A Logical Characterization for Weighted Timed Automata

Karin Quaas

Institut für Informatik, Universität Leipzig
04009 Leipzig, Germany
quaas@informatik.uni-leipzig.de

May 27, 2010

Abstract. We aim to generalize Büchi's fundamental theorem on the coincidence of recognizable and MSO-definable languages to a weighted timed setting. For this, we investigate weighted timed automata and show how we can extend Wilke's relative distance logic with weights taken from an arbitrary semiring. We show that every formula in our logic can effectively be transformed into a weighted timed automaton, and vice versa. The results indicate the robustness of weighted timed automata and may also be used for specification purposes.

Introduction

Recently, *weighted timed automata* [2, 4] have received much attention in the real-time community. Weighted timed automata are timed automata that allow us to assign weights to both the locations and the edges. In this way, they can be used to model continuous consumption of resources [7, 6]. The goal of this paper is to define a monadic second-order (MSO) logic to specify the behaviours of weighted timed automata logically. For finite automata over words, this was first done by Büchi [10]. He showed that MSO is expressively equivalent to finite automata. This theorem is of great practical interest: a specification expressed by a MSO formula is often much easier to read and understand than an automaton. The most important questions that arise in the context of specification, e.g. the satisfiability problem or the model checking problem, can be solved using methods from automata theory owing to Büchi's theorem. In this paper, we aim to generalize Büchi's theorem to weighted timed automata.

Our work is also motivated by recent works on *weighted logics* by Droste and Gastin [12, 13]. The authors introduce a *weighted* MSO logic for characterizing the behaviours of weighted finite automata defined over a semiring. In weighted MSO logic, atomic formulas my additionally comprise elements of the semiring, which may be used to define the weight of a transition in a weighted finite automaton. Since full weighted MSO logic is expressively stronger than weighted finite automata, the authors consider a syntactically restricted fragment. They prove that this fragment is expressively equivalent to weighted finite automata. Recently, this result has been generalized to weighted settings of infinite words [15], trees [16], pictures [21], traces [22], texts [19] and nested words [20].

Here, we aim to generalize the result to timed series. The basis of our work is a Büchi theorem for the class of timed automata by Wilke [29]. For this result, Wilke introduces a timed extension of classical MSO logic. The intuitive idea is to extend $\mathsf{MSO}(\Sigma)$ with formulas of the form $d(y, z) \sim c$, called *distance predicates*, where y, z are first-order variables, $\sim \in \{<, \leq, =, \geq, >\}$ and $c \in \mathbb{N}$. A formula of this form, interpreted over timed words, is supposed to express that the time distance between the positions y and z satisfies the constraint $\sim c$. However, the logic that allows for the unrestricted use of distance predicates is expressively stronger than timed automata [25, 29]. For this reason, Wilke restricts the use of distance predicates. The resulting logic is known as *relative distance logic*. Wilke [29] shows that the relative distance logic is expressively equivalent to timed automata.

Following the approach of Droste and Gastin, we extend Wilke's relative distance logic with weighted formulas. In contrast to weighted MSO, we not only allow elements from the semiring as atomic formulas, but also functions that, intuitively, may be used to express the weight that arises while being in a location of a weighted timed automaton. We then define a fragment of our logic and prove that this fragment is expressively equivalent to weighted timed automata.

For this, we use parts of the proofs presented by Droste and Gastin [13]. However, in the weighted timed setting we are faced with two new problems caused by the weights that are assigned to locations. First, the Hadamard product, which is used for defining the semantics of conjunction in our logic, does not preserve recognizability. Second, Droste and Gastin's restriction on the application of the universal first-order quantifier is too strict in the setting of timed systems. To overcome these problems, we define new syntactical restrictions for these operators.

Besides these problems, the construction for showing closure of recognizable series under application of first-order universal quantification given by Droste and Gastin strongly relies on the fact that every finite automaton can be determinized. However, it is well known that timed automata in general cannot be determinized [1]. We present two different approaches for solving this problem. First, we consider *idempotent* semirings. For this class of semirings the determinizability of weighted timed automata is not needed in the proof of closure under first-order universal quantification. Idempotent semirings comprise amongst others the so called min-plus-semiring over the positive real numbers, which is widely used in the context of weighted timed automata [4, 2, 6]. Second, instead of restricting the discussion to idempotent semirings, we restrict the syntax of our logic. In particular, we will restrict the application of the first-order universal quantifier to formulas that define a special subclass of timed languages whose elements have a *bounded variability*. This notion was introduced by Wilke [29]. Intuitively, the variability of a timed word corresponds to the maximum number of events that may occur within one unit time. Then we take advantage of the fact that every timed language with a bounded variability is *deterministically* recognizable.

Related Work Recently [7, 9, 8, 11], weighted timed extensions of temporal logics like LTL and CTL have been introduced for the logical analysis of weighted timed automata. In these logics, the temporal operators are extended with time and weight constraints. In this way, one can express that the weight in a path is not allowed to exceed a certain value. The crucial difference to the logic presented in this article is that the formulas in these logics only take Boolean values.

For weighted event-clock automata, a strict subclass of weighted timed automata, we recently [24] presented a weighted timed MSO logic and a Büchi theorem. For this, an MSO logic introduced by D'Souza [17] for characterizing the behaviour of event-clock automata is extended with weights. Since event-clock automata - as opposed to timed automata - can always be determinized, some of the proofs are simpler than the corresponding ones presented in the present article. In particular, we do not have to restrict the application of the universal quantifier to timed languages with bounded variability.

In the literature, one can find weighted MSO logics for other types of languages that are also not deterministically recognizable. In the setting of picture languages, Mäurer [21] introduces *first-order step functions* (rather than *recognizable step functions* as in [13]), and exploits the fact that every first-order definable picture language can be recognized by an *unambiguous* (rather than *deterministic*) picture automaton. The same approach is followed by Bollig and Meinecke [5] for Mazurkiewicz traces running over directed acyclic graphs.

1 (Weighted) Timed Automata

Let Σ and Γ denote alphabets, and let $\mathbb{N}$ and $\mathbb{R}_{\geq 0}$ denote the natural numbers and the positive reals, respectively. A *timed word* is a non-empty finite sequence $(a_1, t_1)...(a_k, t_k) \in (\Sigma \times \mathbb{R}_{\geq 0})^+$ such that the sequence $\bar{t} = t_1...t_k$ of timestamps is non-decreasing[1]. Sometimes we denote a timed word as above by $(\bar{a}, \bar{t})$, where $\bar{a} \in \Sigma^+$. We write $T\Sigma^+$ for the set of timed words over Σ. A set $L \subseteq T\Sigma^+$ is called a *timed language*. We let the domain $\mathsf{dom}(w)$ of w be $\{1, ..., k\}$ and define the length $|w|$ of w to be k. Let $\pi : \Sigma \to \Gamma$ be a mapping. The *renaming* $\pi(w)$ of a timed word $w \in T\Sigma^+$ is the timed word $w' \in T\Gamma^+$ of the form $(a'_1, t'_1)...(a'_k, t'_k)$ such that $a'_i = \pi(a_i)$ and $t'_i = t_i$ for all $i \in \mathsf{dom}(w)$.

Let $\mathcal{C}$ be a finite set of *clock variables* ranging over $\mathbb{R}_{\geq 0}$. We define *clock constraints* ϕ over $\mathcal{C}$ to be conjunctions of formulas of the form $x \sim c$, where $x \in \mathcal{C}$, $c \in \mathbb{N}$, and $\sim \in \{<, \leq, =, \geq, >\}$. We use $\Phi(\mathcal{C})$ to denote the set of all clock constraints over $\mathcal{C}$. A *clock valuation* ν is a function from $\mathcal{C}$ to $\mathbb{R}_{\geq 0}$ mapping each clock variable to its current value. We let ν_0 be a special clock valuation assigning 0 to each clock variable. A clock valuation ν satisfies a clock constraint ϕ, written $\nu \models \phi$, if ϕ evaluates to true according to the values given by ν. For $\delta \in \mathbb{R}_{\geq 0}$ and $\lambda \subseteq \mathcal{C}$, respectively, we define $\nu + \delta$ to be $(\nu + \delta)(x) = \nu(x) + \delta$ for each $x \in \mathcal{C}$

[1] We assume a timed word to be non-empty for technical simplicity, see Remark 24.

and $\nu[\lambda := 0]$ by $\nu[\lambda := 0](x) = 0$ if $x \in \lambda$ and $\nu[\lambda := 0](x) = \nu(x)$ otherwise, respectively. A *timed automaton* over Σ is a tuple $\mathcal{A} = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E)$, where

- $\mathcal{L}$ is a finite set of locations (states),
- $\mathcal{L}_0 \subseteq \mathcal{L}$ is a set of initial locations,
- $\mathcal{L}_f \subseteq \mathcal{L}$ is a set of final locations,
- $\mathcal{C}$ is a finite set of clock variables,
- $E \subseteq \mathcal{L} \times \Sigma \times \Phi(\mathcal{C}) \times 2^{\mathcal{C}} \times \mathcal{L}$ is a finite set of edges. An edge $(l, a, \phi, \lambda, l')$ allows a jump from location l to location l' if a is read, provided that for the current valuation ν we have $\nu \models \phi$. After the edge has been executed, the new valuation is $\nu[\lambda := 0]$.

In timed automata, we distinguish between *timed* and *discrete* transitions. A timed transition is of the form $(l, \nu) \xrightarrow{\delta} (l, \nu + \delta)$ for some $\delta \in \mathbb{R}_{\geq 0}$ and represents the elapse of time δ in l, whereas a discrete transition is of the form $(l, \nu) \xrightarrow{e} (l', \nu')$ for some $e = (l, a, \phi, \lambda, l') \in E$ such that $\nu \models \phi$ and $\nu' = \nu[\lambda := 0]$. A *transition* is a timed transition followed by a discrete transition, written $(l, \nu) \xrightarrow{\delta, e} (l', \nu')$. A *run* of $\mathcal{A}$ on a timed word w is a finite sequence $(l_0, \nu_0) \xrightarrow{\delta_1, e_1} (l_1, \nu_1) \xrightarrow{\delta_2, e_2} \ldots \xrightarrow{\delta_{|w|}, e_{|w|}} (l_{|w|}, \nu_{|w|})$ of transitions. We say that a run r is *successful* if $l_0 \in \mathcal{L}_0$ and $l_{|w|} \in \mathcal{L}_f$. The timed language $L(\mathcal{A})$ recognized by a timed automaton $\mathcal{A}$ is defined by $L(\mathcal{A}) = \{w \in T\Sigma^+ :$ there is a successful run of $\mathcal{A}$ on $w\}$. A timed language $L \subseteq T\Sigma^+$ is said to be *TA-recognizable* over Σ, if there is a timed automaton $\mathcal{A}$ over Σ such that $L(\mathcal{A}) = L$. A timed automaton $\mathcal{A}$ is *deterministic* if $|\mathcal{L}_0| = 1$, and whenever $(l, a, \phi_1, \lambda_1, l_1)$ and $(l, a, \phi_2, \lambda_2, l_2)$ are two different edges in $\mathcal{A}$, then for all clock valuations ν we have $\nu \not\models \phi_1 \wedge \phi_2$. A timed language $L \subseteq T\Sigma^+$ is *deterministically TA-recognizable*, if there is a deterministic timed automaton $\mathcal{A}$ over Σ such that $L(\mathcal{A}) = L$.

Proposition 1 ([1, 3]) *The class of TA-recognizable timed languages is closed under union, intersection, renaming and inverse renaming.*

The proof that intersection preserves TA-recognizability of timed languages involves the usual product construction known from the classical theory of formal languages. Contrary to the classical case, the class of deterministically TA-recognizable timed languages forms a strict subclass of TA-recognizable timed languages [1].

We extend timed automata to be equipped with weights taken from a semiring. For this, we let $\mathcal{K}$ be a *semiring*, i.e., an algebraic structure $\mathcal{K} = (K, +, \cdot, 0, 1)$ such that $(K, +, 0)$ is a commutative monoid, $(K, \cdot, 1)$ is a monoid, multiplication distributes over addition and 0 is absorbing. As examples consider the semiring of natural numbers $(\mathbb{N}, +, \cdot, 0, 1)$, the Boolean semiring $(\{0, 1\}, \vee, \wedge, 0, 1)$, the tropical semiring $(\mathbb{R}_{\geq 0} \cup \{\infty\}, \min, +, \infty, 0)$ or the arctical semiring $(\mathbb{R}_{\geq 0} \cup \{-\infty\}, \max, +, -\infty, 0)$. A semiring $\mathcal{K}$ is *commutative* if $(K, \cdot, 1)$ is a commutative monoid. A semiring $\mathcal{K}$ is *idempotent* if $k + k = k$ for all $k \in K$. Let $A, B \subseteq K$ be two subsets of the semiring $\mathcal{K}$. We say that A and B *commute element-wise* if $a \cdot b = b \cdot a$ for each $a \in A, b \in B$. Let $\mathcal{K}_A$ be the subsemiring of $\mathcal{K}$ generated by A. Notice that each element $k \in K_A$ can be written as a finite sum of finite products of elements in A. From this it follows that whenever A and B commute element-wise, then K_A and K_B commute element-wise.

We let $\mathcal{F}$ denote a family of functions from $\mathbb{R}_{\geq 0}$ to $\mathcal{K}$. For instance, if $\mathcal{K}$ is the tropical semiring, $\mathcal{F}$ may be the family of linear functions of the form $f(\delta) = k \cdot \delta$ mapping every $\delta \in \mathbb{R}_{\geq 0}$ to the value $k \cdot \delta$ in K (for some $k \in \mathbb{R}_{\geq 0}$). Given $f_1, f_2 \in \mathcal{F}$, we define the pointwise product $f_1 \odot f_2$ of f_1 and f_2 by $(f_1 \odot f_2)(\delta) = f_1(\delta) \cdot f_2(\delta)$. Further, we define the function $\mathbb{1} : \mathbb{R}_{\geq 0} \to K : \delta \mapsto 1$ for each $\delta \in \mathbb{R}_{\geq 0}$. In the following, we assume that $\mathcal{F}$ always contains $\mathbb{1}$.

A *weighted timed automaton* over $\mathcal{K}$, Σ and $\mathcal{F}$ is a tuple $\mathcal{A} = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E, \mathsf{ewt}, \mathsf{lwt})$ such that $(\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E)$ is a timed automaton over Σ and $\mathsf{ewt} : E \to K$ is a weight function for taking an edge, and $\mathsf{lwt} : \mathcal{L} \to \mathcal{F}$ is a function that defines the weight for letting time elapse in a location. A weighted timed automaton $\mathcal{A}$ maps each timed word $w \in T\Sigma^+$ to a weight in K as follows: first, we define the *running weight* $\mathsf{rwt}(r)$ of a run r as above to be $\prod_{i \in \mathsf{dom}(w)} \mathsf{lwt}(l_{i-1})(\delta_i) \cdot \mathsf{ewt}(e_i)$. Then, the *behaviour* $\|\mathcal{A}\| : T\Sigma^+ \to K$ of $\mathcal{A}$ is given by $(\|\mathcal{A}\|, w) = \sum\{\mathsf{rwt}(r) : r \text{ is a successful run of } \mathcal{A} \text{ on } w\}$. A function $\mathcal{T} : T\Sigma^+ \to K$ is called a *timed series*. A timed series $\mathcal{T}$ is said to be *$\mathcal{F}$-recognizable over $\mathcal{K}$ and Σ* if there is a weighted timed automaton $\mathcal{A}$ over $\mathcal{K}$, Σ and $\mathcal{F}$ such that $\|\mathcal{A}\| = \mathcal{T}$.

Example 2 In Fig. 1, we show a weighted timed automaton over the arctical semiring and the family of functions of the form $f(\delta) = \delta^k$ for some $k \in \mathbb{R}_{\geq 0}$ and each $\delta \in \mathbb{R}_{\geq 0}$ together with $\mathbb{1}$. The locations are labeled with their weight functions. The edges are labeled with letters from Σ, clock constraints and resets, and their weights.

This weighted timed automaton models a situation in a real-time system, where a problem occurs (o), is treated (t) and finally fixed (f). For the problem there a two possible treatments, which differ in time conditions and the consumption of a resource (eg. money or energy). The resource consumption grows quadratic and cubic, respectively, with time (in the locations), but also independently on time (at the edges).

A problem and its treatment is modeled as a timed word. For instance, if a problem occurs at time 1.8, its treatment starts at 5.0 and is fixed at 11.0, then the corresponding timed word is $w = (\mathsf{o}, 1.8)(\mathsf{t}, 5.0)(\mathsf{f}, 11.0)$. Given such a timed word, we are interested in the maximum resource consumption for it.

For w, there are two successful runs. Notice that $\mathbb{1}$ maps each time delay to 0, the unit element of the arctical semiring. The running weight of the run using the upper location equals $5 + 6^2 + 100 = 141$ and that of the run using the lower location equals $3 + 6^3 = 219$. Hence, the weighted timed automaton maps w to 219, which corresponds to the maximum resource consumption of this particular problem situation.

