is an extension of Wilke's relative distance logic by quantitative matchings. To show the expressive equivalence result, we prove a decomposition theorem which establishes a connection between timed pushdown languages and visibly pushdown languages of Alur and Mudhusudan; then we apply their result about the logical characterization of visibly pushdown languages. As a consequence, we obtain the decidability of the satisfiability problem for our new logic.

Keywords: Timed pushdown automata, visibly pushdown languages, timed languages, relative distance logic, matchings

1 Introduction

Timed automata introduced by Alur and Dill [3] are a prominent model for the specification and analysis of real-time systems. Recently, Abdulla, Atig and Stenman [1] proposed *timed pushdown automata* (TPDA) for the verification of real-time recursive systems. In the model of TPDA, classical pushdown automata are augmented with real-valued clocks; moreover, the stack of this automaton keeps track of the age of its elements.

Since the seminal Büchi-Elgot theorem [6, 9] establishing the expressive equivalence of nondeterministic automata and monadic second-order logic, a significant field of research investigates logical descriptions of language classes appearing from practically relevant automata models. The goal of this paper is to provide a logical characterization for timed pushdown automata.

For our purpose, we introduce a *timed matching logic*. As in the logic of Lautemann, Schwentick and Thérien [10], we handle the stack functionality by means of a binary *matching* predicate. As in the logic of Wilke [15], we use

[⋆] Part of this work was done while the author was visiting Immanuel Kant Baltic Federal University, Kaliningrad, Russia

[⋆⋆] Supported by DFG Graduiertenkolleg 1763 (QuantLA)

relative distance predicates to handle the functionality of clocks. Moreover, to handle the ages of stack elements, we lift the binary matchings to the timed setting, i.e., we can compare the time distance between matched positions with a constant. The main result of this paper is the expressive equivalence of TPDA and timed matching logic.

Here, we face the following difficulties in the proof of our main result. The class of timed pushdown languages is most likely not closed under intersection and complement (as the class of context-free languages). Moreover, we cannot directly follow the approaches of [10] and [15], since the proof of [10] appeals to the logical characterization result for trees [14] (but, there is no suitable logical characterization for regular timed tree languages) and the proof of [15] appeals to the classical Büchi-Elgot result [6, 9] (and, this way does not permit to handle matchings). In our case, we appeal to the MSO-like characterization of *visibly pushdown languages* of Alur and Madhusudan [4].

We show our expressive equivalence result as follows.

- We prove a Nivat-like decomposition theorem for TPDA (cf. [12, 5]) which may be of independent interest; this theorem establishes a connection between timed pushdown languages and untimed visibly pushdown languages of [4] by means of operations like renamings and intersections with simple timed pushdown languages. So we can separate the continuous timed part of the model of TPDA from its discrete part. The main difficulty here is to encode the infinite time domain, namely $\mathbb{R}_{\geq 0}$, as a finite alphabet. We will show that it suffices to use several partitions of $\mathbb{R}_{\geq 0}$ into intervals to construct the desired extended alphabet. On the one hand, we interpret these intervals as components of the extended alphabet. On the other hand, we use them to control the timed part of the model.
- In a similar way, we separate the quantitative timed part of timed matching logic from the qualitative part described by MSO logic with matchings over a visibly pushdown alphabet [4].
- Then we can deduce our result from the result of [4].

Since our proof is constructive and the reachability for TPDA is decidable [1], we can also decide the satisfiability for our timed matching logic.

2 Timed Pushdown Automata

An *alphabet* is a non-empty finite set. Let Σ be a non-empty set (possibly infinite). A *finite word* over Σ is a finite sequence $a_1...a_n$ where $n \geq 0$ and $a_1, ..., a_n \in \Sigma$. If $n = 0$, then we say that w is *empty* and denote it by ε. Otherwise, we call w *non-empty*. Let Σ^* denote the set of all words over Σ and let Σ^+ denote the set of all non-empty words over Σ. We denote by $\mathbb{N} = \{0, 1, ...\}$ and $\mathbb{R}_{\geq 0}$ the sets of natural and non-negative real numbers, respectively. A *finite timed word* over Σ is a finite word over the infinite set $\Sigma \times \mathbb{R}_{\geq 0}$. Let $\mathbb{T}\Sigma^* = (\Sigma \times \mathbb{R}_{\geq 0})^*$, the set of all finite timed words over Σ, and $\mathbb{T}\Sigma^+ = (\Sigma \times \mathbb{R}_{\geq 0})^+$, the set of all non-empty finite timed words over

Σ. Any set $\mathcal{L} \subseteq \mathbb{T}\Sigma^+$ of finite timed words is called a *timed language*. For $w = (a_1, t_1)...(a_n, t_n) \in \mathbb{T}\Sigma^+$, let $|w| = n$, the *length* of w, and $\langle w \rangle = t_1 + ... + t_n$, the *time length* of w. For $0 \leq i < j \leq n$, let $\langle w \rangle_{i,j} = t_{i+1} + ... + t_j$.

Let $\mathcal{I}$ denote the set of all intervals of the form $[a, b]$, $(a, b]$, $[a, b)$, (a, b), $[a, \infty)$ or (a, ∞) where $a, b \in \mathbb{N}$. Let C be a finite set of *clock variables* ranging over $\mathbb{R}_{\geq 0}$. A *clock constraint* over C is a mapping $\phi : C \to \mathcal{I}$ which assigns an interval to each clock variable. Let $\mathcal{I}^C$ be the set of all clock constraints over C. A *clock valuation* over C is a mapping $\nu : C \to \mathbb{R}_{\geq 0}$ which assigns a value to each clock variable. Let $\mathbb{R}^C_{\geq 0}$ denote the set of all clock valuations over C. For $\nu \in \mathbb{R}^C_{\geq 0}$ and $\phi \in \mathcal{I}^C$, we write $\nu \models \phi$ if $\nu(c) \in \phi(c)$ for all $c \in C$. For $t \in \mathbb{R}_{\geq 0}$, let $\nu^{+t} : C \to \mathbb{R}_{\geq 0}$ be defined for all $c \in C$ by $\nu^{+t}(c) = \nu(c) + t$. For $\Lambda \subseteq C$, let $\nu^{\Lambda \leftarrow 0} : C \to \mathbb{R}_{\geq 0}$ be defined by $\nu^{\Lambda \leftarrow 0}(c) = 0$ for all $c \in \Lambda$ and $\nu^{\Lambda \leftarrow 0}(c) = \nu(c)$ for all $c \in C \setminus \Lambda$. If Γ is an alphabet, $u = (g_1, t_1)...(g_n, t_n) \in \mathbb{T}\Gamma^*$ and $t \in \mathbb{R}_{\geq 0}$, let $u^{+t} = (g_1, t_1 + t)...(g_n, t_n + t) \in \mathbb{T}\Gamma^*$.

Now we turn to the definition of *timed pushdown automata* which have been introduced and investigated in [1]. These machines are nondeterministic automata equipped with clocks (like timed automata) and a stack (like pushdown automata). In contrast to untimed pushdown automata, in the model of timed pushdown automata we push together with a letter a clock which will measure the age of this letter in the stack. Then, we can pop this letter only if its age satisfies a given constraint. We slightly extend the definition of timed pushdown automata presented in [1] by allowing labels of edges. This, however, does not harm the decidability of the reachability problem which was shown in [1].

Let Γ be a *stack alphabet*. We denote by $\mathcal{S}(\Gamma) = (\{\downarrow\} \times \Gamma) \cup \{\#\} \cup (\{\uparrow\} \times \Gamma \times \mathcal{I})$ the set of *stack commands* over Γ.

Definition 2.1. *Let Σ be an alphabet. A* timed pushdown automaton (TPDA) *over Σ is a tuple $\mathcal{A} = (L, \Gamma, C, L_0, E, L_f)$ where L is a finite set of* locations*, Γ is a finite* stack alphabet*, C is a finite set of* clocks*, $L_0, L_f \subseteq L$ are sets of* initial *resp.* final *locations, and $E \subseteq L \times \Sigma \times \mathcal{S}(\Gamma) \times \mathcal{I}^C \times 2^C \times L$ is a finite set of* edges*.*

Let $e = (\ell, a, s, \phi, \Lambda, \ell') \in E$ be an edge of $\mathcal{A}$ with $\ell, \ell' \in L$, $a \in \Sigma$, $s \in \mathcal{S}(\Gamma)$, $\phi \in \mathcal{I}^C$ and $\Lambda \subseteq C$. We say that a is the *label* of e and denote it by label(e). We also let stack$(e) = s$, the stack command of e. Let $E^{\downarrow} \subseteq E$ denote the set of all *push* edges e with stack$(e) = (\downarrow, \gamma)$ for some $\gamma \in \Gamma$. Similarly, let $E^{\#} = \{e \in E \mid \text{stack}(e) = \#\}$ be the set of *local* edges and $E^{\uparrow} = \{e \in E \mid \text{stack}(e) = (\uparrow, \gamma, I)$ for some $\gamma \in \Gamma$ and $I \in \mathcal{I}\}$ the set of *pop* edges. Then, we have $E = E^{\downarrow} \cup E^{\#} \cup E^{\uparrow}$.

A *configuration* c of $\mathcal{A}$ is described by the present location, the values of the clocks, and the stack, which is a timed word over Γ. That is, c is a triple $\langle \ell, \nu, u \rangle$ where $\ell \in L$, $\nu \in \mathbb{R}^C_{\geq 0}$ and $u \in \mathbb{T}\Gamma^*$. We say that c is *initial* if $\ell \in L_0$, $\nu(x) = 0$ for all $x \in C$ and $u = \varepsilon$. We say that c is *final* if $\ell \in L_f$ and $u = \varepsilon$. Let $\mathcal{C}_{\mathcal{A}}$ denote the set of all configurations of $\mathcal{A}$, $\mathcal{C}^0_{\mathcal{A}}$ the set of all initial configurations of $\mathcal{A}$ and $\mathcal{C}^f_{\mathcal{A}} \subseteq \mathcal{C}_{\mathcal{A}}$ the set of all final configurations.

Let $c = \langle \ell, \nu, u \rangle$ and $c' = \langle \ell', \nu', u' \rangle$ be two configurations with $u = (\gamma_1, t_1)(\gamma_2, t_2)...(\gamma_k, t_k)$ and let $e = (q, a, s, \phi, \Lambda, q') \in E$ be an edge. We say that $c \vdash_e c'$ is a *switch transition* if $\ell = q$, $\ell' = q'$, $\nu \models \phi$, $\nu' = \nu^{\Lambda \leftarrow 0}$, and:

- if $s = (\downarrow, \gamma)$ for some $\gamma \in \Gamma$, then $u' = (\gamma, 0)u$;
- if $s = \#$, then $u' = u$;
- if $s = (\uparrow, \gamma, I)$ with $\gamma \in \Gamma$ and $I \in \mathcal{I}$, then $k \geq 1$, $\gamma = \gamma_1$, $t_1 \in I$ and $u' = (\gamma_2, t_2)...(\gamma_k, t_k)$.

For $t \in \mathbb{R}_{\geq 0}$, we say that $c \vdash_t c'$ is a *delay transition* if $\ell = \ell'$, $\nu' = \nu^{+t}$ and $u' = u^{+t}$. For $t \in \mathbb{R}_{\geq 0}$ and $e \in E$, we write $c \vdash_{t,e} c'$ if there exists $c'' \in \mathcal{C}_{\mathcal{A}}$ with $c \vdash_t c''$ and $c'' \vdash_e c'$.

A *run* ρ of $\mathcal{A}$ is an alternating sequence of delay and switch transitions which starts in an initial configuration and ends in a final configuration, formally, $\rho = c_0 \vdash_{t_1,e_1} c_1 \vdash_{t_2,e_2} ... \vdash_{t_n,e_n} c_n$ where $n \geq 1$, $c_0 \in \mathcal{C}^0_{\mathcal{A}}$, $c_1, ..., c_{n-1} \in \mathcal{C}_{\mathcal{A}}$, $c_n \in \mathcal{C}^f_{\mathcal{A}}$, $t_1, ..., t_n \in \mathbb{R}_{\geq 0}$ and $e_1, ..., e_n \in E$. The *label* of ρ is the timed word $\text{label}(\rho) = (\text{label}(e_1), t_1)...(\text{label}(e_n), t_n) \in \mathbb{T}\Sigma^+$. Let $\mathcal{L}(\mathcal{A}) = \{w \in \mathbb{T}\Sigma^+ \mid \text{there exists a run } \rho \text{ of } \mathcal{A} \text{ with } \text{label}(\rho) = w\}$, the timed language *recognized* by $\mathcal{A}$. We say that a timed language $\mathcal{L} \subseteq \mathbb{T}\Sigma^+$ is a *timed pushdown language* if there exists a TPDA $\mathcal{A}$ over Σ such that $\mathcal{L}(\mathcal{A}) = \mathcal{L}$.

3 Timed Matching Logic

The goal of this section is to develop a logical formalism which is expressively equivalent to TPDA defined in Sect. 2. Our new logic will incorporate Wilke's *relative distance logic* [15] for timed automata as well as the logic with *matchings* [10] introduced by Lautemann, Schwentick and Thérien for context-free languages. Moreover, we augment our logic with the possibility to measure the time distance between matched positions.

For the rest of this paper, we fix countable and pairwise disjoint sets $\mathcal{V}_1$, the set of *first-order variables*, $\mathcal{V}_2$, the set of *second-order variables*, and $\mathcal{D}$, the set of (second-order) *relative distance variables*, as well as a *matching variable* $\mu \notin \mathcal{V}_1 \cup \mathcal{V}_2 \cup \mathcal{D}$. Let $\mathcal{U} = \mathcal{V}_1 \cup \mathcal{V}_2 \cup \mathcal{D} \cup \{\mu\}$ and $\mathbb{K} = \{<, \leq, =, \geq, >\} \times \mathbb{N}$.

Let Σ be an alphabet. The set $\text{TMSO}(\Sigma)$ of *timed matching MSO formulas* is defined by the grammar

$$\varphi ::= P_a(x) \mid x \leq y \mid \mathcal{X}(x) \mid \mathrm{d}^\kappa(D, x) \mid \mu^\kappa(x, y) \mid \varphi \vee \varphi \mid \neg\varphi \mid \exists x.\varphi \mid \exists X.\varphi$$

where $a \in \Sigma$, $x, y \in \mathcal{V}_1$, $X \in \mathcal{V}_2$, $D \in \mathcal{D}$, $\mathcal{X} \in \mathcal{V}_2 \cup \mathcal{D}$ and $\kappa \in \mathbb{K}$. The formulas of the form $\mathrm{d}^\kappa(D, x)$ are called *relative distance predicates* and the formulas of the form $\mu^\kappa(x, y)$ are called *distance matchings*. For $\mu^{\geq 0}(x, y)$, we will write simply $\mu(x, y)$. For $a, b \in \mathbb{N}$, let $\mu^{[a,b)}(x, y) = \mu^{\geq a}(x, y) \wedge \mu^{<b}(x, y)$. In the same manner we define $\mu^I(x, y)$ and $\mathrm{d}^I(D, x)$ for all intervals $I \in \mathcal{I}$.