□

For $L \subseteq T\Sigma^+$, the *characteristic series* 1_L is defined by $(1_L, w) = 1$ if $w \in L$, 0 otherwise. Notice that a timed automaton $\mathcal{A}$ over Σ can be seen as a weighted

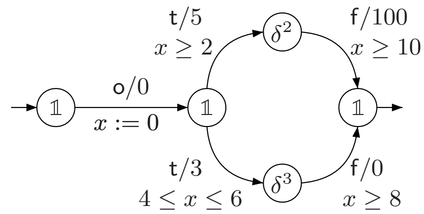

Fig. 1. Weighted timed automaton for Example 2

timed automaton over the Boolean semiring, Σ and $\mathcal{F} = \{\mathbb{1}\}$. The timed series recognized by such a weighted timed automaton is the characteristic series $1_{L(\mathcal{A})}$. The next lemma states some other conditions under which $1_{L(\mathcal{A})}$ is $\mathcal{F}$-recognizable.

Lemma 3 *Let $\mathcal{K}$ be a semiring and $L \subseteq T\Sigma^+$.*

1. *If L is deterministically TA-recognizable over Σ, then 1_L is $\mathcal{F}$-recognizable.*
2. *If $\mathcal{K}$ is idempotent and L is TA-recognizable over Σ, then 1_L is $\mathcal{F}$-recognizable.*

Proof. For the first claim, let $\mathcal{A}$ be a deterministic timed automaton $\mathcal{A}$ such that $L(\mathcal{A}) = L$. Let $\mathcal{A}'$ be the weighted timed automaton obtained from $\mathcal{A}$ by adding weight functions ewt and lwt defined by $\mathsf{ewt}(e) = 1$ for every edge e and $\mathsf{lwt}(l) = \mathbb{1}$ for every location l in $\mathcal{A}$. One can easily see that $\|\mathcal{A}'\| = 1_L$. For the second claim, we use a similar construction. Using the fact that $\mathcal{K}$ is idempotent and hence $1 + 1 = 1$, one can show $\|\mathcal{A}'\| = 1_L$. ■

Given timed series $\mathcal{T}_1, \mathcal{T}_2 : T\Sigma^+ \to K$, we define the *sum* $\mathcal{T}_1 + \mathcal{T}_2$ and the *Hadamard product* $\mathcal{T}_1 \odot \mathcal{T}_2$ pointwise, i.e., by $(\mathcal{T}_1 + \mathcal{T}_2, w) = (\mathcal{T}_1, w) + (\mathcal{T}_2, w)$ and $(\mathcal{T}_1 \odot \mathcal{T}_2, w) = (\mathcal{T}_1, w) \cdot (\mathcal{T}_2, w)$ for each $w \in T\Sigma^+$. If $\mathcal{K}$ is the Boolean semiring, then $+$ and $\odot$ correspond to the union and intersection of timed languages, respectively. Given a mapping $\pi : \Sigma \to \Gamma$ and a timed series $\mathcal{T} : T\Sigma^+ \to K$, we define the renaming $\bar{\pi}(\mathcal{T}) : T\Gamma^+ \to K$ by $(\bar{\pi}(\mathcal{T}), u) = \sum_{\pi(w)=u} (\mathcal{T}, w)$ for all $u \in T\Gamma^+$. Notice that the sum in the equation is finite. For a timed series $\mathcal{T} : T\Gamma^+ \to K$, we define the inverse renaming $\bar{\pi}^{-1}(\mathcal{T}) : T\Sigma^+ \to K$ by $(\bar{\pi}^{-1}(\mathcal{T}), w) = (\mathcal{T}, \pi(w))$ for each $w \in T\Sigma^+$.

Later in the paper, we need closure properties of $\mathcal{F}$-recognizable timed series under these operations. The proof for closure of the class of $\mathcal{F}$-recognizable timed series under sum can be done as usual, namely by taking a disjoint union of two weighted timed automata.

Lemma 4 *The class of $\mathcal{F}$-recognizable timed series is closed under sum.*

In contrast to sum, $\mathcal{F}$-recognizable timed series are not closed under the Hadamard product due to two reasons. First, as in the untimed setting [13], we must ensure that the weights occuring in runs of the two weighted timed automata commute element-wise. This can be solved by assuming $\mathcal{K}$ to be commutative. Second, we have to restrict the location weight functions used in the weighted timed automata in order to guarantee that the pointwise product of each pair of functions is in $\mathcal{F}$. This is illustrated in the next example.

Example 5 In Fig. 2, we show two weighted timed automata $\mathcal{A}_1$ and $\mathcal{A}_2$ over the semiring $(\mathbb{R}, +, \cdot, 0, 1)$ and the family of linear functions. On the right hand side, we show the weighted timed automaton $\mathcal{A}$ with $\|\mathcal{A}\| = \|\mathcal{A}_1\| \odot \|\mathcal{A}_2\|$. It is obtained by using the classical product construction. However, notice that $\mathcal{A}$ uses a quadratic weight function. In fact, it can be proved that there is no weighted timed automaton over the family of linear functions recognizing $\|\mathcal{A}_1\| \odot \|\mathcal{A}_2\|$.

□

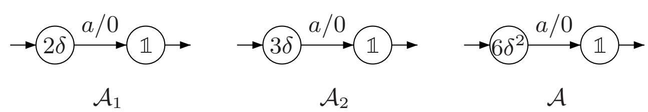

Fig. 2. Weighted Timed Automata $\mathcal{A}_1$ and $\mathcal{A}_2$ and the product automaton $\mathcal{A}$

For this reason, we define the notion of *non-interfering* timed series. Let $\mathcal{A} = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E, \mathsf{ewt}, \mathsf{lwt})$ and $\mathcal{A}' = (\mathcal{L}', \mathcal{L}'_0, \mathcal{L}'_f, \mathcal{C}', E', \mathsf{ewt}', \mathsf{lwt}')$ be two weighted timed automata over $\mathcal{K}$, Σ and $\mathcal{F}$. We say that $\mathcal{A}$ and $\mathcal{A}'$ are *non-interfering* if for all pairs $(l, l') \in \mathcal{L} \times \mathcal{L}'$, whenever there is a run from (l, l') into $\mathcal{L}_f \times \mathcal{L}'_f$, then $\mathsf{lwt}(l) \odot \mathsf{lwt}'(l') \in \mathcal{F}$. This guarantees that the product automaton of $\mathcal{A}$ and $\mathcal{A}'$ is a weighted timed automaton over $\mathcal{F}$. If $\mathcal{F}$ is closed under the pointwise product, all pairs of weighted timed automata are non-interfering. However, $\mathcal{A}$ and $\mathcal{A}'$ are also non-interfering if $\mathsf{lwt}(l) = \mathbb{1}$ or $\mathsf{lwt}'(l') = \mathbb{1}$ for each pair $(l, l') \in \mathcal{L} \times \mathcal{L}'$ from which there is a run into $\mathcal{L}_f \times \mathcal{L}'_f$. Notice that testing for reachability of locations is decidable [1]. We say that two timed series $\mathcal{T}, \mathcal{T}' : T\Sigma^* \to K$ are *non-interfering* over $\mathcal{K}$, Σ and $\mathcal{F}$ if there are two non-interfering weighted timed automata $\mathcal{A}$ and $\mathcal{A}'$ over $\mathcal{K}$, Σ and $\mathcal{F}$ such that $\|\mathcal{A}\| = \mathcal{T}$ and $\|\mathcal{A}'\| = \mathcal{T}'$.

Lemma 6 *Let $\mathcal{K}$ be commutative. If $\mathcal{T}_1, \mathcal{T}_2 : T\Sigma^+ \to K$ are non-interfering timed series over $\mathcal{K}$, Σ and $\mathcal{F}$, then $\mathcal{T}_1 \odot \mathcal{T}_2$ is $\mathcal{F}$-recognizable over $\mathcal{K}$ and Σ.*

Proof. Let $\mathcal{T}_1, \mathcal{T}_2 : T\Sigma^+ \to K$ be non-interfering over $\mathcal{K}$, Σ and $\mathcal{F}$. Then there exist two non-interfering weighted timed automata $\mathcal{A}^i = (\mathcal{L}^i, \mathcal{L}^i_0, \mathcal{L}^i_f, \mathcal{C}^i, E^i, \mathsf{ewt}^i, \mathsf{lwt}^i)$ over $\mathcal{K}$, Σ and $\mathcal{F}$ $(i = 1, 2)$ such that $\|\mathcal{A}^1\| = \mathcal{T}_1$ and $\|\mathcal{A}^2\| = \mathcal{T}_2$. We may assume that $\mathcal{L}^1 \cap \mathcal{L}^2 = \emptyset$ and $\mathcal{C}^1 \cap \mathcal{C}^2 = \emptyset$. Define $\mathcal{L}' = \{(l_1, l_2) \in \mathcal{L}^1 \times \mathcal{L}^2 : \mathsf{lwt}^1(l_1) \odot \mathsf{lwt}^2(l_2) \notin \mathcal{F}\}$. We let $\mathcal{A} = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E, \mathsf{wt})$ be the weighted timed automaton over $\mathcal{K}$, Σ and $\mathcal{F}$ such that

– $\mathcal{L} = (\mathcal{L}^1 \times \mathcal{L}^2) \setminus \mathcal{L}'$,
– $\mathcal{L}_0 = (\mathcal{L}^1_0 \times \mathcal{L}^2_0) \setminus \mathcal{L}'$,

- $\mathcal{L}_0 = (\mathcal{L}_f^1 \times \mathcal{L}_f^2) \backslash \mathcal{L}'$,
- $\mathcal{C} = \mathcal{C}^1 \cup \mathcal{C}^2$,
- $((l_1, l_2), a, \phi_1 \wedge \phi_2, \lambda_1 \cup \lambda_2, (l_1', l_2')) \in E \Leftrightarrow (l_1, a, \phi_1, \lambda_1, l_1') \in E^1$, $(l_2, a, \phi_2, \lambda_2, l_2') \in E^2$ and $(l_1, l_2), (l_1', l_2') \in \mathcal{L}$,
- $\mathsf{ewt}((l_1, l_2), a, \phi_1 \wedge \phi_2, \lambda_1 \cup \lambda_2, (l_1', l_2')) = \mathsf{ewt}^1(l_1, a, \phi_1, \lambda_1, l_1') \cdot \mathsf{ewt}^2(l_2, a, \phi_2, \lambda_2, l_2')$
- $\mathsf{lwt}((l_1, l_2)) = \mathsf{lwt}^1(l_1) \odot \mathsf{lwt}^2(l_2)$ for every $(l_1, l_2) \in \mathcal{L}$.

Intuitively, $\mathcal{A}$ is the classical product automaton, but we remove all "bad" pairs of locations whose pointwise product of their location weight functions is not in $\mathcal{F}$. As a consequence, we obtain $\mathsf{lwt}((l_1, l_2)) \in \mathcal{F}$ for every $(l_1, l_2) \in \mathcal{L}$. The removing of "bad" pairs of locations can be done since by assumption from every such pair there is no run into $\mathcal{L}_f^1 \times \mathcal{L}_f^2$ anyway. Using commutativity of $\mathcal{K}$, it is straightforward to show that for each successful run of $\mathcal{A}$ there are successful runs r^1 of $\mathcal{A}^1$ and r^2 of $\mathcal{A}^2$ such that $\mathsf{rwt}(r) = \mathsf{rwt}(r^1) \cdot \mathsf{rwt}(r^2)$, and vice versa. This can be used to prove $\|\mathcal{A}\| = \|\mathcal{A}^1\| \odot \|\mathcal{A}^2\|$. ■

Lemma 7 *The class of $\mathcal{F}$-recognizable timed series is closed under renamings.*

Proof. Let $\pi : \Sigma \to \Gamma$ be a renaming and let $\mathcal{A} = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E, \mathsf{ewt}, \mathsf{lwt})$ be a weighted timed automaton over $\mathcal{K}$, Σ and $\mathcal{F}$. Define $E' = \{(l, \pi(a), \phi, \lambda, l') : (l, a, \phi, \lambda, l') \in E\}$. Now, define $\mathsf{ewt}' : E' \to K$ by

$$\mathsf{ewt}'(l, b, \phi, \lambda, l') = \sum_{\substack{(l,a,\phi,\lambda,l') \in E \\ \pi(a) = b}} \mathsf{ewt}(l, a, \phi, \lambda, l')$$

and put $\mathcal{A}' = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E', \mathsf{ewt}', \mathsf{lwt})$. Clearly, $\mathcal{A}'$ is a weighted timed automaton over $\mathcal{K}$, Γ and $\mathcal{F}$. Next, we show that $\|\mathcal{A}'\| = \bar{\pi}(\|\mathcal{A}\|)$.

Let $v \in T\Gamma^+$ be of the form $(b_1, t_1)...(b_k, t_k)$ and $\mathcal{R}$ be the set of successful runs of $\mathcal{A}$ on $w \in T\Sigma^+$ such that $\pi(w) = v$. Let $r, r' \in \mathcal{R}$ be of the form

$$r = (l_0, \nu_0) \xrightarrow{\delta_1, e_1} ... \xrightarrow{\delta_k, e_k} (l_k, \nu_k)$$

and

$$r' = (l_0', \nu_0') \xrightarrow{\delta_1, e_1'} ... \xrightarrow{\delta_k, e_k'} (l_k', \nu_k').$$

We say that r and r' are equivalent, written $r \equiv r'$, if $l_i = l_i'$ and $\nu_i = \nu_i'$ for $0 \leq i \leq |w|$. Intuitively, $r \equiv r'$ if the runs differ at most in the labels, guards and reset sets of their edges, provided that π maps the labels to the same image and the resulting clock valuations are the same. We use $\mathcal{R}_{/\equiv}$ to denote the set of all equivalence classes induced by $\equiv$. From the fact that $\equiv$ induces a partition of $\mathcal{R}$, we obtain

$$\sum_{\substack{w \in T\Sigma^+ \\ \pi(w) = v}} (\|\mathcal{A}\|, w) = \sum_{R \in \mathcal{R}_{/\equiv}} \sum_{r \in R} \mathsf{rwt}(r).$$

Next, let $R \in \mathcal{R}_{/\equiv}$ and $r \in R$ be of the form $(l_0, \nu_0) \xrightarrow{\delta_1, e_1} ... \xrightarrow{\delta_k, e_k} (l_k, \nu_k)$. We define r_R to be the sequence that is obtained from r by replacing $e_i = (l_{i-1}, a_i, \phi_i, \lambda_i, l_i)$ for each $i \in \mathsf{dom}(w)$ by the corresponding edge

$e'_i = (l_{i-1}, \pi(a_i), \phi_i, \lambda_i, l_i) \in E'$. We neither change the clock constraints ϕ_i nor the reset sets λ_i, so we have $\nu'_{i-1} \models \phi_i$ and $\nu'_i = (\nu'_{i-1} + \delta_i)[\lambda_i := 0]$ for each $i \in \mathsf{dom}(w)$, and thus, r_R is a successful run of $\mathcal{A}'$ on v. Moreover, the set of successful runs of $\mathcal{A}'$ on v is precisely the set of such runs r_R for each $R \in \mathcal{R}_{/\equiv}$, i.e., we have

$$(\|\mathcal{A}'\|, v) = \sum_{R \in \mathcal{R}_{/\equiv}} \mathsf{rwt}(r_R),$$

where r_R is the run of $\mathcal{A}'$ on v obtained from an arbitrary run $r \in \mathcal{R}$ as described above. Next, we show that for every $R \in \mathcal{R}_{/\equiv}$ we have $\mathsf{rwt}(r_R) = \sum_{r \in R} \mathsf{rwt}(r)$, which, with the help of the two equations above, implies the result. Let $R \in \mathcal{R}_{/\equiv}$ and $r \in R$ as above. Then, the following equation holds by distributivity of $\mathcal{K}$:

$$\begin{aligned}
\mathsf{rwt}(r_R) &= \prod_{1 \le i \le |v|} \mathsf{lwt}(l_{i-1})(\delta_i) \cdot \mathsf{ewt}(e'_i) \\
&= \prod_{1 \le i \le |v|} \mathsf{lwt}(l_{i-1})(\delta_i) \cdot \sum_{\substack{(l_{i-1}, a_i, \phi_i, \lambda_i, l_i) \in E \\ \pi(a_i) = b_i}} \mathsf{ewt}(l_{i-1}, a_i, \phi_i, \lambda_i, l_i) \\
&= \sum_{\substack{(l_{i-1}, a_i, \phi_i, \lambda_i, l_i) \in E \\ \pi(a_i) = b_i}} \prod_{1 \le i \le |v|} \mathsf{lwt}(l_{i-1})(\delta_i) \cdot \mathsf{ewt}(l_{i-1}, a_i, \phi_i, \lambda_i, l_i) \\
&= \sum_{r \in R} \mathsf{rwt}(r).
\end{aligned}$$

Hence, $(\|\mathcal{A}'\|, v) = \sum_{\substack{w \in T\Sigma^+ \\ \pi(w) = v}} (\|\mathcal{A}\|, w)$, and thus $\|\mathcal{A}'\| = \bar{\pi}(\|\mathcal{A}\|)$. ■

Lemma 8 *The class of $\mathcal{F}$-recognizable timed series is closed under inverse renamings.*

Proof. Let $\pi : \Sigma \to \Gamma$ be a renaming and let $\mathcal{A} = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E, \mathsf{ewt}, \mathsf{lwt})$ be a weighted timed automaton over $\mathcal{K}$, Σ and $\mathcal{F}$. Define $E' = \{(l, a, \phi, \lambda, l') : (l, \pi(a), \phi, \lambda, l') \in \mathcal{L}\}$ and $\mathsf{ewt}'(l, a, \phi, \lambda, l') = \mathsf{ewt}(l, \pi(a), \phi, \lambda, l')$. Then the behaviour of the weighted timed automaton $\mathcal{A}' = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E', \mathsf{ewt}', \mathsf{lwt})$ over $\mathcal{K}$, Σ and $\mathcal{F}$ precisely corresponds to $\bar{\pi}^{-1}(\|\mathcal{A}\|)$. ■

2 Weighted Monadic Logic of Relative Distance

In this section, we define a logic for the specification of recognizable timed series. Then we aim to show that this logic is expressively equivalent to weighted timed automata. The first result of this kind, the equivalence between finite automata and sentences in a monadic second-order (MSO) logic over some alphabet Σ, denoted by $\mathsf{MSO}(\Sigma)$, was obtained by Büchi [10]. A result of this kind has recently been presented for *weighted* finite automata over semirings by Droste and Gastin [13]. The authors define a weighted MSO logic, where atomic formulas may additionally comprise elements of the semiring. The full weighted MSO logic is expressively stronger than weighted finite automata. Thus the authors consider a syntactically defined fragment and show that it is expressively equivalent to the class of weighted finite automata. For timed automata, a Büchi theorem was given by Wilke [29]. Wilke extends $\mathsf{MSO}(\Sigma)$ with formulas of the form $d(y,z) \sim c$, called *distance predicates*, where y, z are first-order variables, $\sim \in \{<, \leq, =, \geq, >\}$ and $c \in \mathbb{N}$. Formulas of this form are supposed to express that the time distance between the positions y and z in a timed word satisfies the constraint $\sim c$. However, it is shown by Alur and Henzinger [25], that the unrestricted use of distance predicates leads to an undecidable theory. For instance, formulas like $\forall y.\forall z.d(y,z) = 1$ can be used to specify timed words that correspond precisely to the halting computations of a turing machine. Moreover, since TA-recognizable timed languages are not closed under complement, one cannot expect to find a full MSO logic that is expressively complete for timed automata [29]. For this reason, Wilke restricts the use of distance predicates. He introduces *relative distance predicates* of the form $\overleftarrow{\mathsf{d}}(D,y) \sim c$, where D is a second-order variable, which may only be existentially quantified. Furthermore, this may only be done at the beginning of a formula. The resulting logic is known as *relative distance logic*. Wilke [29] shows that timed languages definable in this logic can be fully characterized in terms of timed automata.

We recall the syntax and semantics of the relative distance logic over Σ. We do this in two steps. We start with the definition of the **underlying auxiliary logic**, denoted by $\mathsf{MSO}(T\Sigma^+)$, which is an extension of $\mathsf{MSO}(\Sigma)$ with relative distance predicates. Formulas of $\mathsf{MSO}(T\Sigma^+)$ are defined by the following grammar

$$\varphi ::= P_a(y) \mid y = z \mid y < z \mid y \in X \mid \overleftarrow{\mathsf{d}}(D,y) \sim c \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists y.\varphi \mid \exists X.\varphi,$$

where y, z are first-order variables, X, D are second-order variables, $a \in \Sigma$, $c \in \mathbb{N}$ and $\sim \in \{<, \leq, =, \geq, >\}$. Notice that the syntax of $\mathsf{MSO}(T\Sigma^+)$ does not allow for the quantification of D occuring in a relative distance predicate. For this reason, we may temporarily interprete D as a constant and note that $\mathsf{MSO}(T\Sigma^+)$ is a full MSO logic. As usual, we may use true, $\varphi \wedge \psi$, $\varphi \longrightarrow \psi$, $\varphi \longleftrightarrow \psi$, $\forall y.\varphi$ and $\forall X.\varphi$ as abbreviations for $\neg\varphi \vee \varphi$, $\neg(\neg\varphi \vee \neg\psi)$, $\neg\varphi \vee \psi$, $(\varphi \longrightarrow \psi) \wedge (\psi \longrightarrow \varphi)$, $\neg\exists y.\neg\varphi$, and $\neg\exists X.\neg\varphi$, respectively.

In the relative distance logic, D will be allowed to be existantially quantified at the beginning of a formula. Formally, we define the **relative distance logic**, denoted by $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$, to be the smallest class of formulas containing all formulas generated by the next two rules.

1. If $\varphi \in \mathsf{MSO}(T\Sigma^+)$, so is $\varphi \in \mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$.
2. If $\varphi \in \mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$, so is $\exists D.\varphi \in \mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$.

Formulas of $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$ are interpreted over timed words over Σ. For this, we associate with $w \in T\Sigma^+$ the relational structure consisting of the domain $\mathsf{dom}(w)$ together with the binary relation $P_a = \{i \in \mathsf{dom}(w) : a_i = a\}$ and the usual $=$

and $<$ relations on $\mathsf{dom}(w)$. We further define the binary relation $\overleftarrow{\mathsf{d}}(\cdot,\cdot) \sim c$ to be $(I, i) \in 2^{\mathsf{dom}(w)} \times \mathsf{dom}(w)$ such that one of the following conditions is satisfied

- there is some $j \in I$ such that $j < i$, $t_i - t_j \sim c$ and there is no $k \in I$ with $j < k < i$,
- there is no $j \in I$ such that $j < i$, and $t_i - 0 \sim c$.

For $\varphi \in \mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$, let $\mathsf{Free}(\varphi)$ be the set of free variables, i.e., variables not bound by any quantifier, $\mathcal{V} \supseteq \mathsf{Free}(\varphi)$ be a finite set of first- and second-order variables, and σ be a $(\mathcal{V}, w)$-assignment mapping first-order (second-order, respectively) variables to elements (subsets, respectively) in $\mathsf{dom}(w)$. For $i \in \mathsf{dom}(w)$, we let $\sigma[y \to i]$ be the assignment that maps y to i and agrees with σ on every variable $\mathcal{V}\backslash\{y\}$. Similarly, we define $\sigma[X \to I]$ for any $I \subseteq \mathsf{dom}(w)$. For Σ, we define the extended alphabets $\Sigma_{\mathcal{V}} = \Sigma \times \{0,1\}^{\mathcal{V}}$ for every finite set $\mathcal{V}$ of variables. A timed word $w \in T\Sigma^+$ and a $(\mathcal{V}, w)$-assignment σ are encoded as timed word over the extended alphabet $\Sigma_{\mathcal{V}}$. A timed word over $\Sigma_{\mathcal{V}}$ is written as $((\bar{a}, \sigma), \bar{t})$, where $(\bar{a}, \bar{t})$ is the projection over $T\Sigma^+$ and σ is the projection over $\{0,1\}^{\mathcal{V}}$. Then, σ represents a *valid* assignment over $\mathcal{V}$ if for each first-order variable $y \in \mathcal{V}$, the y-row of σ contains exactly one 1. In this case, σ is identified with the $(\mathcal{V}, w)$-assignment such that for every first-order variable $y \in \mathcal{V}$, $\sigma(y)$ is the position of the 1 in the y-row, and for each second-order variable $X \in \mathcal{V}$, $\sigma(X)$ is the set of positions with a 1 in the X-row.

Example 9 Let $w = (a, 2.0)(a, 3.5)(b, 4.2)$ be a timed word over Σ. Further let $\mathcal{V} = \{y, X\}$ and consider the valid $(\mathcal{V}, w)$-assignment σ with $\sigma(y) = 2$ and $\sigma(X) = \{1, 2\}$. We encode w and σ as the timed word $\left(\begin{smallmatrix} a \\ 0 \\ 1 \end{smallmatrix}, 2.0\right)\left(\begin{smallmatrix} a \\ 1 \\ 1 \end{smallmatrix}, 3.5\right)\left(\begin{smallmatrix} b \\ 0 \\ 0 \end{smallmatrix}, 4.2\right)$ over $\Sigma_{\mathcal{V}}$. □

We define $N_{\mathcal{V}} = \{((\bar{a}, \sigma), \bar{t}) \in T(\Sigma_{\mathcal{V}})^+ : \sigma \text{ is a valid } (\mathcal{V}, (\bar{a}, \bar{t}))\text{-assignment}\}$. The definition that $((\bar{a}, \sigma), \bar{t})$ satisfies φ, written $((\bar{a}, \sigma), \bar{t}) \models \varphi$, is as usual provided that the domain of σ contains $\mathsf{Free}(\varphi)$. We let $L_{\mathcal{V}}(\varphi) = \{((\bar{a}, \sigma), \bar{t}) \in N_{\mathcal{V}} : ((\bar{a}, \sigma), \bar{t}) \models \varphi\}$. The formula φ *defines* the timed language $L(\varphi) = L_{\mathsf{Free}(\varphi)}(\varphi)$. A formula φ is a *sentence* if $\mathsf{Free}(\varphi) = \emptyset$. A timed language $L \subseteq T\Sigma^+$ is $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$-*definable* if there exists a sentence $\varphi \in \mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$ such that $L(\varphi) = L$.

Theorem 10 ([29]) *A timed language $L \subseteq T\Sigma^+$ is $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$-definable if and only if L is TA-recognizable over Σ. The transformations from a timed automaton over Σ to a $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$-sentence and back can be done efficiently.*

Now, we turn to the weighted extension of these logics. *For this, we fix a semiring $\mathcal{K}$ and a family $\mathcal{F}$ of functions from $\mathbb{R}_{\geq 0}$ to K including $\mathbb{1}$.* We extend $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$ with two kinds of weighted formulas of the form k (where $k \in K$) and $f(y)$ (where $f \in \mathcal{F}$ and y is a first-order variable), the semantics of which correspond to the weights of edges and locations, respectively, in weighted timed automata. Again, we start with the **underlying auxiliary logic** and define it by the following grammar.

$$\varphi ::= P_a(y) \mid \neg P_a(y) \mid y = z \mid \neg(y = z) \mid y < z \mid \neg(y < z) \mid y \in X \mid \neg(y \in X) \mid \overleftarrow{\mathsf{d}}(D, y) \sim c \mid \neg(\overleftarrow{\mathsf{d}}(D, y) \sim c) \mid k \mid f(y) \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid Qy.\varphi \mid QX.\varphi$$

where y, z are first-order variables, X, D are second-order variables, $Q \in \{\exists, \forall\}$, $a \in \Sigma$, $c \in \mathbb{N}$, $\sim \in \{<, \leq, =, \geq, >\}$, $k \in K$ and $f \in \mathcal{F}$. We use $\mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ to denote the collection of all such formulas. Formulas of the form k and $f(y)$ are called *weighted atomic formulas*.

Notice that negation may only be applied to atomic formulas of $\mathsf{MSO}(T\Sigma^+)$. This is because for arbitrary semirings it is not clear what the negation of a weighted atomic formula should mean. In the following, we use the term *atomic formulas* to refer to atomic formulas of $\mathsf{MSO}(T\Sigma^+)$ and their negations.

Finally, we define the **weighted relative distance logic**, denoted by $\mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$, to be the smallest class of formulas containing all formulas generated by the next two rules.

1. If $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, then $\varphi \in \mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$.
2. If $\varphi \in \mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$, then $\exists D.\varphi \in \mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$.

Next, we define the semantics of this logic. Let $\varphi \in \mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$ and $\mathcal{V} \supseteq \mathsf{Free}(\varphi)$. The $\mathcal{V}$-semantics of φ is a timed series $[\![\varphi]\!]_\mathcal{V} : T(\Sigma_\mathcal{V})^+ \to K$. Let $(\bar{a}, \bar{t}) \in T\Sigma^+$. If σ is a valid $(\mathcal{V}, (\bar{a}, \bar{t}))$-assignment, $\big([\![\varphi]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) \in K$ is defined inductively as follows:

$$\begin{aligned}
\big([\![\varphi]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) &= \big(1_{L_\mathcal{V}(\varphi)}, ((\bar{a}, \sigma), \bar{t})\big) \text{ if } \varphi \text{ is atomic} \\
\big([\![k]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) &= k \\
\big([\![f(y)]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) &= f(t_{\sigma(y)} - t_{\sigma(y)-1}) \\
\big([\![\varphi \vee \varphi']\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) &= \big([\![\varphi]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) + \big([\![\varphi']\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) \\
\big([\![\varphi \wedge \varphi']\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) &= \big([\![\varphi]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) \cdot \big([\![\varphi']\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) \\
\big([\![\exists y.\varphi]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) &= \sum_{i \in \mathsf{dom}((\bar{a}, \bar{t}))} \big([\![\varphi]\!]_{\mathcal{V} \cup \{y\}}, ((\bar{a}, \sigma[y \to i]), \bar{t})\big) \\
\big([\![\forall y.\varphi]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) &= \prod_{i \in \mathsf{dom}((\bar{a}, \bar{t}))} \big([\![\varphi]\!]_{\mathcal{V} \cup \{y\}}, ((\bar{a}, \sigma[y \to i]), \bar{t})\big) \\
\big([\![\exists X.\varphi]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) &= \sum_{I \subseteq \mathsf{dom}((\bar{a}, \bar{t}))} \big([\![\varphi]\!]_{\mathcal{V} \cup \{X\}}, ((\bar{a}, \sigma[X \to I]), \bar{t})\big) \\
\big([\![\exists D.\varphi]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) &= \sum_{I \subseteq \mathsf{dom}((\bar{a}, \bar{t}))} \big([\![\varphi]\!]_{\mathcal{V} \cup \{D\}}, ((\bar{a}, \sigma[D \to I]), \bar{t})\big) \\
\big([\![\forall X.\varphi]\!]_\mathcal{V}, ((\bar{a}, \sigma), \bar{t})\big) &= \prod_{I \subseteq \mathsf{dom}(\bar{a}, \bar{t})} \big([\![\varphi]\!]_{\mathcal{V} \cup \{X\}}, ((\bar{a}, \sigma[X \to I]), \bar{t})\big)
\end{aligned}$$

For σ not a valid $(\mathcal{V}, (\bar{a}, \bar{t}))$-assignment, we define $\big([\![\varphi]\!]_\mathcal{V}((\bar{a}, \sigma), \bar{t})\big) = 0$. We write $[\![\varphi]\!]$ for $[\![\varphi]\!]_{\mathsf{Free}(\varphi)}$.

Remark 11 If $\mathcal{K}$ is the Boolean semiring, then $\mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K},\Sigma,\mathcal{F})$ corresponds to $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$. This is because every formula in $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$ is language equivalent to a formula where negation is applied to atomic subformulas only. Also, every such formula in $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$ can be seen as a formula of $\mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K},\Sigma,\mathcal{F})$.