The $\text{TMSO}(\Sigma)$-formulas are interpreted over timed words over Σ and assignments of variables. Let $w \in \mathbb{T}\Sigma^+$ be a timed word and $\text{dom}(w) = \{1, ..., |w|\}$, the *domain* of w. A *w-assignment* is a mapping $\sigma : \mathcal{U} \to \text{dom}(w) \cup 2^{\text{dom}(w)} \cup 2^{(\text{dom}(w))^2}$ such that $\sigma(\mathcal{V}_1) \subseteq \text{dom}(w)$,

$(w,\sigma) \models P_a(x)$	iff	$a_{\sigma(x)} = a$
$(w,\sigma) \models x \leq y$	iff	$\sigma(x) \leq \sigma(y)$
$(w,\sigma) \models \mathcal{X}(x)$	iff	$\sigma(x) \in \sigma(\mathcal{X})$
$(w,\sigma) \models \mathrm{d}^\kappa(D,x)$	iff	$(\sigma(D),\sigma(x)) \in \mathrm{d}^\kappa(w)$
$(w,\sigma) \models \mu^\kappa(x,y)$	iff	$(\sigma(x),\sigma(y),\sigma(\mu)) \in \mu^\kappa(w)$
$(w,\sigma) \models \varphi_1 \vee \varphi_2$	iff	$(w,\sigma) \models \varphi_1$ or $(w,\sigma) \models \varphi_2$
$(w,\sigma) \models \neg\varphi$	iff	$(w,\sigma) \models \varphi$ does not hold
$(w,\sigma) \models \exists x.\varphi$	iff	$\exists i \in \mathrm{dom}(w) : (w,\sigma[x/i]) \models \varphi$
$(w,\sigma) \models \exists X.\varphi$	iff	$\exists I \subseteq \mathrm{dom}(w) : (w,\sigma[X/I]) \models \varphi$

Table 1. The semantics of $\mathrm{TMSO}(\Sigma)$-formulas

$\sigma(\mathcal{V}_2 \cup \mathcal{D}) \subseteq 2^{\mathrm{dom}(w)}$ and $\sigma(\mu) \subseteq 2^{(\mathrm{dom}(w))^2}$. Let σ be a w-assignment. For $x \in \mathcal{V}_1$ and $i \in \mathrm{dom}(w)$, the *update* $\sigma[x/i]$ is the w-assignment defined by $\sigma[x/i](x) = i$ and $\sigma[x/i](y) = \sigma(y)$ for all $y \in \mathcal{U} \setminus \{x\}$. Similarly, for $\mathcal{X} \in \mathcal{V}_2 \cup \mathcal{D}$ and $I \subseteq \mathrm{dom}(w)$, we define the update $\sigma[\mathcal{X}/I]$ and, for $M \subseteq (\mathrm{dom}(w))^2$, the update $\sigma[\mu/M]$.

Let $w \in \mathbb{T}\Sigma^+$ be a timed word and $\kappa = (\bowtie, k) \in \mathbb{K}$. For $j \in \mathrm{dom}(w)$ and $J \subseteq \mathrm{dom}(w)$, we will write $(J,j) \in \mathrm{d}^\kappa(w)$ if $\langle w\rangle_{i,j} \bowtie k$ for the greatest value $i \in J \cup \{0\}$ with $i < j$. For $i,j \in \mathrm{dom}(w)$, $M \subseteq (\mathrm{dom}(w))^2$ and $\kappa = (\bowtie, k) \in \mathbb{K}$, we will write $(i,j,M) \in \mu^\kappa(w)$ if $i < j$, $(i,j) \in M$ and $\langle w\rangle_{i,j} \bowtie k$.

Given a formula $\varphi \in \mathrm{TMSO}(\Sigma)$, a timed word $w = (a_1,t_1)...(a_n,t_n) \in \mathbb{T}\Sigma^+$ and a w-assignment σ; the satisfaction relation $(w,\sigma) \models \varphi$ is defined inductively on the structure of φ as shown in Table 1. Here, $a \in \Sigma$, $x,y \in \mathcal{V}_1$, $X \in \mathcal{V}_2$, $D \in \mathcal{D}$, $\mathcal{X} \in \mathcal{V}_2 \cup \mathcal{D}$ and $\kappa \in \mathbb{K}$.

For $\varphi \in \mathrm{TMSO}(\Sigma)$ and $y \in \mathcal{V}_1$, let $\exists^{\leq 1} y.\varphi$ denote the formula $\neg\exists y.\varphi \vee \exists y.(\varphi \wedge \forall z.(z \neq y \rightarrow \neg\varphi[y/z]))$ where $z \in \mathcal{V}_1$ does not occur in φ and $\varphi[y/z]$ is the formula obtained from φ by replacing y by z. Let $\textsc{Matching}(\mu) \in \mathrm{TMSO}(\Sigma)$ denote the formula

$$\begin{aligned}\textsc{Matching}(\mu) =&\forall x.\forall y.(\mu(x,y) \rightarrow x < y) \wedge \forall x.\exists^{\leq 1} y.(\mu(x,y) \vee \mu(y,x)) \wedge \\ &\forall x.\forall y.\forall u.\forall v.((\mu(x,y) \wedge \mu(u,v) \wedge x < u < y) \rightarrow x < v < y).\end{aligned}$$

This formula demands that a binary relation μ on a timed word domain is a *matching* (cf. [10]), i.e., it is compatible with $<$, each element of the domain belongs to at most one pair in μ and μ is noncrossing.

The set $\mathrm{TML}(\Sigma)$ of the formulas of *timed matching logic* over Σ is defined to be the set of all formulas of the form

$$\psi = \exists\mu.\exists D_1.\ ...\ \exists D_m.(\varphi \wedge \textsc{Matching}(\mu))$$

where $r \geq 0$, $D_1,...,D_m \in \mathcal{D}$ and $\varphi \in \mathrm{TMSO}(\Sigma)$. Let $w \in \mathbb{T}\Sigma^+$ and σ be a w-assignment. Then, $(w,\sigma) \models \psi$ iff there exist $I_1,...,I_m \subseteq \mathrm{dom}(w)$ and a matching $M \subseteq (\mathrm{dom}(w))^2$ such that $(w,\sigma[D_1/I_1,...,D_m/I_m,\mu/M]) \models \varphi$. For simplicity, we will denote ψ by $\exists^{\mathrm{match}}\mu.\exists D_1.\ ...\ \exists D_m.\varphi$.

For a formula $\psi \in \text{TML}(\Sigma)$, the set $\text{Free}(\psi) \subseteq \mathcal{U}$ of *free variables* of ψ is defined as usual. We say that $\psi \in \text{TML}(\Sigma)$ is a *sentence* if $\text{Free}(\psi) = \emptyset$. Note that, for a sentence ψ, the satisfaction relation $(w, \sigma) \models \psi$ does not depend on a w-assignment σ. Then, we will simply write $w \models \psi$. Let $\mathcal{L}(\psi) = \{w \in \mathbb{T}\Sigma^+ \mid w \models \psi\}$, the language *defined* by ψ. We say that a timed language $\mathcal{L} \subseteq \mathbb{T}\Sigma^+$ is TML-*definable* if there exists a sentence $\psi \in \text{TML}(\Sigma)$ such that $\mathcal{L}(\psi) = \mathcal{L}$.

Example 3.1. Here, we consider a timed extension of the well-known *Dyck languages*. Let $\Sigma = \{a_1, ..., a_m\}$ be a set of opening brackets and $\overline{\Sigma} = \{\overline{a}_1, ..., \overline{a}_m\}$ a set of corresponding closing brackets. Let $\Sigma' = \Sigma \cup \overline{\Sigma}$ be our alphabet. Let $I_1, ..., I_m \in \mathcal{I}$ be intervals. We will consider the *timed Dyck language* $\mathcal{D}_\Sigma(I_1, ..., I_m) \subseteq \mathbb{T}(\Sigma')^+$ of timed words $w = (a_1, t_1)...(a_n, t_n)$ where $a_1...a_n$ is a sequence of correctly nested brackets and, for every $j \in \{1, ..., m\}$, the time distance between any two matching brackets a_j and $\overline{a}_j$ is in I_j. The timed language $\mathcal{D}_\Sigma(I_1, ..., I_m)$ is defined by the $\text{TML}(\Sigma)$-sentence

$$\exists^{\text{match}}\mu. \Big(\forall x. \exists y. (\mu(x,y) \vee \mu(y,x)) \wedge \\ \forall x. \forall y. \Big(\mu(x,y) \to \bigvee\nolimits_{j=1}^{m} (P_{a_j}(x) \wedge P_{\overline{a}_j}(y) \wedge \mu^{I_j}(x,y)) \Big) \Big).$$

Our main result is the following theorem.

Theorem 3.2. *Let Σ be an alphabet and $\mathcal{L} \subseteq \mathbb{T}\Sigma^+$ a timed language. Then $\mathcal{L}$ is a timed pushdown language iff $\mathcal{L}$ is* TML*-definable.*

Note that Theorem 3.2 extends the result of [10] for context-free languages as well as the result of [15] for regular timed languages. As already mentioned in the introduction, we will use the logical characterization result for visibly pushdown languages [4]. In Sect. 4, for the convenience of the reader, we recall this result. In Sect. 5, we show a Nivat-like decomposition theorem for timed pushdown languages. Finally, in Sect. 6, we give a proof of Theorem 3.2.

It was shown in [1] that the emptiness problem for TPDA is decidable. Moreover, as we will see later, our proof of Theorem 3.2 is constructive. Then, we obtain the decidability of the satisfiability problem for our timed matching logic.

Corollary 3.3. *It is decidable, given an alphabet Σ and a sentence $\psi \in \text{TML}(\Sigma)$, whether there exists a timed word $w \in \mathbb{T}\Sigma^+$ such that $w \models \psi$.*

4 Visibly Pushdown Languages

For the rest of the paper, we fix a special stack symbol $\perp$.

A *pushdown alphabet* is a triple $\tilde{\Sigma} = \langle \Sigma^\downarrow, \Sigma^\#, \Sigma^\uparrow \rangle$ with pairwise disjoint sets $\Sigma^\downarrow$, $\Sigma^\#$ and $\Sigma^\uparrow$ of *push*, *local* and *pop* letters, respectively. Let $\Sigma = \Sigma^\downarrow \cup \Sigma^\# \cup \Sigma^\uparrow$. A *visibly pushdown automaton (VPA)* over $\tilde{\Sigma}$ is a tuple $\mathcal{A} = (Q, \Gamma, Q_0, T, Q_f)$

where Q is a finite set of states, $Q_0, Q_f \subseteq Q$ are sets of initial resp. final states, Γ is a stack alphabet with $\perp \notin \Gamma$, and $T = T^{\downarrow} \cup T^{\#} \cup T^{\downarrow}$ is a set of transitions where $T^{\downarrow} \subseteq Q \times \Sigma^{\downarrow} \times \Gamma \times Q$ is a set of push transitions, $T^{\#} \subseteq Q \times \Sigma^{\#} \times Q$ is a set of local transitions and $T^{\uparrow} \subseteq Q \times \Sigma^{\uparrow} \times (\Gamma \cup \{\perp\}) \times Q$ is a set of pop transitions.

We define the label of a transition $\tau \in T$ depending on its sort as follows. If $\tau = (p, c, \gamma, p') \in T^{\downarrow} \cup T^{\uparrow}$ or $\tau = (p, c, p') \in T^{\#}$, we let $\mathrm{label}(\tau) = c$, so $c \in \Sigma^{\downarrow} \cup \Sigma^{\uparrow}$ resp. $c \in \Sigma^{\#}$.

A *configuration* of $\mathcal{A}$ is a pair $\langle q, u\rangle$ where $q \in Q$ and $u \in \Gamma^*$. Let $\tau \in T$ be a transition. Then, we define the transition relation $\vdash_\tau$ on configurations of $\mathcal{A}$ as follows. Let $c = \langle q, u\rangle$ and $c' = \langle q', u'\rangle$ be configurations of $\mathcal{A}$.

- If $\tau = (p, a, \gamma, p') \in T^{\downarrow}$, then we put $c \vdash_\tau c'$ iff $p = q$, $p' = q'$ and $u' = \gamma u$.
- If $\tau = (p, a, p') \in T^{\#}$, then we put $c \vdash_\tau c'$ iff $p = q$, $p' = q'$ and $u' = u$,
- If $\tau = (p, a, \gamma, p') \in T^{\uparrow}$ with $\gamma \in \Gamma \cup \{\perp\}$, then we put $c \vdash_\tau c'$ iff $p = q$, $p' = q'$ and either $\gamma \neq \perp$ and $u = \gamma u'$, or $\gamma = \perp$ and $u' = u = \varepsilon$.

We say that $c = \langle q, u\rangle$ is an *initial* configuration if $q \in Q_0$ and $u = \varepsilon$. We call c a *final* configuration if $q \in Q_f$. A *run* of $\mathcal{A}$ is a sequence $\rho = c_0 \vdash_{\tau_1} c_1 \vdash_{\tau_2} \ldots \vdash_{\tau_n} c_n$ where $c_0, c_1, \ldots, c_n$ are configurations of $\mathcal{A}$ such that c_0 is initial, c_n is final and $\tau_1, \ldots, \tau_n \in T$. Let $\mathrm{label}(\rho) = \mathrm{label}(\tau_1) \ldots \mathrm{label}(\tau_n) \in \Sigma^+$, the *label* of ρ. Let $\mathcal{L}(\mathcal{A}) = \{w \in \Sigma^+ \mid \text{there exists a run } \rho \text{ of } \mathcal{A} \text{ with } \mathrm{label}(\rho) = w\}$. We say that a language $\mathcal{L} \subseteq \Sigma^+$ is a *visibly pushdown language* over $\tilde{\Sigma}$ if there exists a VPA $\mathcal{A}$ over Σ with $\mathcal{L}(\mathcal{A}) = \mathcal{L}$.

Remark 4.1. Note that we do not demand for final configurations that $u = \varepsilon$ and we can read a pop letter even if the stack is empty (using the special stack symbol $\perp$). This permits to consider the situations where some pop letters are not balanced by push letters and vice versa.

We note that the visibly pushdown languages over $\tilde{\Sigma}$ form a proper subclass of the context-free languages over Σ, cf. [4] for further properties.

For any word $w = a_1 \ldots a_n \in \Sigma^+$, let $\textsc{Mask}(w) = b_1 \ldots b_n \in \{-1, 0, 1\}^+$ such that, for all $1 \leq i \leq n$, $b_i = 1$ if $a_i \in \Sigma^{\downarrow}$, $b_i = 0$ if $a_i \in \Sigma^{\#}$, and $b_i = -1$ otherwise. Let $\mathbb{L} \subseteq \{-1, 0, 1\}^*$ be the language which contains ε and all words $b_1 \ldots b_n \in \{-1, 0, 1\}^+$ such that $\sum_{j=1}^{n} b_j = 0$ and $\sum_{j=1}^{i} b_j \geq 0$ for all $i \in \{1, \ldots, n\}$. Here, we interpret 1 as the left parenthesis, -1 as the right parenthesis and 0 as an irrelevant symbol. Then, $\mathbb{L}$ is the set of all sequences with correctly nested parentheses.

Next, we turn to the logic $\textsc{Mso}_{\mathbb{L}}(\tilde{\Sigma})$ over the pushdown alphabet $\tilde{\Sigma}$ which extends the classical MSO logic on finite words by the binary relation which checks whether a push letter and a pop letter are matching. This logic was shown in [4] to be equivalent to visibly pushdown automata. The logic $\textsc{Mso}_{\mathbb{L}}(\tilde{\Sigma})$ is defined by the grammar

$$\varphi \ ::= \ P_a(x) \mid x \leq y \mid X(x) \mid \mathbb{L}(x, y) \mid \varphi \vee \varphi \mid \neg\varphi \mid \exists x.\varphi \mid \exists X.\varphi$$

where $a \in \Sigma$, $x, y \in \mathcal{V}_1$ and $X \in \mathcal{V}_2$. The formulas in $\mathrm{MSO}_{\mathbb{L}}(\tilde{\Sigma})$ are interpreted over a word $w = a_1...a_n \in \Sigma^+$ and a variable assignment $\sigma : \mathcal{V}_1 \cup \mathcal{V}_2 \to \mathrm{dom}(w) \cup 2^{\mathrm{dom}(w)}$. We will write $(w, \sigma) \models \mathbb{L}(x, y)$ iff $\sigma(x) < \sigma(y)$, $a_{\sigma(x)} \in \Sigma^{\downarrow}$, $a_{\sigma(y)} \in \Sigma^{\uparrow}$ and $\mathrm{MASK}(a_{\sigma(x)+1}...a_{\sigma(y)-1}) \in \mathbb{L}$. For other formulas, the satisfaction relation is defined as usual. If φ is a sentence, then the satisfaction relation does not depend on a variable assignment and we can simply write $w \models \varphi$. For a sentence $\varphi \in \mathrm{MSO}_{\mathbb{L}}(\tilde{\Sigma})$, let $\mathcal{L}(\varphi) = \{w \in \Sigma^+ \mid w \models \varphi\}$. We say that a language $\mathcal{L} \subseteq \Sigma^+$ is $\mathrm{MSO}_{\mathbb{L}}(\tilde{\Sigma})$-*definable* if there exists a sentence $\varphi \in \mathrm{MSO}_{\mathbb{L}}(\tilde{\Sigma})$ such that $\mathcal{L}(\varphi) = \mathcal{L}$.

The following result states the expressive equivalence of visibly pushdown automata and $\mathrm{MSO}_{\mathbb{L}}$-logic.

Theorem 4.2 (Alur, Madhusudan [4]). *Let $\tilde{\Sigma} = (\Sigma^{\downarrow}, \Sigma^{\#}, \Sigma^{\uparrow})$ be a pushdown alphabet, $\Sigma = \Sigma^{\downarrow} \cup \Sigma^{\#} \cup \Sigma^{\uparrow}$, and $\mathcal{L} \subseteq \Sigma^+$ a language. Then, $\mathcal{L}$ is a visibly pushdown language over $\tilde{\Sigma}$ iff $\mathcal{L}$ is $\mathrm{MSO}_{\mathbb{L}}(\tilde{\Sigma})$-definable.*

5 Decomposition of Timed Pushdown Automata

In this section we prove a Nivat-like (cf. [12, 5]) decomposition theorem for timed pushdown automata. This result establishes a connection between timed pushdown languages and visibly pushdown languages. We will use this theorem for the proof of our Theorem 3.2.

The key idea is to consider a timed pushdown language as a renaming of a timed pushdown language over an extended alphabet which encodes the information about clocks and stack; on the level of this extended alphabet we can separate the setting of visibly pushdown languages from the timed setting. The main difficulty here is to encode the infinite time domain, namely $\mathbb{R}_{\geq 0}$, as a finite alphabet. We will show that it suffices to use several partitions of $\mathbb{R}_{\geq 0}$ into intervals to construct the desired extended alphabet.

Let $\mathbb{N}_+ = \{1, 2, ...\}$ denote the set of positive natural numbers. For a finite set $K = \{n_1, n_2, ..., n_l\} \subseteq \mathbb{N}_+$ with $l \geq 0$ and $n_1 < n_2 < ... < n_l$, let

$$\mathbb{P}(K) = \{[0, 0], (0, n_1), [n_1, n_1], (n_1, n_2), ..., [n_l, n_l], (n_l, \infty)\} \subseteq 2^{\mathcal{I}},$$

the *K-interval partition* of $\mathbb{R}_{\geq 0}$. Note that $\mathbb{P}(K)$ is a finite set. For an alphabet Σ, $m \geq 0$ and finite sets $K_1, ..., K_m, S \subseteq \mathbb{N}_+$, let $\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$ denote the pushdown alphabet $\tilde{\Gamma} = (\Gamma^{\downarrow}, \Gamma^{\#}, \Gamma^{\uparrow})$ where, for $\delta \in \{\downarrow, \#, \uparrow\}$, $\Gamma^{\delta} = \Sigma \times \mathbb{P}(K_1) \times ... \times \mathbb{P}(K_m) \times \mathbb{P}(S) \times \{0, 1\}^m \times \{\delta\}$. Let $\Gamma = \mathcal{R}(\Sigma, K_1, ..., K_m, S) = \Gamma^{\downarrow} \cup \Gamma^{\#} \cup \Gamma^{\uparrow}$.

Let $\mathcal{T}(\Sigma, K_1, ..., K_m, S) \subseteq \mathbb{T}\Gamma^+$ denote the timed language defined as follows. Let $w = (b_1, t_1)...(b_n, t_n) \in \mathbb{T}\Gamma^+$ where, for all $i \in \{1, ..., n\}$, $b_i = (a_i, (k_i^1, ..., k_i^m), s_i, (u_i^1, ..., u_i^m), \delta_i)$ with $a_i \in \Sigma$, $k_i^1 \in \mathbb{P}(K_1)$, ..., $k_i^m \in \mathbb{P}(K_m)$, $s_i \in \mathbb{P}(S)$, $u_i^1, ..., u_i^m \in \{0, 1\}$ and $\delta_i \in \{\downarrow, \#, \uparrow\}$. Then, we let $w \in \mathcal{T}(\Sigma, K_1, ..., K_m, S)$ iff the following hold:

– $\mathrm{MASK}(b_1...b_n) \in \mathbb{L}$ (with respect to the pushdown alphabet $\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$);

– for all $i \in \{1, ..., n\}$ and $j \in \{1, ..., m\}$, letting $u_0^j = 1$, we have $\langle w \rangle_{i',i} \in k_i^j$ for the greatest $i' \in \{0, 1, ..., i-1\}$ with $u_{i'}^j = 1$;
– for all $i, i' \in \{1, ..., n\}$ with $i < i'$, $\delta_i = \downarrow$, $\delta_{i'} = \uparrow$ and $\textsc{Mask}(b_{i+1}...b_{i'-1}) \in \mathbb{L}$, we have $\langle w \rangle_{i,i'} \in s_{i'}$.

Clearly, the timed language $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$ is a non-empty timed pushdown language. Let Δ be an alphabet, $\mathcal{L} \subseteq \Delta^+$ a language and $\mathcal{L}' \subseteq \mathbb{T}\Delta^+$ a timed language. Let $(\mathcal{L} \cap \mathcal{L}') \subseteq \mathbb{T}\Delta^+$ be the "restriction" of $\mathcal{L}'$ to $\mathcal{L}$, i.e., the timed language consisting of all timed words $w = (b_1, t_1)...(b_n, t_n) \in \mathcal{L}'$ such that $b_1...b_n \in \mathcal{L}$. Let Δ, Δ' be alphabets and $h : \Delta \to \Delta'$ a renaming. For a timed word $w = (b_1, t_1)...(b_n, t_n) \in \mathbb{T}\Delta^+$, let $h(w) = (h(b_1), t_1)...(h(b_n), t_n)$. Then, for a timed language $\mathcal{L} \subseteq \mathbb{T}\Delta^+$, let $h(\mathcal{L}) = \{h(w) \mid w \in \mathcal{L}\}$, so $h(\mathcal{L}) \subseteq \mathbb{T}(\Delta')^+$.

Now we formulate our decomposition theorem. This result permits to separate the discrete part of TPDA from their timed part. We show that the discrete part can be described by visibly pushdown languages whereas the timed part can be described by means of timed languages $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$ which have the following interesting property. We can decide whether a timed word w belongs to $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$ by analyzing the components of w. In contrast, if we have a TPDA $\mathcal{A}$ and can use it only as a "black box", then we cannot say whether a timed word w is accepted by this TPDA $\mathcal{A}$ without passing w through $\mathcal{A}$.

Theorem 5.1. *Let Σ be an alphabet and $\mathcal{L} \subseteq \mathbb{T}\Sigma^+$ a timed language. Then the following are equivalent.*

(a) $\mathcal{L}$ is a timed pushdown language.
(b) There exist $m \geq 0$, finite sets $K_1, ..., K_m, S \subseteq \mathbb{N}_+$, a renaming $h : \mathcal{R}(\Sigma, K_1, ..., K_m, S) \to \Sigma$, and a visibly pushdown language $\mathcal{L}' \subseteq (\mathcal{R}(\Sigma, K_1, ..., K_m, S))^+$ over the pushdown alphabet $\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$ such that $\mathcal{L} = h(\mathcal{L}' \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S))$.

First, we show the implication (a) $\Rightarrow$ (b) of Theorem 5.1.

Lemma 5.2. *Let $\mathcal{A}$ be a timed pushdown automaton over Σ. Then, there exist $m \geq 0$, finite sets $K_1, ..., K_m, S \subseteq \mathbb{N}_+$, a renaming $h : \mathcal{R}(\Sigma, K_1, ..., K_m, S) \to \Sigma$, and a visibly pushdown automaton $\mathcal{A}'$ over the pushdown alphabet $\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$ such that $\mathcal{L}(\mathcal{A}) = h(\mathcal{L}(\mathcal{A}') \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S))$.*

Proof. Let $\mathcal{A} = (L, \Gamma, C, L_0, E, L_f)$. Let $m = |C|$ and $C = \{x_1, ..., x_m\}$. For each interval $J \in \mathcal{I}$, let $\mathcal{B}_J$ be the set of its bounds not in $\{0, \infty\}$, i.e., $\mathcal{B}_J = \{\inf J, \sup J\} \setminus \{0, \infty\}$. Then, we define the sets $K_1, ..., K_m, S$ as follows.

– For every $i \in \{1, ..., m\}$, let K_i be the union of all sets $\mathcal{B}_J$ where $J = \phi(x_i)$ for some clock constraint $\phi \in \mathcal{I}^C$ appearing in an edge of $\mathcal{A}$.
– Let S be the union of all sets $\mathcal{B}_J$ where J is any interval which appears in the $\mathcal{S}(\Gamma)$-component of some edge in $E^{\uparrow}$.

Let h be the projection to the Σ-component. Finally, we define the visibly pushdown automaton $\mathcal{A}' = (L, \Gamma, L_0, T, L_f)$ over the pushdown alphabet

$\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$ where the set $T = T^{\downarrow} \cup T^{\#} \cup T^{\uparrow}$ is defined as follows. We simulate every edge $e = (\ell, a, s, \phi, \Lambda, \ell') \in E$ with $\ell, \ell' \in L$, $s \in \mathcal{S}(\Gamma)$, $\phi \in \mathcal{I}^C$ and $\Lambda \subseteq C$ by (possibly multiple) transitions in T depending on the sort of e as follows.

- If $e \in E^{\downarrow}$ and $s = (\downarrow, \gamma)$ for some $\gamma \in \Gamma$, then we let $(\ell, a^{\downarrow}, \gamma, \ell') \in T^{\downarrow}$ for all $a^{\downarrow} = (a, (k_1, ..., k_m), \sigma, (u_1, ..., u_m), \downarrow) \in \mathcal{R}(\Sigma, K_1, ..., K_m, S)$ such that:
  - for all $j \in \{1, ..., m\}$, k_j is an interval in $\mathbb{P}(K_j)$ such that $k_j \subseteq \phi(x_j)$;
  - σ is an arbitary interval in $\mathbb{P}(S)$;
  - for all $j \in \{1, ..., m\}$, $u_j \in \{0, 1\}$, and $u_j = 1$ iff $x_j \in \Lambda$.
- If $e \in E^{\#}$, then we let $(\ell, a^{\#}, \ell') \in T^{\#}$ for all $a^{\#} = (a, (k_1, ..., k_m), \sigma, (u_1, ..., u_m), \#) \in \mathcal{R}(\Sigma, K_1, ..., K_m, S)$ where $k_1, ..., k_m, \sigma, u_1, ..., u_m$ are defined as in the previous case.
- If $e \in E^{\uparrow}$ and $s = (\uparrow, \gamma, I)$ for some $\gamma \in \Gamma$ and $I \in \mathcal{I}$, then we let $(l, a^{\uparrow}, \gamma, \ell') \in T^{\uparrow}$ for all $a^{\uparrow} = (a, (k_1, ..., k_m), \sigma, (u_1, ..., u_m), \uparrow)$ where $k_1, ..., k_m$, u_1, ..., u_m are defined as in the first case, and σ is an interval in $\mathbb{P}(S)$ such that $\sigma \subseteq I$. Note that we do not have transitions in $T^{\uparrow}$ whose stack letter is $\perp$.

Note that although the emptiness of the stack at the end of a run is not required by visibly pushdown automata, it is checked by intersection with the timed language $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$.

One can show that $\mathcal{L}(\mathcal{A}) = h(\mathcal{L}(\mathcal{A}') \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S))$. $\square$

Now we turn to the converse direction of Theorem 5.1.

Lemma 5.3. *Let $m \geq 0$, $K_1, ..., K_m, S \subseteq \mathbb{N}_+$ be finite sets and $\mathcal{A}$ a visibly pushdown automaton over $\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$. Then, there exists a TPDA $\mathcal{A}'$ over the alphabet $\mathcal{R}(\Sigma, K_1, ..., K_m, S)$ such that $\mathcal{L}(\mathcal{A}') = \mathcal{L}(\mathcal{A}) \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S)$.*

Proof. Let $\mathcal{A} = (L, \Gamma, L_0, T, L_f)$. We put $\mathcal{A}' = (L, \Gamma, C, L_0, E, L_f)$ where $C = \{1, ..., m\}$ and $E = E^{\downarrow} \cup E^{\#} \cup E^{\uparrow}$ is defined as follows.