Example 12 Consider the formula $\varphi = \exists D.\exists y.P_b(y) \wedge \overleftarrow{\mathsf{d}}(D,y) < 2$ and let $w = (a,1.0)(a,2.0)(b,3.0)$. If $\mathcal{K}$ is the Boolean semiring or, equivalently, we interprete φ as an $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$-formula, we have $(\llbracket\varphi\rrbracket, w) = 1$, as, for instance, the time difference between the third and second position is less than 2, so we may choose $\sigma(y) = 3$ and $\sigma(D) = \{2\}$. If on the other hand, we let $\mathcal{K}$ be the semiring of the natural numbers, we have $(\llbracket\varphi\rrbracket, w) = 4$, since there are 4 different assignments such that $P_b(y) \wedge \overleftarrow{\mathsf{d}}(D,y) < 2$ is evaluated to 1. In fact, using the semiring of the natural numbers, we can *count* how often a certain property holds. This may give rise to interesting applications in the field of verification. □

Example 13 We let $\mathcal{K}$ be the arctical semiring and $f(\delta) = \delta$ for each $\delta \in \mathbb{R}_{\geq 0}$. Then, the formula $\varphi = \exists y.f(y)$ computes for each timed word w the maximal time difference $t_i - t_{i-1}$ between two consecutive events. Formally, $(\llbracket\varphi\rrbracket, w) = \max\{t_i - t_{i-1} : i \in \mathsf{dom}(w)\}$ for each $w \in T\Sigma^+$. □

The following lemma states that for each formula φ of our logic, the semantics for different finite sets $\mathcal{V}$ of variables containing $\mathsf{Free}(\varphi)$ are consistent with each other. It can be proved by induction on the structure of $\mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K},\Sigma,\mathcal{F})$

Lemma 14 *Let* $\varphi \in \mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K},\Sigma,\mathcal{F})$ *and* $\mathcal{V}$ *a finite set of variables containing* $\mathsf{Free}(\varphi)$*. Then*

$$\big(\llbracket\varphi\rrbracket_{\mathcal{V}}, ((\bar{a},\sigma),\bar{t})\big) = \big(\llbracket\varphi\rrbracket, ((\bar{a},\sigma_{|\mathsf{Free}(\varphi)}),\bar{t})\big)$$

for each $((\bar{a},\sigma),\bar{t}) \in T(\Sigma_{\mathcal{V}})^+$ *such that* σ *is a valid* $(\mathcal{V},(\bar{a},\bar{t}))$*-assignment.*

Let $\mathfrak{L} \subseteq \mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K},\Sigma,\mathcal{F})$. A timed series $\mathcal{T} : T\Sigma^+ \to K$ is called $\mathfrak{L}$-*definable* if there is a sentence $\varphi \in \mathfrak{L}$ such that $\llbracket\varphi\rrbracket = \mathcal{T}$. The goal of this section is to find a suitable fragment $\mathfrak{L} \subseteq \mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K},\Sigma,\mathcal{F})$ such that $\mathfrak{L}$ is expressively equivalent to weighted timed automata. In other words, we want to generalize Theorem 10 to the weighted setting. It is not surprising that $\mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K},\Sigma,\mathcal{F})$ itself does not constitute a suitable candidate for $\mathfrak{L}$, since also in the untimed setting, full weighted MSO logic is expressively stronger than weighted finite automata [13]. In the next section, we explain the problems that occur when we do not restrict the logic. For simplicity, we do this exemplarily for the case of *idempotent and commutative* semirings. To be as general as possible, we will moreover consider families of functions that are *not closed under pointwise product*. Notice that this setting includes e.g. the weighted timed automaton of Example 2. Stepwisely, we develop solutions resulting in a fragment of $\mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K},\Sigma,\mathcal{F})$ for which we are able to present a Büchi theorem for weighted timed automata over this particular setting. Later we will show how to generalize this approach to arbitrary semirings.

3 From Logic To Weighted Timed Automata

We fix an idempotent and commutative semiring $\mathcal{K}$. Moreover, we assume that $\mathcal{F}$ is not necessarily closed under pointwise product.

We want to develop a fragment $\mathfrak{L} \subseteq \overleftarrow{\mathcal{L}\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$ in which, for every sentence $\varphi \in \mathfrak{L}$, $[\![\varphi]\!]$ is an $\mathcal{F}$-recognizable timed series. As in the classical setting, the proof for this is done by induction over the structure of the logic: for the induction base, we show that for every atomic formula φ in $\mathfrak{L}$, there is a weighted timed automaton $\mathcal{A}$ over $\mathcal{K}$, $\Sigma_{\mathsf{Free}(\varphi)}$[2] and $\mathcal{F}$ such that $\|\mathcal{A}\| = [\![\varphi]\!]$. For the induction step, we need to show that $\mathcal{F}$-recognizable timed series are closed under the operators of $\mathfrak{L}$. In the case of disjunction and existential quantification, the proofs are very similar to the classical case [27, 13]. In the case of conjunction and universal quantification, however, problems arise. The problems with unrestricted use of conjunction are due to the fact that $\mathcal{F}$-recognizable timed series are not closed under Hadamard product in general (see Example 5). Problems with unrestricted use of universal quantification are due to the fact that the semantics of formulas may grow with the size of a timed word too fast to be recognizable by a weighted timed automaton. This is demonstrated in the next example.

Example 15 Let $\mathcal{K}$ be the arctical semiring and $\mathcal{F}$ be the family of functions of the form δ^k for some $k \in \mathbb{N}$ and all $\delta \in \mathbb{R}_{\geq 0}$. We let f be the function defined by $f(\delta) = \delta^1$ for each $\delta \in \mathbb{R}_{\geq 0}$. We consider the formula $\varphi = \forall z.\exists y.f(y)$. Then we have $([\![\varphi]\!], w) = |w| \cdot \max\{t_i - t_{i-1} : i \in \mathsf{dom}(w)\}$ for each $w \in T\Sigma^+$. However, $[\![\varphi]\!]$ is not $\mathcal{F}$-recognizable, as is proved in the following: assume $\mathcal{A} = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E, \mathsf{ewt}, \mathsf{lwt})$ is a weighted timed automaton over $\mathcal{K}$, Σ and $\mathcal{F}$ such that $\|\mathcal{A}\| = [\![\varphi]\!]$. Notice that ewt assigns constants to the edges, respectively. Thus, for each location l, there is some $\delta \in \mathbb{R}_{\geq 0}$ such that $\mathsf{lwt}(l)(\delta)$ is strictly greater than each of these constants. For this reason, we may assume $\mathsf{ewt}(e) = 0$ for each $e \in E$. For each $l \in \mathcal{L}$, we use c_l to denote the constant to which power the time delay δ is taken of, i.e., if $\mathsf{lwt}(l)(\delta) = \delta^n$ for each $\delta \in \mathbb{R}_{\geq 0}$ and some $n \in K$, then, $c_l = n$. Let $M = \max\{c_l : l \in \mathcal{L}\}$. Then, for every timed word $w \in T\Sigma^+$ and for each run of $\mathcal{A}$ on w we have $\mathsf{rwt}(r) \leq \sum_{1 \leq i \leq |w|}(t_i - t_{i-1})^M$. Furthermore, we have $(\|\mathcal{A}\|, w) = \max\{\mathsf{rwt}(r) : r \text{ is a run of } \mathcal{A} \text{ on } w\}$ and thus $(\|\mathcal{A}\|, w) \leq \sum_{1 \leq i \leq |w|}(t_i - t_{i-1})^M$. Now choose $w \in T\Sigma^+$ such that $|w| > 2^M$ and there exists some $i \in \mathsf{dom}(w)$ such that $t_i - t_{i-1} = 2$ and for all $j \in \mathsf{dom}(w)$ with $j \neq i$ we have $t_j - t_{j-1} = 0$. Then we obtain $(\|\mathcal{A}\|, w) < [\![\varphi]\!]$, a contradiction. The timed series $[\![\varphi]\!]$ grows too fast to be $\mathcal{F}$-recognizable. □

Similar examples can be given for $\forall X$. For this reason, we restrict the application of conjunction and universal quantification and consider a *syntactically restriced* fragment of $\mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$. We present the definition of this fragment in the

[2] Notice that atomic and weighted atomic formulas may contain free variables. Thus the weighted timed automata recognizing the semantics of an atomic formula φ are defined over the extended alphabet $\Sigma_{\mathsf{Free}(\varphi)}$.

following. We start with the definition of *unweighted* and *almost unambiguous* formulas.

We say that a formula $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is *unweighted*, if it does not contain any weighted atomic formulas. It can be easily seen that unweighted formulas are in $\mathsf{MSO}(T\Sigma^+)$.

Lemma 16 *The semantics* $[\![\varphi]\!]$ *of each unweighted formula* $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ *takes only values in* $\{0, 1\}$. *In particular,* $[\![\varphi]\!] = 1_{L(\varphi)}$.

Proof. Follows from the semiring axioms and idempotence of $\mathcal{K}$. ■

Let y be a first-order variable. We say that a formula $\psi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is *almost unambiguous over* y, if it is in the disjunctive and conjunctive closure of unweighted formulas, constants $k \in K$ and formulas $f(y)$ for some $f \in \mathcal{F}$, such that $f(y)$ may appear at most once in every subformula of ψ of the form $\psi_1 \wedge \psi_2$.

Example 17 Let $f, g \in \mathcal{F}$, $k \in K$ and let z be a first-order variable. The formulas $\exists y.f(y)$ and $\exists y.k$ are not almost unambiguous over y. The formula $[(f(y) \wedge k) \vee g(y)] \wedge P_a(z)$ is almost unambiguous over y, whereas the formulas $[(f(y) \wedge k) \vee g(z)] \wedge P_a(z)$ and $f(y) \wedge k \wedge g(y) \wedge P_a(z)$ are not. □

We say that two formulas ψ and ζ are *equivalent*, written $\psi \equiv \zeta$, if $[\![\psi]\!]_{\mathsf{Free}(\psi) \cup \mathsf{Free}(\zeta)} = [\![\zeta]\!]_{\mathsf{Free}(\psi) \cup \mathsf{Free}(\zeta)}$. The next lemma can be proved using the semiring axioms, commutativity of $\mathcal{K}$ and Lemma 16.

Lemma 18 *Let* y *be a first-order variable,* $k_1, k_2 \in K$, $f \in \mathcal{F}$, *and* $\psi_1, \psi_2, \psi_3 \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ *be unweighted. Then, the following equivalences hold:*

1. $\psi_1 \wedge (\psi_2 \vee \psi_3) \equiv (\psi_1 \wedge \psi_2) \vee (\psi_1 \wedge \psi_3)$
2. $\psi_1 \wedge \psi_2 \equiv \psi_2 \wedge \psi_1$
3. $\psi_1 \equiv 1 \wedge \psi_1$
4. $\psi_1 \equiv \mathbb{1}(y) \wedge \psi_1$
5. $\psi_1 \equiv \psi_1 \wedge \mathsf{true}$
6. $k_1 \wedge k_2 \equiv k_1 \cdot k_2$

Lemma 19 *Let* y *be a first-order variable and* $\psi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ *be almost unambiguous over* y. *Then there is a formula* $\zeta \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ *such that* ζ *is of the form* $\bigvee_{1 \leq i \leq n} f_i(y) \wedge k_i \wedge \psi_i$ *for some* $n \in \mathbb{N}$, $f_i \in \mathcal{F}$, $k_i \in K$ *and unweighted* $\psi_i \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ *for each* $i \in \{1, ..., n\}$, *and* $\zeta \equiv \psi$.

Proof. We transform every almost unambiguous formula into the appropriate form using Lemma 18. First, transform every formula of the form $\psi_1 \wedge (\psi_2 \vee \psi_3)$ into the form $(\psi_1 \wedge \psi_2) \vee (\psi_1 \wedge \psi_3)$ and every formula of the form $(\psi_1 \vee \psi_2) \wedge \psi_3$ into the form $(\psi_1 \wedge \psi_3) \vee (\psi_2 \wedge \psi_3)$. Then, each disjunct can be put in the right form using Lemma 18. ■

Given a formula $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, we define the set $\mathcal{V}_f(\varphi)$ to be the set of all first-order variables y such that $f(y)$ appears in φ.

We define the **syntactically restricted auxiliary logic** $\mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ to be the smallest class of formulas generated by the following rules.

1. If $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is an atomic or a weighted atomic formula, then $\varphi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
2. If $\varphi, \psi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, then $\varphi \vee \psi, \exists y.\varphi, \exists X.\varphi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
3. If $\varphi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is unweighted, then $\forall X.\varphi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
4. If $\varphi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is almost unambiguous over y, then $\forall y.\varphi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
5. If $\varphi, \psi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ and at least one of the following conditions holds
   - $\mathcal{V}_f(\varphi) = \emptyset$,
   - $\mathcal{V}_f(\psi) = \emptyset$,
   - $\mathcal{V}_f(\varphi) \cap \mathcal{V}_f(\psi) = \emptyset$, $\mathcal{V}_f(\varphi) \subseteq \mathsf{Free}(\varphi)$, and $\mathcal{V}_f(\psi) \subseteq \mathsf{Free}(\psi)$,

   then $\varphi \wedge \psi \wedge \bigwedge_{\substack{y \in \mathcal{V}_f(\varphi), z \in \mathcal{V}_f(\psi) \\ y \neq z}} \neg(y = z) \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.

The motivation for these restrictions will be explained later in this section.

Remark 20 If $\mathcal{F}$ is closed under pointwise product, we can replace condition 5 by the following rule: if $\varphi, \psi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, then $\varphi \wedge \psi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.

Now, we want to show that for each formula $\varphi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, there is a weighted timed automaton $\mathcal{A}$ such that $\|\mathcal{A}\| = [\![\varphi]\!]$. As mentioned before, this is done by induction. However, due to our restriction on conjunction in $\mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, we will prove a stronger result, stated in the next theorem.

Given a formula $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, we let $\mathsf{Func}(\varphi)$ be the set of functions $f \in \mathcal{F}$ such that φ contains a subformula $f(y)$ for some first-order variable y. Given a weighted timed automaton $\mathcal{A}$, we let $\mathsf{Func}(\mathcal{A})$ be the set of functions f such that $\mathsf{lwt}(l) = f$ for some location l in $\mathcal{A}$.

Theorem 21 *Let $\varphi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ be a syntactically restricted formula. Then for each finite set $\mathcal{V} \supseteq \mathsf{Free}(\varphi)$ there is some weighted timed automaton $\mathcal{A}_\varphi$ over $\mathcal{K}$, $\Sigma_\mathcal{V}$ and $\mathcal{F}$ such that*

1. $\|\mathcal{A}_\varphi\| = [\![\varphi]\!]_\mathcal{V}$,
2. $\mathsf{Func}(\mathcal{A}_\varphi) \subseteq \mathsf{Func}(\varphi) \cup \{\mathbb{1}\}$,
3. *for each formula $f(y)$ occurring in φ with $y \in \mathsf{Free}(\varphi)$, whenever $\mathsf{lwt}(l) = f$ for some location l in $\mathcal{A}_\varphi$, then for each edge $(l, (a, \sigma), \phi, \lambda, l')$ in $\mathcal{A}_\varphi$ we have $\sigma(y) = 1$.*

The remainder of this section is devoted to the proof of this theorem. The proof is done by induction on the construction of $\mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$-formulas. However, we will only show the claim for $\mathcal{V} = \mathsf{Free}(\varphi)$. For each other finite set $\mathcal{V}'$ of variables with $\mathcal{V}' \supseteq \mathsf{Free}(\varphi)$, the claim follows from the following Lemma together with the fact that $[\![\varphi]\!]_{\mathcal{V}'} = \bar{\pi}^{-1}([\![\varphi]\!]) \odot 1_{N_{\mathcal{V}'}}$.

Let $\mathcal{V}, \mathcal{V}'$ be two finite sets of first- and second-order variables such that $\mathcal{V} \subseteq \mathcal{V}'$. Let $\pi : \Sigma_{\mathcal{V}'} \to \Sigma_{\mathcal{V}}$ be a projection defined by $(a, \sigma) \mapsto (a, \sigma_{|\mathcal{V}})$.

Lemma 22 *For each weighted timed automaton $\mathcal{A}$ over $\Sigma_{\mathcal{V}}$, there is a weighted timed automaton $\mathcal{A}_{\mathcal{V}'}$ over $\Sigma_{\mathcal{V}'}$ such that*

1. $\|\mathcal{A}_{\mathcal{V}'}\| = \bar{\pi}^{-1}(\|\mathcal{A}\|) \odot 1_{N_{\mathcal{V}'}}$,
2. $\mathsf{Func}(\mathcal{A}_{\mathcal{V}'}) = \mathsf{Func}(\mathcal{A})$,
3. *for each $f \in \mathcal{F}$ and each first-order variable $y \in \mathcal{V}$, if each edge of the form $(l, (a, \sigma), \phi, \lambda, l')$ in $\mathcal{A}$ with $\mathsf{lwt}(l) = f$ satisfies $\sigma(y) = 1$, then each edge of the form $(l_1, (b, \sigma'), \phi', \lambda', l_2)$ in $\mathcal{A}_{\mathcal{V}'}$ with $\mathsf{lwt}'(l_1) = f$ satisfies $\sigma'(y) = 1$.*

Proof. Let $\mathcal{A} = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E, \mathsf{ewt}, \mathsf{lwt})$ be a weighted timed automaton over $\Sigma_{\mathcal{V}}$. We define $\mathcal{A}' = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E', \mathsf{ewt}', \mathsf{lwt})$ over $\Sigma_{\mathcal{V}'}$ as follows. For each edge $e \in E$ of the form $(l, (a, \sigma), \phi, \lambda, l')$ with $(a, \sigma) \in \Sigma_{\mathcal{V}}$, for each $y \in \mathcal{V}' \backslash \mathcal{V}$ and for each $i \in \{0, 1\}$, there is an edge $e' \in E'$ of the form $(l, (a, \sigma'), \phi, \lambda, l')$ with $(a, \sigma') \in \Sigma_{\mathcal{V}'}$, where

$$\sigma'(z) = \begin{cases} \sigma(z) & \text{if } z \in \mathcal{V}, \\ i & \text{otherwise.} \end{cases}$$

Moreover, $\mathsf{ewt}'(e') = \mathsf{ewt}(e)$. There are no other edges in E'. Let $\mathcal{A}_{N_{\mathcal{V}'}}$ be a weighted timed automaton over $\Sigma_{\mathcal{V}'}$ with $\|\mathcal{A}_{N_{\mathcal{V}'}}\| = 1_{N_{\mathcal{V}'}}$ and $\mathsf{Func}(\mathcal{A}_{N_{\mathcal{V}'}}) = \{\mathbb{1}\}$. In this way, $\mathcal{A}_{N_{\mathcal{V}'}}$ is non-interfering with any weighted timed automaton. Now, let $\mathcal{A}_{\mathcal{V}'}$ be the product automaton of $\mathcal{A}'$ and $\mathcal{A}_{N_{\mathcal{V}'}}$ as defined in the proof of Lemma 6. It is easy to show that $\mathcal{A}_{\mathcal{V}'}$ satisfies conditions 1. to 3. ■

Example 23 Figures 3 and 4 show the weighted timed automata $\mathcal{A}_{f(y)}$ over $\Sigma_{\{y\}}$ and $\Sigma_{\{y,z\}}$, respectively. Note that taking the Hadamard product with $\mathcal{A}_{N_{\{y,z\}}}$ is necessary, because otherwise there are successful runs over timed words encoding *invalid* variable assignments. □

Now, we prove Theorem 21. For the induction base, we consider the atomic and weighted atomic formulas in $\mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.