- For every $t = (\ell, a^{\downarrow}, \gamma, \ell') \in T^{\downarrow}$ with $a^{\downarrow} = (a, (k_1, ..., k_m), \sigma, (u_1, ..., u_m), \downarrow)$ (where $k_1 \in \mathbb{P}(K_1)$, ..., $k_m \in \mathbb{P}(K_m)$, $\sigma \in \mathbb{P}(S)$, $u_1, ..., u_m \in \{0, 1\}$), we let $(\ell, a^{\downarrow}, (\downarrow, \gamma), \phi, \Lambda, \ell') \in E^{\downarrow}$ where $\phi(c) = k_c$ for all $c \in C$ and $\Lambda = \{c \in C \mid u_c = 1\}$.
- For every $t = (\ell, a^{\#}, \ell') \in T^{\#}$ with $a^{\#} = (a, (k_1, ..., k_m), \sigma, (u_1, ..., u_m), \#)$, we let $(\ell, a^{\#}, \#, \phi, \Lambda, \ell') \in E^{\#}$ where ϕ and Λ are defined as in the previous case.
- For every $t = (\ell, a^{\uparrow}, \gamma, \ell') \in T^{\uparrow}$ with $a^{\uparrow} = (a, (k_1, ..., k_m), \sigma, (u_1, ..., u_m), \uparrow)$ and $\gamma \in \Gamma$, we let $(\ell, a^{\uparrow}, (\uparrow, \gamma, I), \phi, \Lambda, \ell') \in E^{\uparrow}$ where $I = \sigma$ and ϕ, Λ are defined as in the first case. Note that we do not have to simulate the transitions in $T^{\uparrow}$ whose stack symbol is $\perp$, since their use leads to the acceptance of words not in $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$.

Note that although $\mathcal{A}$ may accept words with a non-empty stack after executing a run, these words (as untimed parts of timed words) are excluded after taking the intersection with $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$.

One can show that $\mathcal{L}(\mathcal{A}') = \mathcal{L}(\mathcal{A}) \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S)$. $\square$

The following lemma shows that the class of timed pushdown languages is closed under renamings.

Lemma 5.4. *Let Δ, Δ' be alphabets, $h : \Delta' \to \Delta$ a renaming, and $\mathcal{A}$ a TPDA over Δ'. Then, there exists a TPDA $\mathcal{A}'$ over Δ such that $\mathcal{L}(\mathcal{A}') = h(\mathcal{L}(\mathcal{A}))$.*

Proof. Let $\mathcal{A} = (L, \Gamma, C, L_0, E, L_f)$. Then, we put $\mathcal{A}' = (L, \Gamma, C, L_0, E', L_f)$ with $E' = \{(\ell, h(a), s, \phi, \Lambda, \ell') \mid (\ell, a, s, \phi, \Lambda, \ell') \in E\}$. □

Then, Theorem 5.1 follows immediately from Lemmas 5.2, 5.3 and 5.4.

As a corollary of Theorem 5.1 and its proof, we deduce a decomposition theorem for timed automata. These may be considered as TPDA whose sets of push and pop edges are empty (and hence a stack alphabet is irrelevant for their definition). We slightly modify the extended alphabet needed for the decomposition by excluding the components relevant for the stack. Moreover, instead of visibly pushdown languages we consider the classical regular languages. For $m \geq 0$ and finite sets $K_1, ..., K_m \subseteq \mathbb{N}_+$, let $\Gamma := \mathcal{R}_0(\Sigma, K_1, ..., K_m) = \Sigma \times \mathbb{P}(K_1) \times ... \times \mathbb{P}(K_m) \times \{0,1\}^m$. We define the timed language $\mathcal{T}_0(\Sigma, K_1, ..., K_m) \subseteq \mathbb{T}\Gamma^+$ as follows. Let $w = (b_1, t_1)...(b_n, t_n) \in \mathbb{T}\Gamma^+$ where, for all $i \in \{1, ..., n\}$, $b_i = (a_i, (k_i^1, ..., k_i^m), (u_i^1, ..., u_i^m))$ with $a_i \in \Sigma$, $k_i^1 \in \mathbb{P}(K_1)$, ..., $k_i^m \in \mathbb{P}(K_m)$ and $u_i^1, ..., u_i^m \in \{0,1\}$. Then, $w \in \mathcal{T}_0(\Sigma, K_1, ..., K_m)$ iff, for all $i \in \{1, ..., n\}$ and $j \in \{1, ..., m\}$, letting $u_0^j = 1$, we have $\langle w \rangle_{i',i} \in k_i^j$ for the greatest $i' \in \{0, 1, ..., i-1\}$ with $u_{i'}^j = 1$.

Corollary 5.5. *Let Σ be an alphabet and $\mathcal{L} \subseteq \mathbb{T}\Sigma^+$ a timed language. Then the following are equivalent.*

(a) $\mathcal{L}$ is recognizable by a timed automaton.
(b) There exist $m \geq 0$, finite sets $K_1, ..., K_m \subseteq \mathbb{N}_+$, a renaming $h : \mathcal{R}_0(\Sigma, K_1, ..., K_m) \to \Sigma$ and a regular language $\mathcal{L}' \subseteq \Sigma^+$ such that $\mathcal{L} = h(\mathcal{L}' \cap \mathcal{T}_0(\Sigma, K_1, ..., K_m))$.

6 Recognizability Equals Definability

6.1 Definability Implies Recognizability

In this subsection, we show that TML-definable timed languages are recognizable by timed pushdown automata.

Theorem 6.1. *Let Σ be an alphabet and $\psi \in$ TML(Σ) a sentence. Then, there exists a TPDA $\mathcal{A}$ over Σ such that $\mathcal{L}(\mathcal{A}) = \mathcal{L}(\psi)$.*

To prove this theorem, let $\psi = \exists^{\text{match}}\mu.\exists D_1. \ ... \ \exists D_m.\varphi \in$ TML(Σ) with $m \geq 0$. We may assume that $D_1, ..., D_m \in \mathcal{D}$ are pairwise distinct variables and that, whenever $\mathrm{d}^\kappa(D, x)$ or $\mu^\kappa(x, y)$ is a subformula of φ, then $\kappa \in \{\leq, \geq\} \times \mathbb{N}$.

We wish to use Theorem 5.1. As preparation for this, we prove the following technical lemma.

Lemma 6.2. *There exist finite sets* $K_1, ..., K_m, S \subseteq \mathbb{N}_+$*, a renaming* $h : \mathcal{R}(\Sigma, K_1, ..., K_m, S) \to \Sigma$ *and a sentence* $\varphi^* \in \mathrm{MSO}_{\mathbb{L}}(\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S))$ *such that* $\mathcal{L}(\psi) = h(\mathcal{L}(\varphi^*) \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S))$.

Proof. The idea here is the following. First, we encode in φ the values of the variables $D_1, ..., D_m, \mu$ and the time constraints appearing in subformulas of the form $\mathrm{d}^{\kappa}(D, x)$ and $\mu^{\kappa}(x, y)$ as an input word over the visibly pushdown alphabet $\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$. Then, all time constraints and the correctness of a matching relation are checked by the intersection of the visibly pushdown language $\mathcal{L}(\varphi^*)$ with the timed language $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$. Finally, we remove the auxiliary components of the extended alphabet by applying the projection h.

For every $j \in \{1, ..., m\}$, let K_j be the set of all $n \in \mathbb{N}_+$ such that $\mathrm{d}^{\bowtie n}(D_j, x)$ is a subformula of φ for some $x \in \mathcal{V}_1$ and $\bowtie \in \{\leq, \geq\}$. Let S be the set of all $n \in \mathbb{N}_+$ such that $\mu^{\bowtie n}(x, y)$ is a subformula of φ for some $x, y \in \mathcal{V}_1$ and $\bowtie \in \{\leq, \geq\}$. For simplicity, let $\mathcal{R} = \mathcal{R}(\Sigma, K_1, ..., K_m, S)$, $\tilde{\mathcal{R}} = \tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$ and $\mathcal{T} = \mathcal{T}(\Sigma, K_1, ..., K_m, S)$. Let $\Omega = \mathbb{P}(K_1) \times ... \times \mathbb{P}(K_m)$ and $\Delta = \{\downarrow, \#, \uparrow\}$. We will denote a letter of $\mathcal{R}$ by $(a, \mathcal{K}, s, \mathcal{G}, \delta)$ where $a \in \Sigma$, $\mathcal{K} \in \Omega$, $s \in \mathbb{P}(S)$, $\mathcal{G} \in \{0, 1\}^m$ and $\delta \in \Delta$. Let $h : \mathcal{R} \to \Sigma$ be the projection to the Σ-component. Finally, we define the formula $\varphi^* \in \mathrm{MSO}_{\mathbb{L}}(\tilde{R})$ from the formula φ by the following substitutions.

- If $P_a(x)$ with $a \in \Sigma$ and $x \in \mathcal{V}_1$ is a subformula of φ, then $P_a(x)$ is replaced by the formula $\bigvee(P_{(a,\mathcal{K},s,\mathcal{G},\delta)}(x) \mid \mathcal{K} \in \Omega, s \in \mathbb{P}(S), \mathcal{G} \in \{0, 1\}^m, \delta \in \Delta)$.
- If $D_j(x)$ with $j \in \{1, ..., m\}$ and $x \in \mathcal{V}_1$ appears in φ, then $D_j(x)$ is replaced by the formula $\bigvee(P_{(a,\mathcal{K},s,\mathcal{G},\delta)}(x) \mid a \in \Sigma, \mathcal{K} \in \Omega, s \in \mathbb{P}(S)$, $\mathcal{G} = (g_1, ..., g_m) \in \{0, 1\}^m$ with $g_j = 1, \delta \in \Delta)$.
- If $\mathrm{d}^J(D_j, x)$ is a subformula of φ where $j \in \{1, ..., m\}$, $x \in \mathcal{V}_1$ and $J \in \mathcal{I}$ is an interval of the form $[0, n]$ or $[n, \infty)$ for some $n \in \mathbb{N}$, then we replace $\mathrm{d}^J(D_j, x)$ by the formula $\bigvee(P_{(a,\mathcal{K},s,\mathcal{G},\delta)}(x) \mid a \in \Sigma,\ \mathcal{K} = (k_1, ..., k_m) \in \Omega$ with $k_j \subseteq J, s \in \mathbb{P}(S), \mathcal{G} \in \{0, 1\}^m, \delta \in \Delta)$. Note that we remove the variable D_j from φ.
- If $\mu^J(x, y)$ is a subformula of φ where $x, y \in \mathcal{V}_1$ and $J \in \mathcal{I}$ is an interval of the form $[0, n]$ or $[n, \infty)$ for some $n \in \mathbb{N}$, then $\mu^J(x, y)$ is replaced by the formula $\mathbb{L}(x, y) \wedge \bigvee(P_{(a,\mathcal{K},s,\mathcal{G},\uparrow)}(y) \mid a \in \Sigma, \mathcal{K} \in \Omega, s \in \mathbb{P}(S)$ with $s \subseteq J, \mathcal{G} \in \{0, 1\}^m)$. Note that here we replace the matching relation μ by the matching relation $\mathbb{L}$ with respect to the visibly pushdown alphabet $\tilde{\mathcal{R}}$ and measure the time distance using the $\mathbb{P}(S)$-component of the extended alphabet.

Note that φ^* is a sentence, since $\mathrm{Free}(\varphi) \subseteq \{D_1, ..., D_m, \mu\}$ and we removed $D_1, ..., D_m, \mu$ when constructing φ^*. It remains to show that $\mathcal{L}(\psi) = h(\mathcal{L}(\varphi^*) \cap \mathcal{T})$.

First, we show the inclusion $\subseteq$. Let $w = (a_1, t_1)...(a_n, t_n) \in \mathcal{L}(\psi)$ and σ be a fixed w-assignment. Then, there exist a matching $M \subseteq \mathrm{dom}(w) \times \mathrm{dom}(w)$ and sets $I_1, ..., I_m \subseteq \mathrm{dom}(w)$ such that $(w, \sigma[D_1/I_1, ..., D_m/I_m, \mu/M]) \models \varphi$. We construct a word $w^* = r_1...r_n \in \mathcal{R}^+$ where, for all $i \in \{1, ..., n\}$,

$r_i = (a_i, (k_i^1, ..., k_i^m), s_i, (g_i^1, ..., g_i^m), \delta_i)$ with $a_i \in \Sigma$, $(k_i^1, ..., k_i^m) \in \Omega$, $s_i \in \mathbb{P}(S)$, $g_i^1, ..., g_i^m \in \{0, 1\}$, $\delta_i \in \Delta$ as follows.

- For all $i, i' \in \{1, ..., n\}$ with $(i, i') \in M$, we put $\delta_i = \downarrow$, $\delta_{i'} = \uparrow$, $s_i = [0, 0]$ (in principle, s_i can be chosen arbitrarily) and let $s_{i'}$ be the interval in the partition $\mathbb{P}(S)$ with $\langle w \rangle_{i,i'} \in s_{i'}$ (uniquely determined).
- For all $i \in \{1, ..., n\}$ such that i does not belong to any pair in M, we put $\delta_i = \#$ and $s_i = [0, 0]$ (again, s_i can be chosen arbitrarily).
- For all $i \in \{1, ..., n\}$ and $j \in \{1, ..., m\}$, we let $g_i^j = 1$ iff $i \in I_j$.
- For all $i \in \{1, ..., n\}$ and $j \in \{1, ..., m\}$, we let k_i^j be the interval in the partition $\mathbb{P}(K_j)$ such that $(I_j, i) \in \mathrm{d}^{k_i^j}(w)$.

Let $w^{**} := (r_1, t_1)...(r_n, t_n) \in \mathbb{T}\mathcal{R}^+$. Clearly, $w^{**} \in \mathcal{T}$ and $h(w^{**}) = w$. It can be also shown by induction on the structure of φ that $w^* \in \mathcal{L}(\varphi^*)$.

The converse inclusion $\supseteq$ can be shown in a similar fashion. □

Then, Theorem 6.1 follows from Lemma 6.2, Theorem 4.2 and Theorem 5.1, implication (b) ⇒ (a).

6.2 Recognizability Implies Definability

In this subsection, we show the converse direction of Theorem 3.2.

Theorem 6.3. *Let Σ be an alphabet and $\mathcal{A}$ a timed pushdown automaton over Σ. Then, there exists a sentence $\psi \in \mathrm{TML}(\Sigma)$ such that $\mathcal{L}(\psi) = \mathcal{L}(\mathcal{A})$.*

The proof of this theorem will be given in the rest of this subsection. Again, we will apply our decomposition Theorem 5.1 for TPDA.

Lemma 6.4. *Let Σ be an alphabet, $m \geq 0$, $K_1, ..., K_m, S \subseteq \mathbb{N}_+$ finite sets, $h : \mathcal{R}(\Sigma, K_1, ..., K_m, S) \to \Sigma$ a renaming, and $\varphi \in \mathrm{MSO}_{\mathbb{L}}(\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S))$ a sentence. Then, there exists a sentence $\psi \in \mathrm{TML}(\Sigma)$ such that $\mathcal{L}(\psi) = h(\mathcal{L}(\varphi) \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S))$.*

Proof. For the proof, we follow a similar approach as in [8], Theorem 6.6. Let $\Gamma = \mathcal{R}(\Sigma, K_1, ..., K_m, S)$, $\overline{X} = \{X_\gamma \in \mathcal{V}_2 \mid \gamma \in \Gamma\}$ be a set of pairwise distinct variables not appearing in φ, and $\overline{D} = \{D_j \in \mathcal{D} \mid 1 \leq j \leq m\}$ a set of pairwise distinct relative distance variables. Using the $\overline{X}$-variables, we will describe the renaming h (i.e., we store in these second-order variables the positions of the letters of the extended alphabet Γ before the renaming). We will use the $\overline{D}$-variables as well as the matching variable μ to describe the timed language $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$. Moreover, we transform φ into a $\mathrm{TMSO}(\Sigma)$-formula.

Let $\varphi^* \in \mathrm{TMSO}(\Sigma)$ be the formula obtained from φ by the following transformations.

- Every subformula $\mathbb{L}(x, y)$ of φ with $x, y \in \mathcal{V}_1$ is replaced by $\mu(x, y)$.
- Every subformula $P_\gamma(x)$ of φ with $x \in \mathcal{V}_1$ and $\gamma \in \Gamma$ is replaced by the formula $P_{h(\gamma)}(x) \wedge X_\gamma(x)$.

The formula $\textsc{Partition} = \big(\forall x.\big[\bigvee_{\gamma\in\Gamma}\big(X_\gamma(x) \wedge \bigwedge_{\gamma'\neq\gamma}\neg X_{\gamma'}(x)\big)\big]\big)$ demands that values of $\overline{X}$-variables form a partition of the domain. The formula $\textsc{Renaming} = \forall x.\big(\bigvee_{\gamma\in\Gamma}\big(X_\gamma(x) \to P_{h(\gamma)}(x)\big)\big)$ correlates values of $\overline{X}$-variables with an input word. It remains to handle $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$. For a letter $\gamma = (a, (k_1, ..., k_m), s, (g_1, ..., g_m), \delta) \in \Gamma$ with $a \in \Sigma$, $k_1 \in \mathbb{P}(K_1)$, ..., $k_m \in \mathbb{P}(K_m)$, $s \in \mathbb{P}(S)$, $g_1, ..., g_m \in \{0, 1\}$ and $\delta \in \{\downarrow, \#, \uparrow\}$, we write $k_j(\gamma) = k_j$ $(1 \leq j \leq m)$, $s(\gamma) = s$, $g_j(\gamma) = g_j$ $(1 \leq j < m)$ and $\delta(\gamma) = \delta$. For $x \in \mathcal{V}_1$ and $I \in \mathcal{I}$ and $\delta \in \{\downarrow, \#, \uparrow\}$, let the formula $M^I_\delta(x)$ be defined as follows.

- If $\delta = \downarrow$, then $M^I_\delta(x) = \exists y.\big(\mu(x, y) \wedge \bigvee_{\substack{\gamma\in\Gamma,\\ \delta(\gamma)=\uparrow}} X_\gamma(y)\big)$.
- If $\delta = \#$, then $M^I_\delta(x) = \neg\exists y.(\mu(x, y) \vee \mu(y, x))$.
- If $\delta = \uparrow$, then $M^I_\delta(x) = \exists y.\big(\mu^I(y, x) \wedge \bigvee_{\substack{\gamma\in\Gamma,\\ \delta(\gamma)=\downarrow}} X_\gamma(y)\big)$.

For $j \in \{1, ..., m\}$ and $x \in \mathcal{V}_1$, let $D^1_j(x) = D_j(x)$ and $D^0_j(x) = \neg D_j(x)$. Consider the $\textsc{tMso}(\Sigma)$-formula

$$\xi = \forall x. \bigwedge_{\gamma\in\Gamma} X_\gamma(x) \to \left(\bigwedge_{j=1}^{m}(\mathrm{d}^{k_j(\gamma)}(D_j, x) \wedge D_j^{g_j(\gamma)}(x)) \wedge M^{s(\gamma)}_{\delta(\gamma)}(x)\right)$$

which takes care of matchings and time. Let $(\gamma_i)_{i\in\{1,...,|\Gamma|\}}$ be an enumeration of Γ. Then, we define the desired sentence ψ as

$$\psi = \exists^{\text{match}}\mu.\exists D_1. \ldots \exists D_m.\exists X_1. \ldots \exists X_{|\Gamma|}.(\varphi^* \wedge \textsc{Partition} \wedge \textsc{Renaming} \wedge \xi).$$

One can show that $\mathcal{L}(\psi) = h(\mathcal{L}(\varphi) \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S))$. $\square$

Then, Theorem 6.3 follows immediately from Theorems 5.1 and 4.2 and Lemma 6.4.

Remark 6.5. Alternatively, Theorem 6.3 can be proved by a direct translation of $\mathcal{A}$ into ψ. However, using Theorem 5.1, it suffices to describe a more simple timed language $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$ and a projection h to adopt a logical description of a visibly pushdown language of [4]. In particular, here we do not have to describe some technical details like initial and final states as well as concatenations of transitions.

7 Conclusion and Future Work

In this paper, we introduced a timed matching logic and showed that this logic is equally expressive as timed pushdown automata (and hence the satisfiability problem for our timed matching logic is decidable). When proving our main result, we showed a Nivat-like decomposition theorem for timed pushdown automata. This theorem seems to be the first algebraic characterization of timed pushdown languages and may be of independent interest. Based on the ideas presented in [7, 8, 11, 13] and the ideas of this paper, our ongoing research concerns a logical characterization for *weighted timed pushdown automata* [2]. It could be also interesting to investigate such an extension of timed pushdown automata where each edge permits to push or pop several stack elements. Our conjecture is that this extended model is more expressive than timed pushdown automata as considered in this paper.

References

[1] Abdulla, P.A., Atig, M.F., Stenman, J.: Dense-timed pushdown automata. In: LICS 2012, pp. 35–44. IEEE Computer Society (2012)

[2] Abdulla, P.A., Atig, M.F., Stenman, J.: Computing optimal reachability costs in priced dense-timed pushdown automata. In: LATA 2014. LNCS, vol. 8370, pp. 62–75. Springer (2014)

[3] Alur, R., Dill, D.L.: A theory of timed automata. Theoretical Computer Science 126(2). 183–235 (1994)

[4] Alur, R., Madhusudan, P.: Visibly pushdown languages. In: STOC 2004, pp. 202–211. ACM (2004)

[5] Berstel, J.: Transductions and Context-Free Languages. Teubner Studienbücher: Informatik. Teubner, Stuttgart (1979)

[6] Büchi, J.R.: Weak second order arithmetic and finite automata. Zeitschrift für Mathematische Logik und Grundlagen der Informatik 6, 66–92 (1960)

[7] Droste, M., Gastin, P.: Weighted automata and weighted logics. Theoret. Comp. Sci. 380(1-2), 69–86 (2007)

[8] Droste, M., Perevoshchikov, V.: A Nivat theorem for weighted timed automata and relative distance logic. In: ICALP 2014. LNCS, pp. 171–182. Springer (2014)

[9] Elgot, C.C.: Decision problems of finite automata design and related arithmetics. Trans. Amer. Math. Soc. 98, 21–51 (1961)

[10] Lautemann, C., Schwentick, T., Thérien, D.: Logics for context-free languages. In: CSL 1994. LNCS, vol. 933, pp. 205–216. Springer (1994)

[11] Mathissen, Ch.: Weighted logics for nested words and algebraic formal power series. In: ICALP 2008. LNCS, vol. 5126, pp 221–232. Springer (2008)

[12] Nivat, M.: Transductions des langages de Chomsky. Ann. de l'Inst. Fourier 18, 339–456 (1968)

[13] Quaas, K.: MSO logics for weighted timed automata. Formal Methods in System Design 38(3), 193–222 (2011)

[14] Thatcher, J.W., Wright, J.B.: Generalized finite automata theory with an application to a decision problem of second-order logic. Mathematical System Theory, 2:57–81 (1968)

[15] Wilke, T.: Specifying timed state sequences in powerful decidable logics and timed automata. In: Formal Techniques in Real-Time and Fault-Tolerant Systems 1994. LNCS, vol. 863, pp. 694 – 715. Springer (1994)

A Decomposition of Timed Pushdown Automata

Here, we present the full proofs of Lemmas 5.2 and 5.3.

Lemma A.1. *Let $\mathcal{A}$ be a timed pushdown automaton over Σ. Then, there exist $m \geq 0$, finite sets $K_1, ..., K_m, S \subseteq \mathbb{N}_+$, a renaming $h : \mathcal{R}(\Sigma, K_1, ..., K_m, S) \to \Sigma$, and a visibly pushdown automaton $\mathcal{A}'$ over the pushdown alphabet $\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$ such that $\mathcal{L}(\mathcal{A}) = h(\mathcal{L}(\mathcal{A}') \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S))$.*

Proof. Let $\mathcal{A} = (L, \Gamma, C, L_0, E, L_f)$. Let $m = |C|$ and $C = \{x^1, ..., x^m\}$. For each interval $J \in \mathcal{I}$, let $\mathcal{B}_J$ be the set of its bounds not in $\{0, \infty\}$, i.e., $\mathcal{B}_J = \{\inf J, \sup J\} \setminus \{0, \infty\}$. Then, we define the sets $K_1, ..., K_m, S$ as follows.

- For every $i \in \{1, ..., m\}$, let K_i be the union of all sets $\mathcal{B}_J$ where $J = \phi(x^i)$ for some clock constraint $\phi \in \mathcal{I}^C$ appearing in an edge of $\mathcal{A}$.
- Let S be the union of all sets $\mathcal{B}_J$ where J is any interval which appears in the $\mathcal{S}(\Gamma)$-component of some edge in $E^{\uparrow}$.

Let $\Pi = \mathcal{R}(\Sigma, K_1, ..., K_m, S)$ and $h : \Pi \to \Sigma$ be the projection to the Σ-component. Finally, we define the visibly pushdown automaton $\mathcal{A}' = (L, \Gamma, L_0, T, L_f)$ over the pushdown alphabet $\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$ where the set $T = T^{\downarrow} \cup T^{\#} \cup T^{\uparrow}$ is defined as follows. We simulate every edge $e = (\ell, a, s, \phi, \Lambda, \ell') \in E$ with $\ell, \ell' \in L$, $s \in \mathcal{S}(\Gamma)$, $\phi \in \mathcal{I}^C$ and $\Lambda \subseteq C$ by (possibly multiple) transitions in T depending on the sort of e as follows.

- If $e \in E^{\downarrow}$ and $s = (\downarrow, \gamma)$ for some $\gamma \in \Gamma$, then we let $(\ell, a^{\downarrow}, \gamma, \ell') \in T^{\downarrow}$ for all $a^{\downarrow} = (a, (k^1, ..., k^m), \sigma, (u^1, ..., u^m), \downarrow) \in \Pi$ such that:
  - for all $j \in \{1, ..., m\}$, k^j is an interval in $\mathbb{P}(K_j)$ such that $k^j \subseteq \phi(x^j)$;
  - σ is an arbitary interval in $\mathbb{P}(S)$;
  - for all $j \in \{1, ..., m\}$, $u^j \in \{0, 1\}$, and $u^j = 1$ iff $x^j \in \Lambda$.
- If $e \in E^{\#}$, then we let $(\ell, a^{\#}, \ell') \in T^{\#}$ for all $a^{\#} = (a, (k^1, ..., k^m), \sigma, (u^1, ..., u^m), \#) \in \Pi$ where $k^1, ..., k^m, \sigma, u^1, ..., u^m$ are defined as in the previous case.
- If $e \in E^{\uparrow}$ and $s = (\uparrow, \gamma, I)$ for some $\gamma \in \Gamma$ and $I \in \mathcal{I}$, then we let $(l, a^{\uparrow}, \gamma, \ell') \in T^{\uparrow}$ for all $a^{\uparrow} = (a, (k^1, ..., k^m), \sigma, (u^1, ..., u^m), \uparrow) \in \Pi$ where $k^1, ..., k^m$, u^1, ..., u^m are defined as in the first case, and σ is an interval in $\mathbb{P}(S)$ such that $\sigma \subseteq I$. Note that we do not have transitions in $T^{\uparrow}$ whose stack letter is $\perp$.

Note that although the emptiness of the stack at the end of a run is not required by visibly pushdown automata, it is checked by intersection with the timed language $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$.

We denote the timed language $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$ simply by $\mathcal{T}$. It remains to prove that $\mathcal{L}(\mathcal{A}) = h(\mathcal{L}(\mathcal{A}') \cap \mathcal{T})$.

First, we show $\subseteq$. Let $w = (a_1, t_1)...(a_n, t_n) \in \mathbb{T}\Sigma^+$ be a timed word such that $w \in \mathcal{L}(\mathcal{A})$. Then, there exists a run

$$\rho = \langle \ell_0, \nu_0, r_0 \rangle \vdash_{t_1, e_1} \langle \ell_1, \nu_1, r_1 \rangle \vdash_{t_2, e_2} ... \vdash_{t_n, e_n} \langle \ell_n, \nu_n, r_n \rangle$$

of $\mathcal{A}$ with $\text{label}(\rho) = w$ such that

- for all $i \in \{0, ..., n\}$, $\ell_i \in L$, $\nu_i \in \mathbb{R}^C_{\geq 0}$ and $r_i \in \mathbb{T}\Gamma^*$;
- for all $i \in \{1, ..., n\}$, $e_i = (\ell_{i-1}, a_i, s_i, \phi_i, \Lambda_i, \ell_i)$ for some $s_i \in \mathcal{S}(\Gamma)$, $\phi_i \in \Phi(C)$ and $\Lambda_i \subseteq C$. Assume that

$$s_i = \begin{cases} (\downarrow, \gamma_i), & \text{if } e_i \in E^{\downarrow}, \\ \#, & \text{if } e_i \in E^{\#}, \\ (\uparrow, \gamma_i, I_i), & \text{if } e_i \in E^{\uparrow}. \end{cases}$$

  where $\gamma_i \in \Gamma$ and $I_i \subseteq \mathcal{I}$.

For all $i \in \{0, ..., n\}$, assume that $r_i = (g_i^1, \tau_i^1)...(g_i^{p_i}, \tau_i^{p_i})$ where $p_i \geq 0$, $g_i^1, ..., g_i^{p_i} \in \Gamma$ and $\tau_i^1, ..., \tau_i^{n_i} \in \mathbb{R}_{\geq 0}$. Let $\overline{g}_i = g_i^1 ... g_i^{p_i} \in \Gamma^*$. Clearly, $\overline{g}_0 = \varepsilon$.