Atomic and Weighted Atomic Formulas Let $\varphi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ be atomic. If φ equals $P_a(y)$, $y < z$, $y = z$, $y \in X$, or one of its negations, we can construct a timed automaton $\mathcal{A}'_\varphi$ using the same approach as for formulas in $\mathsf{MSO}(\Sigma)$ (see e.g. Thomas [26]). We define $\mathcal{A}_\varphi$ to be the weighted timed automaton obtained from $\mathcal{A}'_\varphi$ by adding weight functions ewt and lwt defined by $\mathsf{lwt}(l) = \mathbb{1}$ for each location l and $\mathsf{ewt}(e) = 1$ for each edge e. For φ not as above, the corresponding weighted timed automata $\mathcal{A}_\varphi$ are shown in Fig. 3. The idea behind the construction of $\mathcal{A}_{\overleftarrow{\mathsf{d}}(D,y)\sim c}$ is as follows: the automaton non-deterministically guesses when the last edge labeled with a letter in $\Sigma_{\{y,D\}}$ with a 1 in the D-row is taken and resets the clock variable at this edge. Then

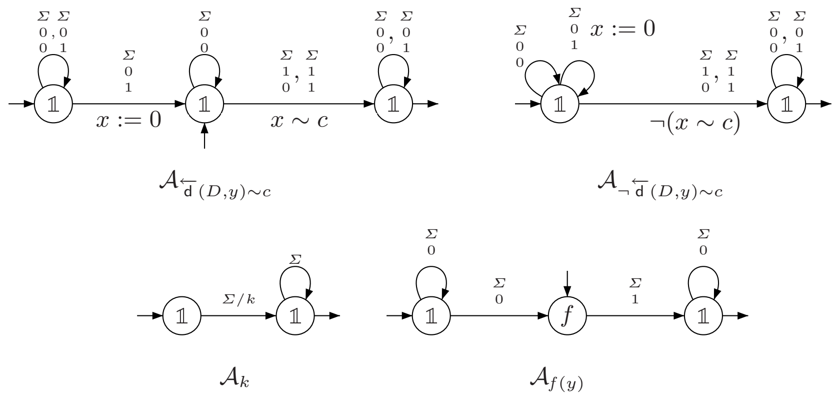

Fig. 3. Weighted timed automata recognizing (weighted) atomic formulas. Edge weights equal to 1 are omitted.

it verifies that whenever an edge is labeled by a letter with a 1 in the y-row, the time distance to the last event labeled with a letter with a 1 in the D-row satisfies $\sim c$. This can be done by adding a corresponding clock constraint to this edge. The idea for $\varphi = \neg \overleftarrow{\mathsf{d}}(D, y) \sim c$ is similar. The weighted timed automaton $\mathcal{A}_{f(y)}$ verifies that whenever an edge is labeled with a letter such that there is a 1 in the y-row, then the source location of this edge must be assigned the weight function f. All the other locations must be assigned the weight function $\mathbb{1}$. We further construct $\mathcal{A}_k$ in such a way that there is exactly one edge with cost k, and all other edges are assigned 1. Finally, it can be shown in a straightforward manner that conditions 2. and 3. of Theorem 21 are satisfied.

Remark 24 We remark that for e.g. the tropical semiring, there is no weighted timed automaton that corresponds to the formula k if we allow for empty timed words. For including empty timed words, one may extend the model of weighted timed automaton with weight functions assigning a weight for entering a location and leaving a location, see [23]. Then, the semantics of k are $\mathcal{F}$-recognizable.

For the induction step, we consider the operators of $\mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$. We start with disjunction and existential quantification.

Disjunction Let $\psi, \zeta \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ and assume $\varphi = \psi \vee \zeta$. Note that $\mathsf{Free}(\varphi) = \mathsf{Free}(\psi) \cup \mathsf{Free}(\zeta)$ and thus $\mathsf{Free}(\psi) \subseteq \mathsf{Free}(\varphi)$ and $\mathsf{Free}(\zeta) \subseteq \mathsf{Free}(\varphi)$. By induction hypothesis, there are weighted timed automata $\mathcal{A}_\psi$ over $\Sigma_{\mathsf{Free}(\varphi)}$ and $\mathcal{A}_\zeta$ over $\Sigma_{\mathsf{Free}(\varphi)}$, respectively, satisfying condition 1. to 3. of Theorem 21. Let $\mathcal{A}_\varphi$ be the disjoint union of $\mathcal{A}_\psi$ and $\mathcal{A}_\zeta$. We have $\|\mathcal{A}_\varphi\| = \|\mathcal{A}_\psi\| + \|\mathcal{A}_\zeta\|$ and thus $\|\mathcal{A}_\varphi\| = [\![\varphi]\!]$. Clearly, also conditions 2. and 3. hold.

Existential quantification Let $\psi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ and assume $\varphi = \exists y.\psi$. We further let $\mathcal{V} = \mathsf{Free}(\varphi)$ and $\mathcal{V}' = \mathcal{V} \cup \{y\} = \mathsf{Free}(\psi)$. By induction hypothesis, there is a weighted timed automaton $\mathcal{A}_\psi$ over $\Sigma_{\mathcal{V}'}$ satisfying conditions 1. to 3. of Theorem 21. Let $p : \Sigma_{\mathcal{V}'} \to \Sigma_{\mathcal{V}}$ be the projection that simply erases the y-row. Let $\mathcal{A}_\varphi$ be the weighted timed automaton over $\Sigma_{\mathcal{V}}$ obtained from $\mathcal{A}_\psi$ as defined in the proof of Lemma 7. Hence, we have $\|\mathcal{A}_\varphi\| = \bar{p}(\|\mathcal{A}_\psi\|)$. However, for each $((\bar{a}, \sigma), \bar{t}) \in T(\Sigma_{\mathcal{V}})^+$, we also have

$$\begin{aligned}
&\big(\bar{p}(\|\mathcal{A}_\psi\|), ((\bar{a}, \sigma), \bar{t})\big) \\
&= \big(\bar{p}([\![\psi]\!]_{\mathcal{V}\cup\{y\}}), ((\bar{a}, \sigma), \bar{t})\big) \\
&= \sum_{\substack{((\bar{a},\sigma'),\bar{t}) \in T(\Sigma_{\mathcal{V}\cup\{y\}})^+ \\ p((\bar{a},\sigma'),\bar{t}) = ((\bar{a},\sigma),\bar{t})}} \big([\![\psi]\!]_{\mathcal{V}\cup\{y\}}, ((\bar{a}, \sigma'), \bar{t})\big) \\
&\stackrel{\star}{=} \sum_{i \in \mathsf{dom}(\bar{a},\bar{t})} \big([\![\psi]\!]_{\mathcal{V}\cup\{y\}}, ((\bar{a}, \sigma[y \to i], \bar{t})\big) \\
&= \big([\![\exists y.\psi]\!]_{\mathcal{V}}, ((\bar{a}, \sigma), \bar{t})\big)
\end{aligned}$$

where * uses the equivalences

$$p\big((\bar{a}, \sigma'), \bar{t}\big) = ((\bar{a}, \sigma), \bar{t}) \quad \Leftrightarrow \quad \sigma' = \sigma[y \to i] \text{ for some } i \in \mathsf{dom}(\bar{a}, \bar{t})$$

and

$$\sigma \text{ is a valid } (\mathcal{V}, (\bar{a}, \bar{t}))\text{-assignment}$$

$$\Leftrightarrow$$

$$\sigma[y \to i] \text{ is a valid } (\mathcal{V} \cup \{y\}, (\bar{a}, \bar{t}))\text{-assignment for every } i \in \mathsf{dom}(\bar{a}, \bar{t}).$$

Hence, $\|\mathcal{A}_\varphi\| = [\![\varphi]\!]$. It is obvious that condition 2. is satisfied. For showing condition 3., let $f(z)$ be a subformula occuring in φ with $z \in \mathsf{Free}(\varphi)$. Thus, $z \neq y$. Since $\mathcal{A}_\varphi$ is obtained from $\mathcal{A}_\psi$ by only removing the y-row from the labels of all edges, condition 3. is satisfied.

The proof for the case $\varphi = \exists X.\psi$ can be done analogously.

Before we come to the case of conjunction, we consider the following example.

Example 25 Consider the formula $f(y) \wedge g(z)$. Notice that $\mathcal{A}_{f(y)}$ and $\mathcal{A}_{g(z)}$ over $\Sigma_{\{y,z\}}$ are *not* non-interfering: in both automata, there is a run from a location which is assigned the weight function f (or g, respectively) to a final location. This is due to common labels of the form $(a, 1, 1)$, which represent the fact that y and z may be assigned the same position in a timed word (see the uppermost part of $\mathcal{A}_{f(y)}$ in Figure 4). We eliminate this by adding the formula $\neg(y = z)$. This can be seen in the right automaton in Figure 4. The formula $f(y) \wedge g(z) \wedge \neg(y = z)$ is in $\mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, whereas the formula $f(y) \wedge g(z)$ is not.

□

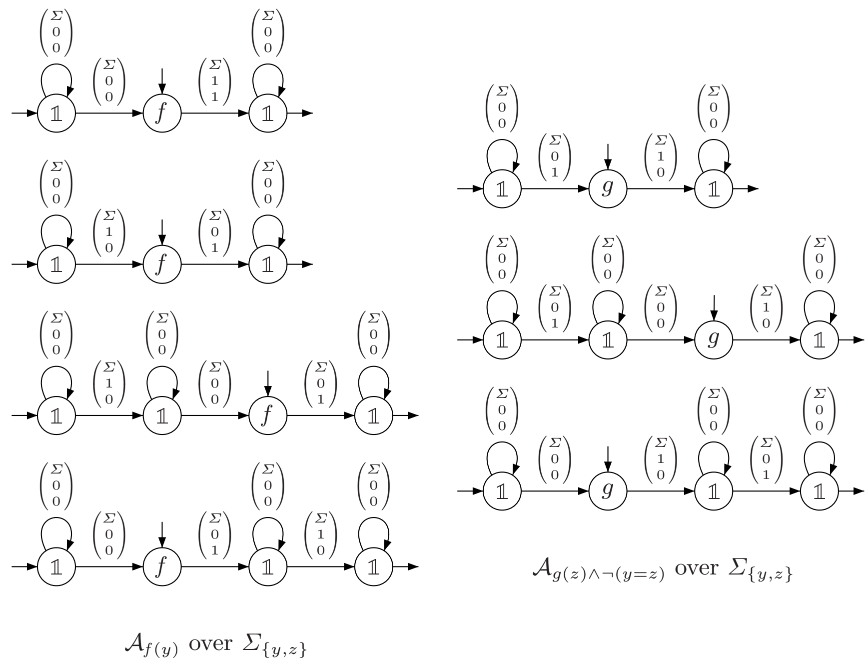

Fig. 4. Two non-interfering weighted timed automata. Edge weights equal to 1 are omitted.

Conjunction Let $\psi, \zeta \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ and assume that one of the following conditions hold:

- $\mathcal{V}_f(\psi) = \emptyset$,
- $\mathcal{V}_f(\zeta) = \emptyset$,
- $\mathcal{V}_f(\psi) \cap \mathcal{V}_f(\zeta) = \emptyset$, $\mathcal{V}_f(\psi) \subseteq \mathsf{Free}(\psi)$, and $\mathcal{V}_f(\zeta) \subseteq \mathsf{Free}(\zeta)$,

Further assume $\varphi = \psi \wedge \zeta \wedge \bigwedge_{\substack{y \in \mathcal{V}_f(\psi), z \in \mathcal{V}_f(\zeta) \\ y \neq z}} \neg(y = z)$.

Note that $\mathsf{Free}(\varphi) = \mathsf{Free}(\psi) \cup \mathsf{Free}(\zeta)$ and thus $\mathsf{Free}(\psi) \subseteq \mathsf{Free}(\varphi)$ and $\mathsf{Free}(\zeta) \subseteq \mathsf{Free}(\varphi)$. By induction hypothesis, there are weighted timed automata $\mathcal{A}_\psi$ over $\Sigma_{\mathsf{Free}(\varphi)}$ and $\mathcal{A}_\zeta$ over $\Sigma_{\mathsf{Free}(\varphi)}$ satisfying conditions 1. to 3. of Theorem 21.

We now distinguish between three cases.

(Case 1) We assume that $\mathcal{V}_f(\psi) = \emptyset$. Hence, $\mathsf{Func}(\psi) = \emptyset$. By induction hypothesis, we thus have $\mathsf{Func}(\mathcal{A}_\psi) = \{\mathbb{1}\}$. This implies that $\mathcal{A}_\psi$ is non-interfering with $\mathcal{A}_\zeta$. We let $\mathcal{A}_\varphi$ be the product automaton of $\mathcal{A}_\psi$ and $\mathcal{A}_\zeta$ as defined in the proof of Lemma 6. Notice that we are allowed to apply this lemma since we assume $\mathcal{K}$ to be commutative. Hence, we have $\|\mathcal{A}_\varphi\| = \|\mathcal{A}_\psi\| \odot \|\mathcal{A}_\zeta\|$ and thus $\|\mathcal{A}_\varphi\| = [\![\varphi]\!]$. It is straightforward to show that $\mathcal{A}_\varphi$ also satisfies conditions 2. and 3.

(Case 2) We assume that $\mathcal{V}_f(\zeta) = \emptyset$. This case be done analogously to case 1.

(Case 3) Assume that both $\mathcal{V}_f(\psi) \neq \emptyset$ and $\mathcal{V}_f(\zeta) \neq \emptyset$, and thus we may assume

(a) $\mathcal{V}_f(\psi) \cap \mathcal{V}_f(\zeta) = \emptyset$,
(b) $\mathcal{V}_f(\psi) \subseteq \mathsf{Free}(\psi)$, and
(c) $\mathcal{V}_f(\zeta) \subseteq \mathsf{Free}(\zeta)$.

Let $\chi = \bigwedge_{\substack{y \in \mathcal{V}_f(\psi), z \in \mathcal{V}_f(\zeta) \\ y \neq z}} \neg(y = z)$ and put $\varrho = \zeta \wedge \chi$. Since $\mathcal{V}_f(\chi) = \emptyset$, the conditions of case 2 are satisfied and hence there is a weighted timed automaton $\mathcal{A}_\varrho$ over $\Sigma_{\mathsf{Free}(\varphi)}$ satisfying conditions 1. to 3.

Next, we show that $\mathcal{A}_\psi$ and $\mathcal{A}_\varrho$ are non-interfering. For this, let l be a location in $\mathcal{A}_\psi$ such that $\mathsf{lwt}_\psi(l) = f$ for some $f \in \mathcal{F}$. By condition 2. of Theorem 21, there is some subformula $f(y)$ occurring in ψ for some first-order variable y. By (b), we know that $y \in \mathsf{Free}(\psi)$, and thus by condition 3. of Theorem 21, for each edge $(l, (a, \sigma), \phi, \lambda, l_1)$ in $\mathcal{A}_\psi$ we have $\sigma(y) = 1$.

Now, let l' be a location in $\mathcal{A}_\varrho$ such that $\mathsf{lwt}_\varrho(l') = f'$ for some $f' \in \mathcal{F}$. By condition 2. of Theorem 21, there is some subformula $f'(z)$ occurring in ϱ for some first-order variable z. Clearly, by definition of ϱ, this subformula $f'(z)$ can only occur in ζ. Let $(l', (b, \sigma), \phi', \lambda', l_2)$ be an edge of $\mathcal{A}_\varrho$. By (b), we have $z \in \mathsf{Free}(\zeta) \subseteq \mathsf{Free}(\varrho)$, and thus by condition 3. of Theorem 21, we have $\sigma(z) = 1$. We further know by (a) that $y \neq z$, which implies $\sigma(y) = 0$. From this it follows that for l and l', there is no edge labeled with a common letter in $\Sigma_{\mathsf{Free}(\varphi)}$. Hence, from (l, l') there is no run into $\mathcal{L}_f^\psi \times \mathcal{L}_f^\rho$, and thus $\mathcal{A}_\psi$ and $\mathcal{A}_\varrho$ are non-interfering.