We show that there exist $\pi_1, ..., \pi_n \in \Pi$ such that:

(i) $h(\pi_i) = a_i$ for all $i \in \{1, ..., n\}$;
(ii) $\pi_1 ... \pi_n \in \mathcal{L}(\mathcal{A}')$;
(iii) $(\pi_1, t_1)...(\pi_n, t_n) \in \mathcal{T}$.

For every $i \in \{1, ..., n\}$, we let $\pi_i = (a_i, (k_i^1, ..., k_i^m), \sigma_i, (u_i^1, ..., u_i^m), \delta_i)$ defined as follows.

- For all $j \in \{1, ..., m\}$, let $k_i^j \in \mathbb{P}(K_j)$ be the interval such that $\nu_{i-1}(x^j) + t_i \in k_i^j$. Since ρ is a run of $\mathcal{A}$, we have $k_i^j \subseteq \phi(x_j)$.
- If $e_i \in E^{\downarrow} \cup E^{\#}$, then we let $\sigma_i = [0, 0]$. Otherwise, if $e_i \in E^{\uparrow}$, we let $\sigma_i \in \mathbb{P}(S)$ be the interval containing $\tau_{i-1}^1 + t_i$ (note that $p_{i-1} \geq 1$ since ρ is a run of $\mathcal{A}$). Since ρ is a run of $\mathcal{A}$, we have for i with $e_i \in E^{\uparrow}$: $\sigma_i \subseteq I_i$.
- For all $j \in \{1, ..., m\}$, $u_i^j = \begin{cases} 1, & \text{if } x^j \in \Lambda_i, \\ 0, & \text{otherwise.} \end{cases}$
- Let $\delta_i = \begin{cases} \downarrow, & \text{if } e_i \in E^{\downarrow}, \\ \#, & \text{if } e_i \in E^{\#}, \\ \uparrow, & \text{if } e_i \in E^{\uparrow} \end{cases}$

Clearly, (i) holds. Now we show (ii). Let

$$\varrho = \langle \ell_0, \overline{g}_0 \rangle \vdash_{e'_1} \langle \ell_1, \overline{g}_1 \rangle \vdash_{e'_2} ... \vdash_{e'_n} \langle \ell_n, \overline{g}_n \rangle$$

where, for all $i \in \{1, ..., n\}$ with $e_i \in E^{\downarrow} \cup E^{\uparrow}$, we have $e'_i = (\ell_{i-1}, \pi_i, \gamma_i, \ell_i) \in T$ and, for all $i \in \{1, ..., n\}$ with $e_i \in E^{\#}$, we have $e'_i = (\ell_{i-1}, \pi_i, \ell_i) \in T$. Then, ϱ is a run of $\mathcal{A}'$ and $\text{label}(\varrho) = \pi_1 ... \pi_n$. Then, $\pi_1 ... \pi_n \in \mathcal{L}(\mathcal{A}')$.

Next, we show (iii).

- Since ρ is a run of $\mathcal{A}$, it is easy to see that $\textsc{Mask}(\pi_1 ... \pi_n) \in \mathbb{L}$.
- Let $i \in \{1, ..., n\}$ and $j \in \{1, ..., m\}$ and assume that $u_0^j = 1$. Then, for the greatest $i' \in \{0, 1, ..., i-1\}$ with $u_{i'}^j = 1$, we have: $\langle w \rangle_{i', i} = \nu_{i-1}(x^j) + t_i \in k_i^j$.

– Let $i, i' \in \{1, ..., n\}$ with $i < i'$, $\delta_i = \downarrow$, $\delta_{i'} = \uparrow$ and $\textsc{Mask}(\pi_{i+1}...\pi_{i'-1}) \in \mathbb{L}$. This means that, at the position i, some letter $\gamma \in \Gamma$ was pushed into the stack and at the position i', this letter was popped. Then, $\langle w \rangle_{i,i'} = \tau^1_{i-1} + t_i \in I_i$.

Then, $(\pi_1, t_1)...(\pi_n, t_n) \in \mathcal{T}$. Hence, $\mathcal{L}(\mathcal{A}) \subseteq h(\mathcal{L}(\mathcal{A}') \cap \mathcal{T})$.

Now we show the converse inclusion $\mathcal{L}(\mathcal{A}) \supseteq h(\mathcal{L}(\mathcal{A}') \cap \mathcal{T})$. Let $w = (a_1, t_1)...(a_n, t_n) \in h(\mathcal{L}(\mathcal{A}') \cap \mathcal{T})$. Then, there exist $\pi_1, ..., \pi_n \in \Pi$ such that:

(i) $h(\pi_i) = a_i$ for all $i \in \{1, ..., n\}$;
(ii) $\pi_1...\pi_n \in \mathcal{L}(\mathcal{A}')$;
(iii) $(\pi_1, t_1)...(\pi_n, t_n) \in \mathcal{T}$.

By (i), assume that, for all $i \in \{1, ..., n\}$, $\pi_i = (a_i, (k_i^1, ..., k_i^m), \sigma_i, (u_i^1, ..., u_i^m), \delta_i)$ where $k_i^1 \in \mathbb{P}(K_1)$, ..., $k_i^m \in \mathbb{P}(K_m)$, $\sigma_i \in \mathbb{P}(S)$, $u_i^1, ..., u_i^m \in \{0, 1\}$ and $\delta_i \in \{\downarrow, \#, \uparrow\}$. By (ii), there exists a run ϱ of $\mathcal{A}'$ with label$(\varrho) = \pi_1...\pi_n$. Assume that

$$\varrho = \langle \ell_0, \overline{g}_0 \rangle \vdash_{e'_1} \langle \ell_1, \overline{g}_1 \rangle \vdash_{e'_2} ... \vdash_{e'_n} \langle \ell_n, \overline{g}_n \rangle$$

where $\ell_0, ..., \ell_n \in L$, $\overline{g}_0, ..., \overline{g}_n \in \Gamma^*$ and $e'_1, ..., e'_n \in T$. We assume that, for all $i \in \{1, ..., n\}$ with $\delta_i \in \{\downarrow, \uparrow\}$, $e'_i = (\ell_{i-1}, \pi_i, \gamma_i, \ell_i)$ for some $\gamma_i \in \Gamma$ and, for all $i \in \{1, ..., n\}$ with $\delta_i = \#$, we have $e'_i = (\ell_{i-1}, \pi_i, \ell_i)$.

By definition of $\mathcal{A}'$, for every $i \in \{1, ..., n\}$, there exists an edge $e_i = (\ell_{i-1}, a_i, s_i, \phi_i, \Lambda_i, \ell_i) \in E$ such that:

– $s_i = \begin{cases} (\downarrow, \gamma_i), & \text{if } \delta_i = \downarrow, \\ \#, & \text{if } \delta_i = \#, \\ (\uparrow, \gamma_i, I_i), & \text{if } \delta_i = \uparrow \end{cases}$
  where $I_i \in \mathcal{I}$ with $\sigma_i \subseteq I_i$.
– $k_i^j \subseteq \phi_i(x^j)$ for all $j \in \{1, ..., m\}$;
– $\Lambda_i = \{x^j \mid j \in \{1, ..., m\} \text{ and } u_i^j = 1\}$;

Now we show that there exists a run ρ of $\mathcal{A}$ such that label$(\rho) = w$. We let

$$\rho = \langle \ell_0, \nu_0, r_0 \rangle \vdash_{t_1, e_1} \langle \ell_1, \nu_1, r_1 \rangle \vdash_{t_2, e_2} ... \vdash_{t_n, e_n} \langle \ell_n, \nu_n, r_n \rangle$$

where $\nu_i \in \mathbb{R}^C_{\geq 0}$ and $r_i \in \mathbb{T}\Gamma^*$ are defined as follows.

– We let $\nu_0(x^j) = 0$ for all $j \in \{1, ..., m\}$ and $r_0 = \varepsilon$.
– Assume that $i \geq 1$ and ν_{i-1} is defined. Then, by (iii), we have for all $j \in \{1, ..., m\}$: $\nu_{i-1}(x^j) + t_i \in k_i^j \subseteq \phi_i(x^j)$. Hence, $\nu_{i-1} + t_i \models \phi_i$. Then, we let $\nu_i = (\nu_{i-1} + t_i)^{\Lambda_i \leftarrow 0}$.
– Assume that $i \geq 1$ and r_{i-1} is defined. We distinguish between the following cases:
  • If $\delta_i = \downarrow$, then, we let $r_i = (\gamma_i, 0)((r_{i-1})^{+t_i})$.
  • If $\delta_i = \#$, then we let $r_i = (r_{i-1})^{+t_i}$.
  • If $\delta_i = \uparrow$, then, using (iii) and the fact that ϱ is a run of $\mathcal{A}'$, we have: $r_{i-1} = (\gamma_i, \tau')r'$ where $\tau' \in \sigma_i \subseteq I_i$. Then, we let $r_i = r'$.

Finally, since $\textsc{Mask}(\pi_1...\pi_n) \in \mathbb{L}$, we have $r_n = \varepsilon$. This shows that ρ is a run of $\mathcal{A}$. Since $\text{label}(\rho) = w$, we obtain $w \in \mathcal{L}(\mathcal{A})$. ☐

Lemma A.2. *Let $m \geq 0$, $K_1, ..., K_m, S \subseteq \mathbb{N}_+$ be finite sets and $\mathcal{A}$ a visibly pushdown automaton over $\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$. Then, there exists a timed pushdown automaton $\mathcal{A}'$ over the alphabet $\mathcal{R}(\Sigma, K_1, ..., K_m, S)$ such that $\mathcal{L}(\mathcal{A}') = \mathcal{L}(\mathcal{A}) \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S)$.*

Proof. Let $\mathcal{A} = (L, \Gamma, L_0, T, L_f)$. We put $\mathcal{A}' = (L, \Gamma, C, L_0, E, L_f)$ where $C = \{1, ..., m\}$ and $E = E^{\downarrow} \cup E^{\#} \cup E^{\uparrow}$ is defined as follows.

- For every $t = (\ell, a^{\downarrow}, \gamma, \ell') \in T^{\downarrow}$ with $a^{\downarrow} = (a, (k^1, ..., k^m), \sigma, (u^1, ..., u^m), \downarrow)$ (where $k^1 \in \mathbb{P}(K_1)$, ..., $k^m \in \mathbb{P}(K^m)$, $\sigma \in \mathbb{P}(S)$, $u^1, ..., u^m \in \{0, 1\}$), we let $(\ell, a^{\downarrow}, (\downarrow, \gamma), \phi, \Lambda, \ell') \in E^{\downarrow}$ where $\phi(j) = k^j$ for all $j \in C$ and $\Lambda = \{j \in C \mid u^j = 1\}$.
- For every $t = (\ell, a^{\#}, \ell') \in T^{\#}$ with $a^{\#} = (a, (k^1, ..., k^m), \sigma, (u^1, ..., u^m), \#)$, we let $(\ell, a^{\#}, \#, \phi, \Lambda, \ell') \in E^{\#}$ where ϕ and Λ are defined as in the previous case.
- For every $t = (\ell, a^{\uparrow}, \gamma, \ell') \in T^{\uparrow}$ with $a^{\uparrow} = (a, (k^1, ..., k^m), \sigma, (u^1, ..., u^m), \uparrow)$, we let $(\ell, a^{\uparrow}, (\uparrow, \gamma, I), \phi, \Lambda, \ell') \in E^{\uparrow}$ where ϕ and Λ are defined as in the first case, and $I = \sigma$.

Let $\Pi = \mathcal{R}(\Sigma, K_1, ..., K_m, S)$ and $\mathcal{T} = \mathcal{T}(\Sigma, K_1, ..., K_m, S)$. We prove that $\mathcal{L}(\mathcal{A}') = \mathcal{L}(\mathcal{A}) \cap \mathcal{T}$.

First, we show the inclusion $\subseteq$. Let $v = (\pi_1, t_1)...(\pi_n, t_n) \in \mathcal{L}(\mathcal{A}')$ where, for all $i \in \{1, ..., n\}$, $\pi_i = (a_i, (k_i^1, ..., k_i^m), \sigma_i, (u_i^1, ..., u_i^m), \delta_i)$. Then, there exists a run ρ of $\mathcal{A}'$ with $\text{label}(\rho) = v$. Assume that

$$\rho = \langle \ell_0, \nu_0, r_0 \rangle \vdash_{t_1, e_1} \langle \ell_1, \nu_1, r_1 \rangle \vdash_{t_2, e_2} ... \vdash_{t_n, e_n} \langle \ell_n, \nu_n, r_n \rangle.$$

where, for all $i \in \{1, ..., n\}$, $e_i = (\ell_{i-1}, \pi_i, s_i, \phi_i, \Lambda_i, \ell_i)$ with

$$s_i = \begin{cases} (\downarrow, \gamma_i), & \text{if } \delta_i = \downarrow, \\ \#, & \text{if } \delta_i = \#, \\ (\uparrow, \gamma_i, I_i), & \text{if } \delta_i = \uparrow \end{cases}$$

For all $i \in \{0, ..., n\}$, assume that $r_i = (g_i^1, \tau_i^1)...(g_i^{p_i}, \tau_i^{p_i})$ where $p_i \geq 0$, $g_i^1, ..., g_i^{p_i} \in \Gamma$ and $\tau_i^1, ..., \tau_i^{p_i} \in \mathbb{R}_{\geq 0}$. Let $\overline{g}_i = g_i^1 ... g_i^{p_i} \in \Gamma^*$.

Note that, for all $i \in \{1, ..., n\}$ with $\delta_i \in \{\downarrow, \uparrow\}$, we have $e_i' := (\ell_{i-1}, \pi_i, \gamma_i, \ell_i) \in T$ and, for all $i \in \{1, ..., n\}$ with $\delta_i = \#$, we have $e_i' := (\ell_{i-1}, \pi_i, \ell_i) \in T$. Then,

$$\langle \ell_0, \overline{g}_0 \rangle \vdash_{e_1'} \langle \ell_1, \overline{g}_1 \rangle \vdash_{e_2'} ... \vdash_{e_n'} \langle \ell_n, \overline{g}_n \rangle$$

is a run of $\mathcal{A}$ and hence $\pi_1...\pi_n \in \mathcal{L}(\mathcal{A})$. Since, for all $i \in \{1, ..., n\}$, $\phi_i(j) = k_i^j$ for all $j \in C$, $\Lambda_i = \{j \in C \mid u_i^j = 1\}$ and $\sigma_i = I_i$, it is easy to see that $v \in \mathcal{T}$. Then, $v \in \mathcal{L}(\mathcal{A}) \cap \mathcal{T}$ and hence $\mathcal{L}(\mathcal{A}') \subseteq \mathcal{L}(\mathcal{A}) \cap \mathcal{T}$.