Finally, let $\mathcal{A}_\varphi$ be the product automaton of $\mathcal{A}_\psi$ and $\mathcal{A}_\varrho$ as defined in the proof of Lemma 6. Clearly, we have

$$\|\mathcal{A}_\varphi\| = [\![\psi \wedge \zeta \wedge \bigwedge_{\substack{y \in \mathcal{V}_f(\psi), z \in \mathcal{V}_f(\zeta) \\ y \neq z}} \neg(y = z)]\!].$$

It is straightforward to show that conditions 2. and 3. also hold.

Second-Order Universal Quantification Now, let $\psi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ be unweighted and assume $\varphi = \forall X.\psi$. Clearly, φ is also unweighted. By Lemma 16, $[\![\varphi]\!] = 1_{L(\varphi)}$. By Theorem 10, there is a timed automaton $\mathcal{A}$ such that $L(\mathcal{A}) = L(\varphi)$. Let $\mathcal{A}_\varphi$ be the weighted timed automaton obtained from $\mathcal{A}$ as defined in the proof of Lemma 3.2. Then $\mathcal{A}_\varphi$ satisfies conditions 1. to 3. of Theorem 21.

Before we come to first-order universal quantification, we introduce a normalization technique and some notations.

Lemma 26 *For every TA-recognizable timed language $L \subseteq T\Sigma^+$, there is a timed automaton $\mathcal{A}'$ such that $L(\mathcal{A}') = L$, and for each location l in $\mathcal{A}'$ there is a unique $a \in \Sigma$ such that every edge $(l, a', \phi, \lambda, l')$ in $\mathcal{A}'$ satisfies $a' = a$.*

Proof. Let $L \subseteq T\Sigma^+$ be TA-recognizable over Σ. Then there is a timed automaton $\mathcal{A} = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E)$ such that $L(\mathcal{A}) = L$. Define $\mathcal{A}' = (\mathcal{L}', \mathcal{L}'_0, \mathcal{L}'_f, \mathcal{C}, E')$, where

- $\mathcal{L}' = \mathcal{L} \times \Sigma$,
- $\mathcal{L}'_0 = \mathcal{L}_0 \times \Sigma$,
- $\mathcal{L}'_f = \mathcal{L}_f \times \Sigma$,
- $E' = \{((l, a), a, \phi, \lambda, (l', a')) : (l, a, \phi, \lambda, l') \in E, a' \in \Sigma\}$.

Then we have $L(\mathcal{A}') = L(\mathcal{A})$, which can be proved in a straightforward way. ■

Let $n \in \mathbb{N} \backslash \{0\}$. We define $\Sigma^{(n)} = \Sigma \times \{1, ..., n\}$. Similarly to timed words over the extended alphabet $\Sigma_{\mathcal{V}}$ for some finite set $\mathcal{V}$ of variables, we write $((\bar{a}, \mu), \bar{t})$ to denote a timed word over $\Sigma^{(n)}$, where $(\bar{a}, \bar{t}) \in T\Sigma^+$ and $\mu \in \{1, ..., n\}^{\mathsf{dom}(\bar{a}, \bar{t})}$. We define for every $\xi \in \mathsf{MSO}(T\Sigma^+)$ the formula $\widetilde{\xi} \in \mathsf{MSO}(T(\Sigma^{(n)})^+)$ by replacing in ξ every occurence of $P_a(y)$ by $\bigvee_{1 \leq j \leq n} P_{(a,j)}(y)$. The next lemma can be proved by induction over the structure of ξ.

Lemma 27 *Let $\xi \in \mathsf{MSO}(T\Sigma^+)$ and $\mathcal{V} \supseteq \mathsf{Free}(\xi)$. Then for every $((\bar{a}, \mu, \sigma), \bar{t}) \in T((\Sigma^{(n)})_{\mathcal{V}})^+$ with $((\bar{a}, \sigma), \bar{t}) \in N_{\mathcal{V}}$ we have*

$$((\bar{a}, \sigma), \bar{t}) \models \xi \quad \Leftrightarrow \quad ((\bar{a}, \mu, \sigma), \bar{t}) \models \widetilde{\xi}.$$

First-Order Universal Quantification Let $\psi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ be almost unambiguous over y and assume $\varphi = \forall y.\psi$. By Lemma 19, we may assume that ψ is of the form

$$\psi = \bigvee_{1 \leq j \leq n} f_j(y) \wedge k_j \wedge \psi_j$$

where $n \in \mathbb{N}$, $k_j \in K$, $f_j \in \mathcal{F}$, unweighted $\psi_j \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ for each $j \in \{1, ..., n\}$.

Let $\mathcal{W} = \mathsf{Free}(\psi)$ and $\mathcal{V} = \mathsf{Free}(\varphi) = \mathcal{W} \backslash \{y\}$. Recall that $\psi_1, ..., \psi_n$ can be considered as formulas in $\mathsf{MSO}(T\Sigma^+)$. We may assume that $\psi_1, ..., \psi_n$ define a partition of $N_{\mathcal{W}}$. We define $\widetilde{L} \subseteq T((\Sigma^{(n)})_{\mathcal{V}})^+$ to be the set of timed words $((\bar{a}, \mu, \sigma), \bar{t})$ in $T((\Sigma^{(n)})_{\mathcal{V}})^+$ such that $((\bar{a}, \sigma), \bar{t}) \in N_{\mathcal{V}}$, and for all $i \in \mathsf{dom}(\bar{a}, \bar{t})$ and $j \in \{1, ..., n\}$ we have

$$\mu(i) = j \text{ implies } ((\bar{a}, \sigma[y \to i]), \bar{t}) \models \psi_j.$$

Notice that for every $((\bar{a}, \sigma), \bar{t}) \in N_{\mathcal{V}}$, there is a unique μ such that $((\bar{a}, \mu, \sigma), \bar{t}) \in \widetilde{L}$, since $(\psi_1, ..., \psi_n)$ forms a partition of $N_{\mathcal{W}}$. Next, we prove that $\widetilde{L}$ is TA-recognizable. For this, consider the formula $\zeta \in \mathsf{MSO}(T(\Sigma^{(n)})^+)$

$$\zeta = \forall y. \bigwedge_{1 \leq j \leq n} \bigwedge_{a \in \Sigma} \left(P_{(a,j)}(y) \longrightarrow \widetilde{\psi_j} \right).$$

Let $((\bar{a}, \mu, \sigma), \bar{t}) \in T((\Sigma^{(n)})_{\mathcal{V}})^+$ such that $((\bar{a}, \sigma), \bar{t}) \in N_{\mathcal{V}}$. Using the semantics of $\mathsf{MSO}(T(\Sigma^{(n)})^+)$, one can show that $((\bar{a}, \mu, \sigma), \bar{t}) \models \zeta$ if and only if for every $i \in \mathsf{dom}(\bar{a}, \bar{t})$ and $j \in \{1, ..., n\}$ we have that $\mu(i) = j$ implies $((\bar{a}, \mu, \sigma[y \to i]), \bar{t}) \models \widetilde{\psi_j}$. This, by Lemma 27, holds if and only if $((\bar{a}, \sigma[y \to i]), \bar{t}) \models \psi_j$.

Thus, $((\bar{a},\mu,\sigma),\bar{t}) \models \zeta$ if and only if $((\bar{a},\mu,\sigma),\bar{t}) \in \widetilde{L}$, and we have $L(\zeta) = \widetilde{L}$. By Theorem 10, $\widetilde{L}$ is TA-recognizable over $(\Sigma^{(n)})_{\mathcal{V}}$.

Next, we will use the information encoded in μ to build a weighted timed automaton over $\mathcal{K}$, $(\Sigma^{(n)})_{\mathcal{V}}$ and $\mathcal{F}$. Let $\widetilde{\mathcal{A}} = (\mathcal{L},\mathcal{L}_0,\mathcal{L}_f,\mathcal{C},E)$ be a timed automaton such that $L(\widetilde{\mathcal{A}}) = \widetilde{L}$. By Lemma 26, there is a timed automaton $\mathcal{A}' = (\mathcal{L}',\mathcal{L}'_0,\mathcal{L}'_f,\mathcal{C},E')$ such that $L(\mathcal{A}') = L(\widetilde{\mathcal{A}})$, the locations in $\mathcal{A}'$ are elements in $\mathcal{L}\times(\Sigma^{(n)})_{\mathcal{V}}$, and for each $(l,(a,b,\sigma)) \in \mathcal{L}'$, every outgoing edge is labeled with (a,b,σ). Observe that this latter fact is crucial for assigning the weight functions to the locations in a proper way. Now define $\mathcal{A} = (\mathcal{L}',\mathcal{L}_0,\mathcal{L}_f,\mathcal{C},E',\mathsf{ewt},\mathsf{lwt})$ by

- $\mathsf{ewt}\big((l,(a,b,\sigma)),(a,b,\sigma),\phi,\lambda,(l',(a',b',\sigma'))\big) = k_b$ for each $\big((l,(a,b,\sigma)),(a,b,\sigma),\phi,\lambda,(l',(a',b',\sigma'))\big) \in E'$,
- $\mathsf{lwt}\big((l,(a,b,\sigma))\big) = f_b$ for every $(l,(a,b,\sigma)) \in \mathcal{L}'$.

Note that $\mathsf{Func}(\mathcal{A}) = \{f_1,...,f_n\}$. We also observe that for each $w = ((\bar{a},\mu,\sigma),\bar{t}) \in T((\Sigma^{(n)})_{\mathcal{V}})^+$, and for each run r of $\mathcal{A}$ on w with $\mathsf{rwt}(r) \neq 0$ we have

$$\mathsf{rwt}(r) = \prod_{i\in\mathsf{dom}(\bar{a},\bar{t})} f_{\mu(i)}(t_i - t_{i-1}) \cdot k_{\mu(i)}. \tag{1}$$

Consider the renaming $p : (\Sigma^{(n)})_{\mathcal{V}} \to \Sigma_{\mathcal{V}}$ defined by $(a,b,\sigma) \mapsto (a,\sigma)$ for each $(a,b,\sigma) \in (\Sigma^{(n)})_{\mathcal{V}}$. We show that $\bar{p}(\|\mathcal{A}\|) = [\![\forall y.\psi]\!]$. First, for every $((\bar{a},\sigma),\bar{t}) \in N_{\mathcal{V}}$ and the unique μ such that $((\bar{a},\mu,\sigma),\bar{t}) \in \widetilde{L}$, we have

$$\begin{aligned}
\big(\bar{p}(\|\mathcal{A}\|),((\bar{a},\sigma),\bar{t})\big) &= \big(\|\mathcal{A}\|,((\bar{a},\mu,\sigma),\bar{t})\big)\\
&\overset{\star}{=} \prod_{i\in\mathsf{dom}(\bar{a},\bar{t})} f_{\mu(i)}(t_i - t_{i-1}) \cdot k_{\mu(i)}\\
&= \prod_{i\in\mathsf{dom}(\bar{a},\bar{t})} \big([\![\varphi]\!]_{\mathcal{W}},((\bar{a},\sigma[y\to i]),\bar{t})\big)\\
&= \big([\![\forall y.\varphi]\!],((\bar{a},\sigma),\bar{t})\big)
\end{aligned}$$

where $\star$ is due to (1) and idempotence of $\mathcal{K}$. For every $((\bar{a},\sigma),\bar{t}) \notin N_{\mathcal{V}}$, we obtain 0 for both $\big(\bar{p}(\|\mathcal{A}\|),((\bar{a},\sigma),\bar{t})\big)$ and $\big([\![\forall y.\varphi]\!],((\bar{a},\sigma),\bar{t})\big)$. Thus, $\bar{p}(\mathcal{A}) = [\![\forall y.\psi]\!]$.

Finally, let $\mathcal{A}_\varphi$ be the weighted timed automaton over $\Sigma_{\mathcal{V}}$ obtained from $\mathcal{A}$ as defined in the proof of Lemma 7. Hence, we have $\|\mathcal{A}_\varphi\| = \bar{p}(\|\mathcal{A}\|) = [\![\varphi]\!]$. Since the construction of $\mathcal{A}_\varphi$ according to Lemma 7 does not add any location weight functions, condition 2. is satisfied. Condition 3 is trivially satisfied, since the set of subformulas $f(z)$ occurring in φ with $z \in \mathsf{Free}(\varphi)$ is empty. This finishes the proof of Theorem 21. ■

We proved that each formula $\varphi \in \mathsf{sRMSO}(\mathcal{K},T\Sigma^+,\mathcal{F})$ is recognizable by a weighted timed automaton. Now, we give the definition of the **syntactically restricted weighted relative distance logic**, denoted by $\mathsf{sR}\mathcal{L}\overleftarrow{\mathsf{d}}(\mathcal{K},\Sigma,\mathcal{F})$. It is defined as the smallest class of formulas containing all formulas generated by the next two rules.

1. If $\varphi \in \mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, then $\varphi \in \mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$.
2. If $\varphi \in \mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$, then $\exists D.\varphi \in \mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$.

Altogether, using the same lines of argumentation as in the proof of Theorem 21 in the case of existential quantification, we can show that if the semantics of $\varphi \in \mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$ is $\mathcal{F}$-recognizable over $\Sigma_{\mathsf{Free}(\varphi)}$, so is the semantics of $\exists D.\varphi$ $\mathcal{F}$-recognizable over $\Sigma_{\mathsf{Free}(\exists D.\varphi)}$. Altogether, we obtain the following theorem, which corresponds to one direction of a Büchi theorem for the class of $\mathcal{F}$-recognizable timed series.

Theorem 28 *Let $\mathcal{K}$ be idempotent and commutative. If $\varphi \in \mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$, then $[\![\varphi]\!]$ is $\mathcal{F}$-recognizable over $\Sigma_{\mathsf{Free}(\varphi)}$.*

We remark that the transformations can be done effectively provided that the operations of $\mathcal{K}$ and $\mathcal{F}$ are given effectively. The only critical point in the proof is the construction of a weighted timed automaton recognizing $[\![\forall y.\varphi]\!]$ if φ is almost unambiguous. However, by Lemma 19 we can transform each almost unambiguous formula into the form $\bigvee_{1\leq i\leq n} f_i(y) \wedge k_i \wedge \psi_i$ with $f_i \in \mathcal{F}$, $k_i \in K$ and unweighted $\psi_i \in \mathsf{MSO}(T\Sigma^+)$ for each $i \in \{1, ..., n\}$ and some $n \in \mathbb{N}$, as it is required in the corresponding construction.

4 From Weighted Timed Automata to Logic

In this section, we show that the behaviour of each weighted timed automaton can be defined by a sentence in $\mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$. For this, we extend the proof proposed by Droste and Gastin [13] to the timed setting.

Theorem 29 *Let $\mathcal{K}$ be idempotent and commutative. Each $\mathcal{F}$-recognizable timed series is $\mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$-definable.*

Proof. Let $\mathcal{A} = (\mathcal{L}, \mathcal{L}_0, \mathcal{L}_f, \mathcal{C}, E, \mathsf{wt})$ be a weighted timed automaton over $\mathcal{K}$, Σ and $\mathcal{F}$. We choose an enumeration $(x_1, ..., x_m)$ of $\mathcal{C}$ together with an enumeration $(e_1, ..., e_n)$ of E and assume $e_i = (l_i, a_i, \phi_i, \lambda_i, l'_i)$. Let $\bar{D} = D_1, ..., D_m$, where D_i stands for the clock variable x_i for each $i \in \{1, ..., m\}$, and let $\bar{Y} = Y_1, ..., Y_n$, where Y_j stands for the edge e_j for each $j \in \{1, ..., n\}$. Intuitively, D_i stores all the positions in a timed word where the clock variable x_i has been reset, and Y_i stores all the positions in a timed word that have been reached by executing e_i. We define an *unweighted* formula $\psi(\bar{D}, \bar{Y})$ describing the successful runs of $\mathcal{A}$. This can be done similarly to the unweighted and untimed settings, respectively [29, 13]. For instance, the clock constraints at the edges may be defined by the formula

$$\psi_{test} := \forall y. \bigwedge_{1\leq i\leq n} \left(y \in Y_i \longrightarrow \bigwedge_{(x_j \sim c)\in\phi_i} \overleftarrow{\mathsf{d}}(D_j, y) \sim c \right)$$

and the resets of the clock constraints can be defined by

$$\psi_{reset} := \bigwedge_{1\le i\le m} \forall y. \left(y \in D_i \longleftrightarrow \bigvee_{\substack{1\le j\le n\\ x_i\in\lambda_j}} y \in Y_j \right),$$

see [28] for further explanations. Then, for every timed word $(\bar{a},\bar{t})$ and valid $(\{\bar{D},\bar{Y}\},(\bar{a},\bar{t}))$-assignment σ, we have $[\![\psi(\bar{D},\bar{Y})]\!]((\bar{a},\sigma),\bar{t}) = 1$ if there is a successful run of $\mathcal{A}$ on $(\bar{a},\bar{t})$, and $[\![\psi(\bar{D},\bar{Y})]\!]((\bar{a},\sigma),\bar{t}) = 0$ otherwise. This can be proved similar to [13] by showing that there is a bijective correspondence between the set of successful runs and the set of $(\mathcal{V},w)$-assignments as above.