Second, we show the inclusion $\mathcal{L}(\mathcal{A}) \cap \mathcal{T} \subseteq \mathcal{L}(\mathcal{A}')$. Let $v = (\pi_1, t_1)...(\pi_n, t_n) \in \mathcal{L}(\mathcal{A}) \cap \mathcal{T}$ such that $\pi_i = (a_i, (k_i^1, ..., k_i^m), \sigma_i, (u_i^1, ..., u_i^m), \delta_i)$ for all $i \in \{1, ..., n\}$. Then, $\pi_1...\pi_n \in \mathcal{L}(\mathcal{A})$ and hence there exists a run ϱ of $\mathcal{A}$ with $\mathrm{label}(\varrho) = v$. Assume that

$$\varrho = \langle \ell_0, \overline{g}_0 \rangle \vdash_{e_1'} \langle \ell_1, \overline{g}_1 \rangle \vdash_{e_2'} ... \vdash_{e_n'} \langle \ell_n, \overline{g}_n \rangle$$

such that:

- for all $i \in \{0, ..., n\}$: $\ell_i \in L$ and $\overline{g}_i \in \Gamma^*$ (since $v \in \mathcal{T}$, we have $\overline{g}_n = \varepsilon$)
- for all $i \in \{1, ..., n\}$ with $\delta_i \in \{\downarrow, \uparrow\}$: $e_i' = (\ell_{i-1}, \pi_i, \gamma_i, \ell_i)$ for some $\gamma_i \in \Gamma$ (note that $\gamma_i \neq \perp$, since $v \in \mathcal{T}$);
- for all $i \in \{1, ..., n\}$ with $\delta_i = \#$: $e_i' = (\ell_{i-1}, \pi_i, \ell_i)$.

We define edges $e_1, ..., e_n \in E$ as follows. For every $i \in \{1, ..., n\}$, let $e_i = (\ell_{i-1}, \pi_i, s_i, \phi_i, \Lambda_i, \ell_i)$ where:

- $s_i = \begin{cases} (\downarrow, \gamma_i), & \text{if } \delta_i = \downarrow, \\ \#, & \text{if } \delta_i = \#, \\ (\uparrow, \gamma_i, \sigma_i), & \text{if } \delta_i = \uparrow; \end{cases}$
- $\phi_i(j) = k_i^j$ for all $j \in C$;
- $\Lambda_i = \{j \in C \mid u_i^j = 1\}$.

Then, the condition $v \in \mathcal{T}$ guarantees that there exists a run ρ of $\mathcal{A}'$ of the form

$$\rho = \langle \ell_0, \nu_0, r_0 \rangle \vdash_{t_1, e_1} \langle \ell_1, \nu_1, r_1 \rangle \vdash_{t_2, e_2} ... \vdash_{t_n, e_n} \langle \ell_n, \nu_n, r_n \rangle$$

where $\nu_0, ..., \nu_n \in \mathbb{R}_{\geq 0}^C$ and $r_0, ..., r_n \in \mathbb{T}\Gamma^*$ (note that $\nu_0(j) = 0$ for all $j \in C$, $r_0 = \varepsilon$ and $\nu_1, ..., \nu_n$, $r_1, ..., r_n$ are uniquely determined by $t_1, ..., t_n$, $e_1, ..., e_n$). Since $\mathrm{label}(\rho) = v$, we have $v \in \mathcal{L}(\mathcal{A}')$. Then, $\mathcal{L}(\mathcal{A}) \cap \mathcal{T} \subseteq \mathcal{L}(\mathcal{A}')$. □

B Definability Implies Recognizability

In this section, we present a full proof of Lemma 6.2.

Recall that $\psi = \exists^{\mathrm{match}}\mu.\exists D_1.\ ...\ \exists D_m.\varphi \in \mathrm{TML}(\Sigma)$ where $m \geq 0$, $D_1, ..., D_m \in \mathcal{D}$ are pairwise distinct variables and, whenever $\mathrm{d}^\kappa(D, x)$ or $\mu^\kappa(x, y)$ is a subformula of φ, then $\kappa \in \{\leq, \geq\} \times \mathbb{N}$.

Lemma B.1. *There exist finite sets $K_1, ..., K_m, S \subseteq \mathbb{N}_+$, a renaming $h : \mathcal{R}(\Sigma, K_1, ..., K_m, S) \to \Sigma$ and a sentence $\varphi^* \in \mathrm{MSO}_{\mathbb{L}}(\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S))$ such that $\mathcal{L}(\psi) = h(\mathcal{L}(\varphi^*) \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S))$.*

Proof. The idea here is the following. First, we encode in φ the values of the variables $D_1, ..., D_m, \mu$ and the time constraints appearing in subformulas of the form $\mathrm{d}^\kappa(D, x)$ and $\mu^\kappa(x, y)$ as an input word over the visibly pushdown alphabet $\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$. Then, all time constraints and the correctness of a matching relation are checked by the intersection of the visibly pushdown language $\mathcal{L}(\varphi^*)$ with the timed language $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$. Finally, we remove the auxiliary components of the extended alphabet by applying the projection h.

For every $j \in \{1, ..., m\}$, let K_j be the set of all $n \in \mathbb{N}_+$ such that $\mathrm{d}^{\bowtie n}(D_j, x)$ is a subformula of φ for some $x \in \mathcal{V}_1$ and $\bowtie \in \{\leq, \geq\}$. Let S be the set of all $n \in \mathbb{N}_+$ such that $\mu^{\bowtie n}(x, y)$ is a subformula of φ for some $x, y \in \mathcal{V}_1$ and $\bowtie \in \{\leq, \geq\}$. For the simplicity, let $\mathcal{R} = \mathcal{R}(\Sigma, K_1, ..., K_m, S)$, $\tilde{\mathcal{R}} = \tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S)$ and $\mathcal{T} = \mathcal{T}(\Sigma, K_1, ..., K_m, S)$. Let $\Omega = \mathbb{P}(K_1) \times ... \times \mathbb{P}(K_m)$ and $\Delta = \{\downarrow, \#, \uparrow\}$. We will denote a letter of $\mathcal{R}$ by $(a, \mathcal{K}, s, \mathcal{G}, \delta)$ where $a \in \Sigma$, $\mathcal{K} \in \Omega$, $s \in \mathbb{P}(S)$, $\mathcal{G} \in \{0,1\}^m$ and $\delta \in \Delta$. Let $h : \mathcal{R} \to \Sigma$ be the projection to the Σ-component. Finally, we define the formula $\varphi^* \in \mathrm{MSO}_{\mathbb{L}}(\tilde{R})$ from the formula φ by the following substitutions.

- If $P_a(x)$ with $a \in \Sigma$ and $x \in \mathcal{V}_1$ is a subformula of φ, then $P_a(x)$ is replaced by the formula $\bigvee(P_{(a,\mathcal{K},s,\mathcal{G},\delta)}(x) \mid \mathcal{K} \in \Omega, s \in \mathbb{P}(S), \mathcal{G} \in \{0,1\}^m, \delta \in \Delta)$.
- If $D_j(x)$ with $j \in \{1, ..., m\}$ and $x \in \mathcal{V}_1$ appears in φ, then $D_j(x)$ is replaced by the formula $\bigvee(P_{(a,\mathcal{K},s,\mathcal{G},\delta)}(x) \mid a \in \Sigma, \mathcal{K} \in \Omega, s \in \mathbb{P}(S)$, $\mathcal{G} = (g_1, ..., g_m) \in \{0,1\}^m$ with $g_j = 1, \delta \in \Delta)$.
- If $\mathrm{d}^I(D_j, x)$ is a subformula of φ where $j \in \{1, ..., m\}$, $x \in \mathcal{V}_1$ and $I \in \mathcal{I}$ is an interval of the form $[0, n]$ or $[n, \infty)$ for some $n \in \mathbb{N}$, then we replace $\mathrm{d}^I(D_j, x)$ by the formula $\bigvee(P_{(a,\mathcal{K},s,\mathcal{G},\delta)}(x) \mid a \in \Sigma,\ \mathcal{K} = (k_1, ..., k_m) \in \Omega$ with $k_j \subseteq I, s \in \mathbb{P}(S), \mathcal{G} \in \{0,1\}^m, \delta \in \Delta)$. Note that we remove the variable D_j from φ.
- If $\mu^I(x, y)$ is a subformula of φ where $x, y \in \mathcal{V}_1$ and $I \in \mathcal{I}$ is an interval of the form $[0, n]$ or $[n, \infty)$ for some $n \in \mathbb{N}$, then $\mu^I(x, y)$ is replaced by the formula $\mathbb{L}(x, y) \wedge \bigvee(P_{(a,\mathcal{K},s,\mathcal{G},\uparrow)}(y) \mid a \in \Sigma, \mathcal{K} \in \Omega, s \in \mathbb{P}(S)$ with $s \subseteq I, \mathcal{G} \in \{0,1\}^m)$. Note that here we replace the matching relation μ by the matching relation $\mathbb{L}$ with respect to the visibly pushdown alphabet $\tilde{\mathcal{R}}$ and measure the time distance using the $\mathbb{P}(S)$-component of the extended alphabet.

Note that φ^* is a sentence, since $\mathrm{Free}(\varphi) \subseteq \{D_1, ..., D_m, \mu\}$ and we removed $D_1, ..., D_m, \mu$ when constructing φ^*. Using the same substitutions, we transform every subformula η of φ into the formula $\eta^* \in \mathrm{MSO}_{\mathbb{L}}(\tilde{R})$.

It remains to show that $\mathcal{L}(\psi) = h(\mathcal{L}(\varphi^*) \cap \mathcal{T})$. First, we show the inclusion $\subseteq$. Let $w = (a_1, t_1)...(a_n, t_n) \in \mathcal{L}(\psi)$ and σ be a fixed w-assignment. Then, there exist a matching $M \subseteq \mathrm{dom}(w) \times \mathrm{dom}(w)$ and sets $I_1, ..., I_m \subseteq \mathrm{dom}(w)$ such that $(w, \sigma[D_1/I_1, ..., D_m/I_m, \mu/M]) \models \varphi$. We construct a word $w^* = r_1...r_n \in \mathcal{R}^+$ where, for all $i \in \{1, ..., n\}$, $r_i = (a_i, (k_i^1, ..., k_i^m), s_i, (g_i^1, ..., g_i^m), \delta_i)$ with $a_i \in \Sigma$, $(k_i^1, ..., k_i^m) \in \Omega$, $s_i \in \mathbb{P}(S)$, $g_i^1, ..., g_i^m \in \{0,1\}$, $\delta_i \in \Delta$ as follows.

- For all $i, i' \in \{1, ..., n\}$ with $(i, i') \in M$, we put $\delta_i = \downarrow$, $\delta_{i'} = \uparrow$, $s_i = [0, 0]$ (in principle, s_i can be chosen arbitrarily) and let $s_{i'}$ be the interval in the partition $\mathbb{P}(S)$ with $\langle w \rangle_{i,i'} \in s_{i'}$ (uniquely determined).
- For all $i \in \{1, ..., n\}$ such that i does not belong to any pair in M, we put $\delta_i = \#$ and $s_i = [0, 0]$ (again, s_i can be chosen arbitrarily).
- For all $i \in \{1, ..., n\}$ and $j \in \{1, ..., m\}$, we let $g_i^j = 1$ iff $i \in I_j$.
- For all $i \in \{1, ..., n\}$ and $j \in \{1, ..., m\}$, we let k_i^j be the interval in the partition $\mathbb{P}(K_j)$ such that $(I_j, i) \in \mathrm{d}^{k_i^j}(w)$ (note that k_i^j is uniquely determined).

Let $w^{**} := (r_1, t_1)...(r_n, t_n) \in \mathbb{T}\mathcal{R}^+$. Clearly, $w^{**} \in \mathcal{T}$ and $h(w^{**}) = w$.

For a w-assignment $\sigma : \mathcal{U} \to \mathrm{dom}(w) \cup 2^{\mathrm{dom}(w)} \cup 2^{(\mathrm{dom}(w))^2}$, let $\overline{\sigma} = \sigma[D_1/I_1, ..., D_m/I_m, \mu/M]$ and $\overline{\overline{\sigma}}$ be the restriction of σ to $\mathcal{V}_1 \cup \mathcal{V}_2$, i.e., $\overline{\overline{\sigma}} = \sigma|_{\mathcal{V}_1 \cup \mathcal{V}_2}$.

Now we show that $w^* \in \mathcal{L}(\varphi^*)$. For this, we show that, for every w-assignment σ and every subformula η of φ, the following holds:

$$(w, \overline{\sigma}) \models \eta \ \text{ iff } \ (w^*, \overline{\overline{\sigma}}) \models \eta^*. \tag{1}$$

We proceed by induction on the structure of η.

(i) The cases $\eta = P_a(x)$, $\eta = (x \leq y)$ and $\eta = X(x)$ with $a \in \Sigma$, $x, y \in \mathcal{V}_1$ and $X \in \mathcal{V}_2$ are straightforward.

(ii) Let $\eta = D_j(x)$ with $j \in \{1, ..., m\}$ and $x \in \mathcal{V}_1$. Then:

$$(w, \overline{\sigma}) \models \eta \ \Leftrightarrow \ \sigma(x) \in I_j \ \Leftrightarrow \ g^j_{\sigma(x)} = 1 \ \Leftrightarrow \ (w^*, \overline{\overline{\sigma}}) \models \eta^*$$

(iii) Let $\eta = \mathrm{d}^J(D_j, x)$ where $j \in \{1, ..., m\}$, $x \in \mathcal{V}_1$ and J is an interval of the form $[0, k]$ or $[k, \infty)$ for some $k \in K_j \cup \{0\}$. Then:

$$(w, \overline{\sigma}) \models \eta \ \Leftrightarrow \ k^j_{\sigma(x)} \subseteq J \ \Leftrightarrow \ (w^*, \overline{\overline{\sigma}}) \models \eta^*.$$

(iv) Let $\eta = \mu^J(x, y)$ where $x, y \in \mathcal{V}_1$ and J is an interval of the form $[0, s]$ or $[s, \infty)$ for some $s \in S \cup \{0\}$. Then:

$$\begin{aligned}(w, \overline{\sigma}) \models \eta \ \Leftrightarrow \ (&\sigma(x) < \sigma(y) \wedge \delta_{\sigma(x)} = \downarrow \wedge\, \delta_{\sigma(y)} = \uparrow \\ &\wedge \textsc{Mask}(r_{\sigma(x)+1}...r_{\sigma(y)-1}) \in \mathbb{L} \wedge s_{\sigma(y)} \subseteq J) \ \Leftrightarrow \ (w, \overline{\overline{\sigma}}) \models \eta^*\end{aligned}$$

(v) Let $\eta = \eta_1 \vee \eta_2$. By induction hypothesis, $(w, \overline{\sigma}) \models \eta_i \ \Leftrightarrow \ (w^*, \overline{\overline{\sigma}}) \models (\eta_i)^*$ for $i \in \{1, 2\}$. Then:

$$\begin{aligned}(w, \overline{\sigma}) \models \eta \ &\Leftrightarrow \ (w, \overline{\sigma}) \models \eta_1 \text{ or } (w, \overline{\sigma}) \models \eta_2 \\ &\overset{(!)}{\Leftrightarrow} \ (w^*, \overline{\overline{\sigma}}) \models (\eta_1)^* \text{ or } (w^*, \overline{\overline{\sigma}}) \models (\eta_2)^* \\ &\Leftrightarrow \ (w^*, \overline{\overline{\sigma}}) \models (\eta_1)^* \vee (\eta_2)^* \models (w^*, \overline{\overline{\sigma}}) \models \eta^*\end{aligned}$$

Here, at the place (!), we apply induction hypothesis.

(vi) Let $\eta = \neg\eta'$. By induction hypothesis, $(w, \overline{\sigma}) \models \eta' \ \Leftrightarrow \ (w^*, \overline{\overline{\sigma}}) \models (\eta')^*$. Then:

$$(w, \overline{\sigma}) \models \eta \ \Leftrightarrow \ (w, \overline{\sigma}) \not\models \eta' \ \Leftrightarrow \ (w^*, \overline{\overline{\sigma}}) \not\models (\eta')^* \ \Leftrightarrow \ (w^*, \overline{\overline{\sigma}}) \models \eta.$$

Here, at the place (!), we apply induction hypothesis.

(vii) Let $\eta = \exists x.\eta'$ with $x \in \mathcal{V}_1$. For every $i \in \{1, ..., n\}$, by induction hypothesis, we have $(w, \overline{\sigma}[x/i]) \models \eta' \ \Leftrightarrow \ (w, \overline{\overline{\sigma}}[x/i]) \models (\eta')^*$. Then, since $\eta^* = \exists x.(\eta')^*$, (1) holds true.

(viii) The case $\eta = \exists X.\eta'$ with $X \in \mathcal{V}_2$ is similar to the previous case.

Since φ^* is a sentence and $(w, \overline{\sigma}) \models \varphi$ for any w-assignment σ, it follows from (1) that $w^* \models \varphi^*$ and hence $w^* \in \mathcal{L}(\varphi^*)$. Thus, $w \in h(\mathcal{L}(\varphi^*) \cap \mathcal{T})$.

Second, we show the converse inclusion $\mathcal{L}(\psi) \supseteq h(\mathcal{L}(\varphi^*) \cap \mathcal{T})$. Let $w = (a_1, t_1)...(a_n, t_n) \in h(\mathcal{L}(\varphi^*) \cap \mathcal{T})$. Then, there exists a timed word $w^{**} = (r_1, t_1)...(r_n, t_n) \in \mathcal{L}(\varphi^*) \cap \mathcal{T}$ such that $h(w^{**}) = w$. Assume that, for all $i \in \{1, ..., n\}$, $r_i = (a_i, (k_i^1, ..., k_i^m), s_i, (g_i^1, ..., g_i^m), \delta_i)$. Let $M = \{(i,j) \in (\mathrm{dom}(w))^2 \mid i < j, \delta_i = \downarrow, \delta_j = \uparrow \text{ and } \textsc{Mask}(r_{i+1}...r_{j-1}) \in \mathbb{L}\}$. Since $w^{**} \in \mathcal{T}$, M is a matching. For every $j \in \{1, ..., m\}$, let $I_j = \{i \in \mathrm{dom}(w) \mid g_i^j = 1\}$. Let σ be a w-assignment. To prove that $w \in \mathcal{L}(\psi)$, we show that $(w, \sigma[D_1/I_1, ..., D_m/I_m, \mu/M]) \models \varphi$. As in the proof of the inclusion $\subseteq$, for any w-assignment $\sigma : \mathcal{U} \to \mathrm{dom}(w) \cup 2^{\mathrm{dom}(w)} \cup 2^{(\mathrm{dom}(w))^2}$, let $\overline{\sigma} = \sigma[D_1/I_1, ..., D_m/I_m, \mu/M]$ and $\overline{\overline{\sigma}}$ be the restriction of σ to $\mathcal{V}_1 \cup \mathcal{V}_2$. Let $w^* = r_1...r_n \in \mathcal{L}(\varphi^*)$. We show that, for every w-assignment σ and every subformula η of φ, (1) holds. We proceed by induction on the structure of η in the same fashion as it was done in (i) - (viii) with the only difference that we apply the fact that $w^{**} \in \mathcal{T}$. Then, $(w, \sigma[D_1/I_1, ..., D_m/I_m, \mu/M]) \models \varphi$ and hence $w \models \psi$. This finishes the proof of this lemma. $\square$

C Recognizability Implies Definability

We present a full proof of Lemma 6.4.

Lemma C.1. *Let Σ be an alphabet, $m \geq 0$, $K_1, ..., K_m, S \subseteq \mathbb{N}_+$ finite sets, $h : \mathcal{R}(\Sigma, K_1, ..., K_m, S) \to \Sigma$ a renaming, and $\varphi \in \textsc{Mso}_{\mathbb{L}}(\tilde{\mathcal{R}}(\Sigma, K_1, ..., K_m, S))$ a sentence. Then, there exists a sentence $\psi \in \textsc{Tml}(\Sigma)$ such that $\mathcal{L}(\psi) = h(\mathcal{L}(\varphi) \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S))$.*

Proof. For the proof, we follow a similar approach as in [8], Theorem 6.6. Let $\Gamma = \mathcal{R}(\Sigma, K_1, ..., K_m, S)$, $\overline{X} = \{X_\gamma \in \mathcal{V}_2 \mid \gamma \in \Gamma\}$ be a set of pairwise distinct variables not appearing in φ, and $\overline{D} = \{D_i \in \mathcal{D} \mid 1 \leq i \leq m\}$ be a set of pairwise distinct relative distance variables. Using the $\overline{X}$-variables, we will describe the renaming h (i.e., we store in these second-order variables the positions of the letters of the extended alphabet Γ before the renaming). We will use the $\overline{D}$-variables as well as the matching variable μ to describe the timed language $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$. Moreover, we transform φ to a $\textsc{tMso}(\Sigma)$-formula.

Let $\varphi^* \in \textsc{tMso}(\Sigma)$ be the formula obtained from φ by the following transformations.

- Every subformula $\mathbb{L}(x, y)$ of φ with $x, y \in \mathcal{V}_1$ is replaced by $\mu(x, y)$.
- Every subformula $P_\gamma(x)$ of φ with $x \in \mathcal{V}_1$ and $\gamma \in \Gamma$ is replaced by the formula $P_{h(\gamma)}(x) \wedge X_\gamma(x)$.

The formula

$$\textsc{Partition} = \left(\forall x. \left[\bigvee_{\gamma \in \Gamma} \left(X_\gamma(x) \wedge \bigwedge_{\gamma' \neq \gamma} \neg X_{\gamma'}(x)\right)\right]\right)$$

demands that values of $\overline{X}$-variables form a partition of the domain. The formula

$$\textsc{Renaming} = \forall x. \Big(\bigvee_{\gamma \in \Gamma} \big(X_\gamma(x) \to P_{h(\gamma)}(x) \big) \Big)$$

correlates values of $\overline{X}$-variables with an input word. It remains to handle $\mathcal{T}(\Sigma, K_1, ..., K_m, S)$. For a letter $\gamma = (a, (k_1, ..., k_m), s, (g_1, ..., g_m), \delta) \in \Gamma$ with $a \in \Sigma$, $k_1 \in \mathbb{P}(K_1)$, ..., $k_m \in \mathbb{P}(K_m)$, $s \in \mathbb{P}(S)$, $g_1, ..., g_m \in \{0, 1\}$ and $\delta \in \{\downarrow, \#, \uparrow\}$, we write $k_j(\gamma) = k_j$ $(1 \le j \le m)$, $s(\gamma) = s$, $g_j(\gamma) = g_j$ $(1 \le j < m)$ and $\delta(\gamma) = \delta$. For $x \in \mathcal{V}_1$ and $I \in \mathcal{I}$ and $\delta \in \{\downarrow, \#, \uparrow\}$, let the formula $M^I_\delta(x)$ be defined as follows.

- If $\delta = \downarrow$, then $M^I_\delta(x) = \exists y. \big(\mu(x, y) \wedge \bigvee_{\substack{\gamma \in \Gamma, \\ \delta(\gamma) = \uparrow}} X_\gamma(y) \big)$.
- If $\delta = \#$, then $M^I_\delta(x) = \neg \exists y. (\mu(x, y) \vee \mu(y, x))$.
- If $\delta = \uparrow$, then $M^I_\delta(x) = \exists y. \big(\mu^I(y, x) \wedge \bigvee_{\substack{\gamma \in \Gamma, \\ \delta(\gamma) = \downarrow}} X_\gamma(y) \big)$.

For $j \in \{1, ..., m\}$ and $x \in \mathcal{V}_1$, let $D^1_j(x) = D_j(x)$ and $D^0_j(x) = \neg D_j(x)$. Consider the $\textsc{tMso}(\Sigma)$-formula

$$\xi = \forall x. \bigwedge_{\gamma \in \Gamma} X_\gamma(x) \to \Big(\bigwedge_{j=1}^{m} (\mathrm{d}^{k_j(\gamma)}(D_j, x) \wedge D_j^{g_j(\gamma)}(x)) \wedge M^{s(\gamma)}_{\delta(\gamma)}(x) \Big)$$

which takes care of matchings and time. Let $(\gamma_i)_{i \in \{1, ..., |\Gamma|\}}$ be an enumeration of Γ. Then, we define the desired sentence ψ as

$$\psi = \exists^{\mathrm{match}} \mu. \exists D_1. \; ... \; \exists D_m. \exists X_1. \; ... \; \exists X_{|\Gamma|}. (\varphi^* \wedge \textsc{Partition} \wedge \textsc{Renaming} \wedge \xi).$$

We show that $\mathcal{L}(\psi) = h(\mathcal{L}(\varphi) \cap \mathcal{T}(\Sigma, K_1, ..., K_m, S))$.

Again, for the sake of simplicity, let $\mathcal{T} = \mathcal{T}(\Sigma, K_1, ..., K_m, S)$. First, we show that $\mathcal{L}(\psi) \subseteq h(\mathcal{L}(\varphi) \cap \mathcal{T})$. Let $w = (a_1, t_1)...(a_n, t_n) \in \mathcal{L}(\psi)$ and σ be a fixed w-assignment. Then, there exist sets $I_1, ..., I_m, J_1, ..., J_{|\Gamma|} \subseteq \mathrm{dom}(w)$ and a matching relation $M \subseteq (\mathrm{dom}(w))^2$ such that:

(i) the sets $J_1, ..., J_{|\Gamma|}$ form a partition of $\mathrm{dom}(w)$;
(ii) for all $i \in \mathrm{dom}(w)$: whenever $i \in J_v$ for some $v \in \{1, ..., |\Gamma|\}$, we have $a_i = h(\gamma_v)$;
(iii) $(w, \sigma[X_1/J_1, ..., X_{|\Gamma|}/J_{|\Gamma|}, \mu/M]) \models \varphi^*$;
(iv) $(w, \sigma[D_1/I_1, ..., D_m/I_m, X_1/J_1, ..., X_{|\Gamma|}/J_{|\Gamma|}, \mu/M]) \models \xi$; moreover, note that the sets $I_1, ..., I_m$ are uniquely determined by the sets $J_1, ..., J_{|\Gamma|}$.

For each $i \in \{1, ..., |\Gamma|\}$, let $\pi_i = \gamma_j$ for $j \in \{1, ..., |\Gamma|\}$ such that $i \in J_j$ (here we take into account (i)). Then, by (iii), we have $\pi_1 ... \pi_n \in \mathcal{L}(\varphi)$. Let $u = (\pi_1, t_1)...(\pi_n, t_n)$. Then, by (iv), $u \in \mathcal{T}$ and, by (ii), $h(u) = w$. Hence, $w \in h(\mathcal{L}(\varphi) \cap \mathcal{T})$.

Second, we show that $h(\mathcal{L}(\varphi) \cap \mathcal{T}) \subseteq \mathcal{L}(\psi)$. Assume that $w = (a_1, t_1)...(a_n, t_n) \in h(\mathcal{L}(\varphi) \cap \mathcal{T})$. Then, there exist $\pi_1, ..., \pi_n \in \Gamma$ such that:

(a) $\pi_1 ... \pi_n \in \mathcal{L}(\varphi)$;
(b) $v := (\pi_1, t_1)...(\pi_n, t_n) \in \mathcal{T}$;
(c) $h(\pi_i) = a_i$ for all $i \in \{1, ..., n\}$.

Assume that, for all $i \in \{1, ..., n\}$, $\pi_i = (a_i, (k_i^1, ..., k_i^m), s_i, (g_i^1, ..., g_i^m), \delta_i)$. For each $j \in \{1, ..., m\}$, let $I_j = \{i \in \{1, ..., n\} \mid g_i^j = 1\}$. For each $v \in \{1, ..., |\Gamma|\}$, let $J_v = \{i \in \{1, ..., n\} \mid \pi_i = \gamma_v\}$. Then, clearly, the sets $J_1, ..., J_{|\Gamma|}$ form a partition of $\mathrm{dom}(w)$. We define a relation $M \subseteq (\mathrm{dom}(w))^2$ as follows:

$$M = \{(i, i') \mid i < i', \delta_i = \downarrow, \delta_{i'} = \uparrow \text{ and } \textsc{Mask}(\pi_{i+1} ... \pi_{i'-1}) \in \mathbb{L}\}.$$

By (b), M is a matching relation. Note also that M is uniquely determined by the sets $J_1, ..., J_{|\Gamma|}$.

Let σ be a fixed w-assignment. Then, the following holds.

- Since the sets $J_1, ..., J_{|\Gamma|}$ form a partition of $\mathrm{dom}(w)$, we have $(w, \sigma[X_1/J_1, ..., X_{|\Gamma|}/J_{|\Gamma|}]) \models \textsc{Partition}$.
- By (c), we have $(w, \sigma[X_1/J_1, ..., X_{|\Gamma|}/J_{|\Gamma|}]) \models \textsc{Renaming}$.
- Using (a), it can be easily checked that $(w, \sigma[X_1/J_1, ..., X_{|\Gamma|}/J_{|\Gamma|}, \mu/M]) \models \varphi^*$.
- By (b), we have $(w, \sigma[D_1/I_1, ..., D_m/I_m, X_1/J_1, ..., X_{|\Gamma|}/J_{|\Gamma|}, \mu/M]) \models \xi$.

Then, $(w, \sigma[D_1/I_1, ..., D_m/I_m, X_1/J_1, ..., X_{|\Gamma|}/J_{|\Gamma|}, \mu/M]) \models \varphi^* \wedge \textsc{Partition} \wedge \textsc{Renaming} \wedge \xi$ and hence $w \in \mathcal{L}(\psi)$. This finishes the proof of this theorem. $\square$