Now, we "add weights" to $\psi(\bar{D},\bar{Y})$ to obtain a formula $\xi(\bar{D},\bar{Y})$ whose semantics corresponds to the running weight of a successful run of $\mathcal{A}$ on $(\bar{a},\bar{t})$. Define

$$\xi(\bar{D},\bar{Y}) = \psi(\bar{D},\bar{Y}) \wedge \bigwedge_{e_i\in E} \forall y.(\neg(y\in Y_i) \vee [y\in Y_i \wedge \mathsf{lwt}(l_i)(y) \wedge \mathsf{ewt}(e_i)]).$$

Finally, let $\zeta = \exists D_1...\exists D_m \exists Y_1...\exists Y_n.\xi(\bar{D},\bar{Y})$. Using similar methods as in the untimed setting [13], we obtain $[\![\zeta]\!] = \|\mathcal{A}\|$.

As a consequence of the results presented in this and the previous section, we obtain a Büchi theorem for the class of $\mathcal{F}$-recognizable timed series over idempotent and commutative semirings.

Theorem 30 *Let $\mathcal{K}$ be commutative and idempotent and let $\mathcal{F}$ contain $\mathbb{1}$. Each timed series $\mathcal{T} : T\Sigma^+ \to K$ is $\mathcal{F}$-recognizable if and only if $\mathcal{T}$ is $\mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}\,(\mathcal{K},\Sigma,\mathcal{F})$-definable.*

5 Generalizations to Arbitrary Semirings

In this section, we explain how we can generalize Theorem 30 to non-idempotent semirings. Later on, we will indicate how we skip the restriction on the semiring being commutative.

Let $\mathcal{K}$ be a commutative semiring, not necessarily being idempotent.

In the last section, we used the idempotence of $\mathcal{K}$ in two crucial steps. First, in the proof of Lemma 16, where we showed that each unweighted formula $\psi \in \mathsf{MSO}(\mathcal{K},T\Sigma^+,\mathcal{F})$ takes only values in $\{0,1\}$. This no longer holds if $\mathcal{K}$ is not idempotent. We thus cannot use Lemma 16 to show Lemmas 18 and 19. Notice that if we exclude the usage of disjunction and existential quantification in an unweighted formula $\varphi \in \mathsf{MSO}(\mathcal{K},T\Sigma^+,\mathcal{F})$, then the semantics of φ takes only values in $\{0,1\}$. However, not every unweighted formula can be transformed into a language equivalent such formula, due to the syntactical restriction on negation. Instead, we will introduce *syntactically unambiguous formulas.*

Second, we used the idempotence of $\mathcal{K}$ in the proof of Theorem 21 in the case of universal first-order quantification. We used that for each timed word w, the running weights of all successful runs of $\mathcal{A}$ on w are the same, and thus, by idempotence of $\mathcal{K}$, the behaviour of $\mathcal{A}$ on w is the same as the running weight of an arbitrary successful run of $\mathcal{A}$ on w. Notice that if $\mathcal{A}$ is deterministic, i.e., there is only one successful run of $\mathcal{A}$ on w, then the behaviour of $\mathcal{A}$ on w is also the same as the running weight of an arbitrary successful (i.e., the only) run. However, as we have noted in Sect. 1, the class of deterministically TA-recognizable timed languages is a strict subclass of TA-recognizable timed languages. For this reason, we focus on a subclass of timed languages whose elements have a *bounded variability*. Then we take advantage of the fact that every such timed language is deterministically TA-recognizable.

The notion of bounded variability of timed words has been introduced by Wilke [29]. Intuitively, the variability of a timed word corresponds to the maximum number of events that may occur within one time unit. When bounding the variability of timed words, we can always construct deterministic timed automata. Using this, Wilke showed that $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$ is fully decidable over the class of timed languages with bounded variability (as opposed to the class of all timed languages) [29]. The restriction to timed languages with bounded variability is a reasonable assumption as *practically* any system can only handle a bounded number of tasks within a time unit. Recently, another positive decidability result concerning MTL model checking was shown for this particular class of timed languages [18].

Let $M \subseteq T\Sigma^+$ be a set of timed words. We say that $L \subseteq T\Sigma^+$ is *TA-recognizable over Σ relatively to M* if there is a timed automaton $\mathcal{A}$ over Σ such that $L = L(\mathcal{A}) \cap M$. Let $w = (a_1, t_1)...(a_k, t_k) \in T\Sigma^+$. The *variability* of w, denoted by $var(w)$, is defined as $\sup\{b + 1 : \exists i.1 \leq i \leq k - b \text{ and } t_{i+b} - t_i < 1\}$. Intuitively, the variability of a timed word gives the maximum number of events in a unit time interval. Let $b \in \mathbb{N}$. We say that w is of bounded variability b if the variability of w is less than or equal to b. We use $T_b\Sigma^+$ to denote the set $\{w \in T\Sigma^+ : var(w) \leq b\}$ of all timed words of bounded variability b. By bounding the variability of a timed word we fix the maximum number of events in a unit time interval.

Remark 31 In the literature, there are also other restrictions on the occurence of events within timed words, the most known of which is the restriction of being *non-Zeno*. A timed word is non-Zeno if the sequence of timestamps of the word is diverging. Hence, every finite word is non-Zeno and thus this notion is weaker than that of bounded variability. The restriction of being non-Berkeley for some positive real number δ has been introduced by Furia and Rossi [18] and means that between any two events more than δ time units must pass. For a comparison between these three restrictions see the paper of Furia and Rossi.

In the following, we fix a bound $b \in \mathbb{N}$.

Proposition 32 ([28]) *1. If $L \subseteq T\Sigma^+$ is TA-recognizable over Σ, we can effectively construct a deterministic timed automaton $\mathcal{A}$ over Σ such that $L(\mathcal{A}) = L \cap T_b\Sigma^+$.*

2. *The class of TA-recognizable timed languages over Σ relatively to $T_b\Sigma^+$ is closed under boolean operations, renamings and inverse renamings.*
3. *The set $T_b\Sigma^+$ is $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$-definable.*

We let $\exists D_1 ... \exists D_b.\varphi_b$ denote a sentence in $\mathcal{L}\overleftarrow{\mathsf{d}}(\Sigma)$ defining $T_b\Sigma^+$. For instance, φ_b may be the formula

$$\varphi_b = \begin{pmatrix} (1 \in D_1 \wedge 2 \in D_2 \wedge ... \wedge b \in D_b) \\ \wedge \\ \bigwedge_{1 \le i \le b} \forall y.(y \in D_i \longleftrightarrow (y+b) \in D_i) \\ \wedge \\ \bigwedge_{1 \le i \le b} \forall y.(y \in D_i \longrightarrow \overleftarrow{\mathsf{d}}(D_i, y) < 1) \end{pmatrix}$$

where $1, 2, ..., b$ stand for the first, second,..., b-th position in w, and $(y + b)$ stands for the b-th position in w after y. These terms can easily expressed in $\mathsf{MSO}(T\Sigma^+)$.

In the following, we define *syntactically unambiguous formulas.* Recall that if the semiring is idempotent, each unweighted formula evaluates each timed word to either 0 or 1. This is no longer the case if the semiring is not idempotent. For instance, the formula $\exists y.P_a(y)$ over the semiring of the natural numbers *counts* the number of a's in a timed word. However, we will show that each unweighted formula $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ can be transformed into a language equivalent formula ψ that has at most one assignment evaluating a timed word to 1, and thus ψ evaluates each timed word to either 0 or 1 even if the semiring is not idempotent. Since in general there may be more than one such assignment, we choose the *first* such assignment, in the following sense: if y is a free first-order variable, then we choose the smallest position with a one in the y-row; if X is a free second-order variable, then we choose the set of the smallest positions with a one in the X-row.

Let $\varphi, \xi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ be unweighted. We define the formulas φ^+, φ^-, $\varphi \xrightarrow{+} \xi$ and $\varphi \xleftrightarrow{+} \xi$ inductively as follows.

1. If φ is of the form $P_a(y)$, $y < z$, $y = z$, $y \in X$, $\overleftarrow{\mathsf{d}}(D, y) \sim c$, then $\varphi^+ = \varphi$ and $\varphi^- = \neg\varphi$.
2. If $\varphi = \neg\psi$, then $\varphi^+ = \psi^-$ and $\varphi^- = \psi^+$.
3. If $\varphi = \psi \vee \zeta$, then $\varphi^+ = \psi^+ \vee (\psi^- \wedge \zeta^+)$ and $\varphi^- = \psi^- \wedge \zeta^-$.
4. If $\varphi = \psi \wedge \zeta$, then $\varphi^- = \psi^- \vee (\psi^+ \wedge \zeta^-)$ and $\varphi^+ = \psi^+ \wedge \zeta^+$.
5. If $\varphi = \exists y.\psi$, then $\varphi^+ = \exists y.(\psi^+(y) \wedge \forall z.(z < y \wedge \psi(z))^-)$ and $\varphi^- = \forall y.\psi^-$.
6. If $\varphi = \forall y.\psi$, then $\varphi^- = \exists y.(\psi^-(y) \wedge \forall z.(y \le z \vee \psi(z))^+)$ and $\varphi^+ = \forall y.\psi^+$.
7. $\varphi \xrightarrow{+} \xi = \varphi^- \vee (\varphi^+ \wedge \xi^+)$
8. $\varphi \xleftrightarrow{+} \xi = (\varphi^+ \wedge \xi^+) \vee (\varphi^- \wedge \xi^-)$
9. For second-order variables X, Y, we define

$$X = Y = \forall y.(y \in X \xleftrightarrow{+} y \in Y),$$
$$X < Y = \exists y.(y \in Y \wedge \neg(y \in X) \wedge \forall z.[z < y \xrightarrow{+} (z \in X \xleftrightarrow{+} z \in Y)]),$$
$$X \le Y = (X = Y) \vee (X < Y).$$

10. If $\varphi = \exists X.\psi$, then $\varphi^+ = \exists X.(\psi^+(X) \wedge \forall Y.(Y < X \wedge \psi(Y))^-)$ and $\varphi^- = \forall X.\psi^-$.
11. If $\varphi = \forall X.\psi$, then $\varphi^- = \exists X.(\psi^-(X) \wedge \forall Y.(X \leq Y \vee \psi(Y))^+)$ and $\varphi^+ = \forall X.\psi^+$.

We define the class of *syntactically unambiguous* formulas in $\mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ as the smallest class of formulas containing all formulas of the form

- $\varphi^+, \varphi^-, \varphi \xrightarrow{+} \xi$ and $\varphi \xleftrightarrow{+} \xi$ if $\varphi, \xi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ are unweighted, and
- $\forall y.\varphi, \forall X.\varphi$ or $\varphi \wedge \psi$ if it contains φ and ψ.

Example 33 Let $\mathcal{K}$ be the semiring of the natural numbers and let $w = (a, 0.7)(a, 1.5)(b, 3.0)$ be a timed word. As stated above, the formula $\varphi = \exists y.P_a(y)$ counts the number of a's in w, i.e., we have $([\![\varphi]\!], w) = 2$. In contrast to this, we have $([\![\varphi^+]\!], w) = 1$, where $\varphi^+ = \exists y.P_a(y) \wedge \forall z.[z \geq y \vee (z < y \wedge \neg P_a(z))]$.

□

By induction it is easy to show:

Lemma 34 *Let $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ be unweighted. Then we have*

1. *$L(\varphi^+) = L(\varphi)$ and $L(\varphi^-) = L(\neg\varphi)$,*
2. *$[\![\varphi^+]\!] = 1_{L(\varphi)}$ and $[\![\varphi^-]\!] = 1_{L(\neg\varphi)}$.*

Lemma 35 *Let $\psi_1, \psi_2 \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ be unweighted. Then the following equivalences hold.*

1. *$\psi_1^- \equiv (\psi_1^-)^+$,*
2. *$\psi_1^+ \wedge \psi_2^+ \equiv (\psi_1 \wedge \psi_2)^+$.*

We say that a formula $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is *syntactically unambiguous of bounded variability b* if it is of the form $\psi \wedge (\varphi_b)^+$ for some syntactically unambiguous formula ψ. Similarly, we say that φ is *almost unambiguous over y of bounded variability b* if it is in the disjunctive and conjunctive closure of syntactically unambiguous formulas of bounded variability b, constants $k \in K$ and formulas $f(y)$ for some $f \in \mathcal{F}$, such that $f(y)$ may appear at most once in every subformula of φ of the form $\varphi_1 \wedge \varphi_2$. Similarly to Lemma 19, we can prove the following lemma.

Lemma 36 *Let y be a first-order variable and $\psi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ be almost unambiguous over y of bounded variability b. Then there is a formula $\zeta \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ such that ζ is of the form $\bigvee_{1 \leq i \leq n} f_i(y) \wedge k_i \wedge \psi_i^+ \wedge (\varphi_b)^+$ for some $n \in \mathbb{N}$, $f_i \in \mathcal{F}$, $k_i \in K$ and unweighted $\psi_i \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ for each $i \in \{1, ..., n\}$, and $\zeta \equiv \psi$.*

We define the **syntactically restricted auxiliary logic of bounded variability** b $\mathsf{sRMSO}^\mathsf{b}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ to be the smallest class of formulas generated by the following rules.

1. If $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is an atomic or a weighted atomic formula, then $\varphi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
2. If $\varphi, \psi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, then $\varphi \vee \psi, \exists y.\varphi, \exists X.\varphi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
3. If $\varphi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is syntactically unambiguous of bounded variability b, then $\forall X.\varphi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
4. If $\varphi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is almost unambiguous over y of bounded variability b, then $\forall y.\varphi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
5. If $\varphi, \psi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ and at least one of the following conditions holds
   - $\mathcal{V}_f(\varphi) = \emptyset$,
   - $\mathcal{V}_f(\psi) = \emptyset$,
   - $\mathcal{V}_f(\varphi) \cap \mathcal{V}_f(\psi) = \emptyset$, $\mathcal{V}_f(\varphi) \subseteq \mathsf{Free}(\varphi)$, and $\mathcal{V}_f(\psi) \subseteq \mathsf{Free}(\psi)$,

   then $\varphi \wedge \psi \wedge \bigwedge_{\substack{y \in \mathcal{V}_f(\varphi), z \in \mathcal{V}_f(\psi) \\ y \neq z}} \neg(y = z) \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.

Note that the definition of $\mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ differs from the definition of $\mathsf{sRMSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ only in rules 3. and 4. Next, we want to prove the following theorem, the analogon to Theorem 21.

Theorem 37 *Let $\varphi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$. Then for each finite set $\mathcal{V} \supseteq \mathsf{Free}(\varphi)$ there is some weighted timed automaton $\mathcal{A}_\varphi$ over $\mathcal{K}$, $\Sigma_\mathcal{V}$ and $\mathcal{F}$ such that*

1. $\|\mathcal{A}_\varphi\| = [\![\varphi]\!]_\mathcal{V}$,
2. $\mathsf{Func}(\mathcal{A}_\varphi) \subseteq \mathsf{Func}(\varphi) \cup \{\mathbb{1}\}$,
3. *for each formula $f(y)$ occurring in φ with $y \in \mathsf{Free}(\varphi)$, whenever $\mathsf{lwt}(l) = f$ for some location l in $\mathcal{A}_\varphi$, then for each edge $(l, (a, \sigma), \phi, \lambda, l')$ in $\mathcal{A}_\varphi$ we have $\sigma(y) = 1$.*

The proof is along the lines of the proof of Theorem 21. However, we have to give new proofs for both universal quantifiers.

Second-Order Universal Quantification Let $\psi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ be syntactically unambiguous of bounded variability b and assume $\varphi = \forall X.\psi$. Hence, ψ is of the form $\zeta \wedge (\varphi_b)^+$ for some syntactically unambiguous ζ. Note that X does not occur in φ_b. Hence, we have $\forall X.(\zeta \wedge (\varphi_b)^+) \equiv \forall X.\zeta \wedge (\varphi_b)^+$, and thus φ is also syntactically unambiguous of bounded variability b. We consider the case where ζ is of the form η^+ for some unweighted $\eta \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$. The other cases can be reduced to this case. By definition of syntactically unambiguity and Lemma 35, we obtain

$$\forall X.\eta^+ \wedge (\varphi_b)^+ \equiv (\forall X.\eta)^+ \wedge (\varphi_b)^+ \equiv (\forall X.\eta \wedge \varphi_b)^+.$$

By Lemma 34, $[\![(\forall X.\eta \wedge \varphi_b)^+]\!] = 1_{L(\forall X.\eta \wedge \varphi_b)}$. We also have

$$L(\forall X.\eta \wedge \varphi_b) = L(\forall X.\eta) \cap L(\varphi_b) = L(\forall X.\eta) \cap T_b\Sigma^+.$$

By Theorem 10, $L(\forall X.\zeta)$ is TA-recognizable over $\Sigma_{\mathsf{Free}(\varphi)}$. Hence, by the first claim of Prop. 32, there is a deterministic timed automaton $\mathcal{A}$ such that $L(\mathcal{A}) = L(\forall X.\zeta \wedge \varphi_b)$. Let $\mathcal{A}_\varphi$ be the weighted timed automaton obtained from $\mathcal{A}$ as defined in the proof of Lemma 3.1. Then $\mathcal{A}_\varphi$ satisfies conditions 1. to 3. of Theorem 37.

For proving Theorem 37 for first-order universal quantification, we have to consider a modification of Lemma 26.

Lemma 38 *For every deterministically TA-recognizable timed language $L \subseteq T\Sigma^+$ over Σ, there is a timed automaton $\mathcal{A}'$ over Σ such that $L(\mathcal{A}') = L$, for each location l in $\mathcal{A}'$ there is a unique $a \in \Sigma$ such that every edge $(l, a', \phi, \lambda, l')$ in $\mathcal{A}'$ satisfies $a' = a$, and $\mathcal{A}'$ is unambiguous, i.e., for each timed word $w \in T\Sigma^+$, there is at most one successful run of $\mathcal{A}'$ on w.*

Proof (sketch). The construction of $\mathcal{A}'$ is very similar to that in the proof of Lemma 26. However, for obtaining the unambiguity of $\mathcal{A}'$, we let $\mathcal{L}'_f$ be a singleton set containing a *new* location l_f, and we add new edges of the form $(l, a, \phi, \lambda, l_f)$ for each $(l, a, \phi, \lambda, l')$ such that $l' \in \mathcal{L}_f$. This must be done to guarantee the uniqueness of the successful runs, because if we let $\mathcal{L}'_f = \mathcal{L}_f \times \Sigma$ (as in the proof of Lemma 26), we could not conclude that the last location (l, a) of a successful run is uniquely determined by the subsequent letter as it is for the other locations in the run. ■

First-Order Universal Quantification The proof is along the lines of the proof of Theorem 21. However, we have to show that $\widetilde{L}$ is *deterministically* TA-recognizable in order to apply Lemma 38.

Let $\psi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ be almost unambiguous over y of bounded variability b and assume $\varphi = \forall y.\psi$. By Lemma 36, we may assume that ψ is of the form

$$\psi = \bigvee_{1 \le j \le n} f_j(y) \wedge k_j \wedge \psi_j^+ \wedge (\varphi_b)^+$$

where $n \in \mathbb{N}$, $k_j \in K$, $f_j \in \mathcal{F}$, unweighted $\psi_j \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ for each $j \in \{1, ..., n\}$.

Let $\mathcal{W} = \mathsf{Free}(\psi)$ and $\mathcal{V} = \mathsf{Free}(\varphi) = \mathcal{W} \backslash \{y\}$. For each $i \in \{1, ..., n\}$, we have $\psi_j^+ \wedge (\varphi_b)^+ \equiv (\psi_j \wedge \varphi_b)^+$ by Lemma 35, and thus $[\![\psi_j^+ \wedge (\varphi_b)^+]\!] = 1_{L(\psi_j \wedge \varphi_b)}$ by Lemma 34. We define $\widetilde{L} \subseteq T((\Sigma^{(n)})_\mathcal{V})^+$ to be the set of timed words $((\bar{a}, \mu, \sigma), \bar{t})$ in $T((\Sigma^{(n)})_\mathcal{V})^+$ such that $((\bar{a}, \sigma), \bar{t}) \in N_\mathcal{V}$, and for all $i \in \mathsf{dom}(\bar{a}, \bar{t})$ and $j \in \{1, ..., n\}$ we have

$$\mu(i) = j \text{ implies } ((\bar{a}, \sigma[y \to i]), \bar{t}) \models \psi_j \wedge \varphi_b$$

We prove that $\widetilde{L}$ is unambiguously TA-recognizable. For this, consider the formula $\zeta \in \mathsf{MSO}(T(\Sigma^{(n)})^+)$

$$\zeta = \forall y. \bigwedge_{1 \le j \le n} \bigwedge_{a \in \Sigma} \left(P_{(a,j)}(y) \longrightarrow \widetilde{\psi_j} \wedge \widetilde{\varphi_b} \right).$$

Let $((\bar{a},\mu,\sigma),\bar{t}) \in T((\Sigma^{(n)})_{\mathcal{V}})^+$ such that $((\bar{a},\sigma),\bar{t}) \in N_{\mathcal{V}}$. Then we have

$$\begin{aligned}
&((\bar{a},\mu,\sigma),\bar{t}) \models \zeta \\
\Leftrightarrow\ & \forall i \in \mathsf{dom}(\bar{a},\bar{t}), \forall j \in \{1,...,n\}.\mu(i) = j \Rightarrow ((\bar{a},\mu,\sigma[y \to i]),\bar{t}) \models \widetilde{\psi_j} \wedge \widetilde{\varphi_b} \\
\Leftrightarrow\ & \forall i \in \mathsf{dom}(\bar{a},\bar{t}), \forall j \in \{1,...,n\}.\mu(i) = j \Rightarrow ((\bar{a},\sigma[y \to i]),\bar{t}) \models \psi_j \wedge \varphi_b \\
\Leftrightarrow\ & ((\bar{a},\mu,\sigma),\bar{t}) \in \widetilde{L}.
\end{aligned}$$

Now, observe that for each $i \in \mathsf{dom}(\bar{a},\bar{t})$ there exists some $j \in \{1,..,n\}$ and some $a \in \Sigma$ such that $P_{(a,j)}(y)$ holds. This implies that φ_b always holds. Hence, ζ is equivalent to $\zeta' \wedge \varphi_b$, where $\zeta' = \forall y. \Big(\bigwedge_{1 \le j \le n} \bigwedge_{a \in \Sigma} (P_{(a,j)}(y) \longrightarrow \widetilde{\psi_j})\Big)$. Now, Theorem 10 implies that $L(\zeta')$ is TA-recognizable over $(\Sigma^{(n)})_{\mathcal{V}}$. But then, by Prop. 32, we know that there is a deterministic timed automaton $\widetilde{\mathcal{A}}$ recognizing $L(\zeta) = \widetilde{L}$.

From $\widetilde{\mathcal{A}}$, we construct an unambiguous timed automaton $\mathcal{A}'$ using Lemma 38. From this, we can define a weighted timed automaton $\mathcal{A}$ as described in the proof of Theorem 21. Note that $\mathcal{A}$ is unambiguous, and thus we have

$$\big(\|\mathcal{A}\|, ((\bar{a},\mu,\sigma),\bar{t})\big) = \prod_{i \in \mathsf{dom}(\bar{a},\bar{t})} f_{\mu(i)}(t_i - t_{i-1}) \cdot k_{\mu(i)}$$

for each $((\bar{a},\mu,\sigma),\bar{t}) \in T((\Sigma^{(n)})_{\mathcal{V}})^+$. Then we can proceed exactly as in the proof of Theorem 21. This finishes the proof of Theorem 37.

■

We define the **syntactically restricted weighted relative distance logic of bounded variability** b, denoted by $\mathsf{sR}\mathcal{L}\overleftarrow{\mathsf{d}}^{\,\mathsf{b}}(\mathcal{K},\Sigma,\mathcal{F})$ to be the smallest class of formulas containing all formulas generated by the next two rules.

1. If $\varphi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, then $\varphi \in \mathsf{sR}\mathcal{L}\overleftarrow{\mathsf{d}}^{\,\mathsf{b}}(\mathcal{K},\Sigma,\mathcal{F})$.
2. If $\varphi \in \mathsf{sR}\mathcal{L}\overleftarrow{\mathsf{d}}^{\,\mathsf{b}}(\mathcal{K},\Sigma,\mathcal{F})$, then $\exists D.\varphi \in \mathsf{sR}\mathcal{L}\overleftarrow{\mathsf{d}}^{\,\mathsf{b}}(\mathcal{K},\Sigma,\mathcal{F})$.

For the other direction, i.e., that every $\mathcal{F}$-recognizable timed series can be defined by a sentence in $\mathsf{sR}\mathcal{L}\overleftarrow{\mathsf{d}}^{\,\mathsf{b}}(\mathcal{K},\Sigma,\mathcal{F})$, we can adopt the proof of Theorem 29 by

– considering the syntactically unambiguous version of $\psi(\bar{\mathcal{D}},\bar{Y})$, and
– combining formulas with φ_b whenever it is needed.

We obtain a Büchi theorem for the class of $\mathcal{F}$-recognizable timed series over commutative semirings.

Theorem 39 *Let $\mathcal{K}$ be commutative and $\mathcal{F}$ contain $\mathbb{1}$. Each timed series $\mathcal{T} : T\Sigma^+ \to K$ is $\mathcal{F}$-recognizable if and only if $\mathcal{T}$ is $\mathsf{sR}\mathcal{L}\overleftarrow{\mathsf{d}}^{\,\mathsf{b}}(\mathcal{K},\Sigma,\mathcal{F})$-definable.*

Next, we explain how we can even skip the restriction on $\mathcal{K}$ being commutative.

For this, we let $\mathcal{K}$ be a semiring, not necessarily being commutative. We follow the approach of Droste and Gastin [13], and only present the main ideas. Commutativity of $\mathcal{K}$ is mainly needed for showing closure of the class of $\mathcal{F}$-recognizable timed series under the Hadamard product (Lemma 6). In the proof, we exploit the fact that the weights occuring in the runs of $\mathcal{A}_1$ *commute element-wise* with the weights occuring in the runs of $\mathcal{A}_2$. However, commutativity of $\mathcal{K}$ is a sufficient but not a necessary condition for the element-wise commutativity of weights occuring in the runs of weighted timed automata. For instance, the weights occuring in a weighted timed automaton over $\mathcal{K}$ commute element-wise with the weights occuring in a weighted timed automaton over the semiring which is generated by $\{0,1\} \subseteq K$.

For the proof of the following lemma we may proceed as in the proof of Lemma 6.

Lemma 40 *Let $\mathcal{K}_1, \mathcal{K}_2$ be two subsemirings of $\mathcal{K}$ such that $\mathcal{K}_1$ commutes element-wise with $\mathcal{K}_2$. If $\mathcal{T}_1$ is recognizable by a weighted timed automaton $\mathcal{A}_1$ over $\mathcal{K}_1$, Σ and $\mathcal{F}$, and $\mathcal{T}_2$ is recognizable by a weighted timed automaton $\mathcal{A}_2$ over $\mathcal{K}_2$, Σ and $\mathcal{F}$, and $\mathcal{A}_1$ and $\mathcal{A}_2$ are non-interfering, then $\mathcal{T}_1 \odot \mathcal{T}_2$ is $\mathcal{F}$-recognizable.*

Let $\varphi \in \overleftarrow{\mathcal{L}\mathsf{d}}(\mathcal{K}, \Sigma, \mathcal{F})$. We define $\mathsf{wgt}(\varphi) = \mathsf{wgt}_E(\varphi) \cup \mathsf{wgt}_{\mathcal{F}}(\varphi)$, where $\mathsf{wgt}_E(\varphi) = \{k : k \text{ is a subformula of } \varphi\}$ and $\mathsf{wgt}_{\mathcal{F}}(\varphi) = \{f(\delta) : f(y) \text{ is a subformula of } \varphi, \delta \in \mathbb{R}_{\geq 0}\}$. We define the **syntactically restricted auxiliary logic of bounded variability** b **for non-commutative semirings** $\mathsf{sRMSO}^{\mathsf{bnc}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ to be the smallest class of formulas generated by the following rules.

1. If $\varphi \in \mathsf{MSO}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is an atomic or a weighted atomic formula, then $\varphi \in \mathsf{sRMSO}^{\mathsf{bnc}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
2. If $\varphi, \psi \in \mathsf{sRMSO}^{\mathsf{bnc}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, then $\varphi \vee \psi, \exists y.\varphi, \exists X.\varphi \in \mathsf{sRMSO}^{\mathsf{bnc}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
3. If $\varphi \in \mathsf{sRMSO}^{\mathsf{bnc}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is syntactically unambiguous of bounded variability b, then $\forall X.\varphi \in \mathsf{sRMSO}^{\mathsf{bnc}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
4. If $\varphi \in \mathsf{sRMSO}^{\mathsf{bnc}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ is almost unambiguous over y of bounded variability b, then $\forall y.\varphi \in \mathsf{sRMSO}^{\mathsf{b}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.
5. If $\varphi, \psi \in \mathsf{sRMSO}^{\mathsf{bnc}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$ and at least one of the following three conditions hold
   - $\mathcal{V}_f(\varphi) = \emptyset$,
   - $\mathcal{V}_f(\psi) = \emptyset$,
   - $\mathcal{V}_f(\varphi) \cap \mathcal{V}_f(\psi) = \emptyset$, $\mathcal{V}_f(\varphi) \subseteq \mathsf{Free}(\varphi)$, and $\mathcal{V}_f(\psi) \subseteq \mathsf{Free}(\psi)$,

   and
   - φ and ψ are not in the scope of a universal first-order quantifier, and
   - $\mathsf{wgt}(\varphi)$ and $\mathsf{wgt}(\psi)$ commute element-wise,

   then $\varphi \wedge \psi \wedge \bigwedge_{\substack{y \in \mathcal{V}_f(\varphi), z \in \mathcal{V}_f(\psi) \\ y \neq z}} \neg(y = z) \in \mathsf{sRMSO}^{\mathsf{bnc}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$.

Notice that we added further restrictions on the application of conjunction. We use $\mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}^{\,\mathsf{bnc}}(\mathcal{K}, \Sigma, \mathcal{F})$ to denote the smallest class of formulas containing formulas generated by the next two rules.

1. If $\varphi \in \mathsf{sRMSO}^{\mathsf{bnc}}(\mathcal{K}, T\Sigma^+, \mathcal{F})$, then $\varphi \in \mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}^{\mathsf{bnc}}(\mathcal{K}, \Sigma, \mathcal{F})$.
2. If $\varphi \in \mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}^{\mathsf{bnc}}(\mathcal{K}, \Sigma, \mathcal{F})$, then $\exists D.\varphi \in \mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}^{\mathsf{bnc}}(\mathcal{K}, \Sigma, \mathcal{F})$.

Note that, as opposed to the other conditions, it depends on $\mathcal{K}$ and $\mathcal{F}$ whether one can check syntactically whether two given formulas φ and ψ satisfy that $\mathsf{wgt}(\varphi)$ and $\mathsf{wgt}(\psi)$ commute element-wise. If $\mathsf{wgt}(\varphi)$ is a finite set, which is e.g. the case if there are no location weight functions occuring in φ, or $\mathcal{F}$ is the family of step functions, then we can easily check syntactically whether the weights in $\mathsf{wgt}(\varphi)$ commute element-wise. For other cases, this might not be so easy or even impossible.

Using fairly the same proof methods as before, we obtain the following theorem, generalizing the previous Büchi-type theorems.

Theorem 41 *Let $\mathcal{F}$ contain $\mathbb{1}$. Then a timed series $\mathcal{T} : T\Sigma^+ \to K$ is $\mathcal{F}$-recognizable if and only if $\mathcal{T}$ is $\mathsf{sR}\overleftarrow{\mathcal{L}\mathsf{d}}^{\mathsf{bnc}}(\mathcal{K}, \Sigma, \mathcal{F})$-definable.*

6 Conclusion

We have presented weighted timed MSO logics, which is - at least to our knowledge - the first MSO logic allowing for the description of both timed and quantitative properties. On the one hand, our logic may be used as a new tool for specifying properties. It sometimes may be easier to specify properties in terms of logic rather than by automata devices. On the other hand, this logic gives rise to some interesting new directions in future research work. For instance, Wilke [29] showed that some real-time temporal logics are effectively embeddable into the relative distance logic. All his constructions for obtaining a Büchi theorem are effective. By the decidability of the emptiness problem for timed automata [1], one can thus conclude that these real-time temporal logics have a decidable theory. This gives rise to the question whether one can obtain similar results for weighted extensions of real-time temporal logics.

We also would like to mention that our logic and constructions follow the ideas of the work of Droste and Gastin [13] and thus we keep the spirit of the untimed theory. However, we additionally allowed functions from the family as atomic formulas, which complicates most of the proofs, first and foremost the proof for showing closure of the class of recognizable timed series under conjunction and first-order universal quantification. For conjunction, we have stated conditions for closure of recognizable timed series under the Hadamard product, which corresponds to the intersection operation in the unweighted setting. Moreover, we had to deal with the problem that - unlike finite automata - timed automata are not determinizable in general.

Lastly, the coincidence between recognizable and MSO-definable timed series, together with a previous work on weighted timed automata concerning a Kleene-Schützenberger Theorem [14], shows the robustness of the notion of recognizable timed series, as they can equivalently be characterized in terms of automata, logics and rational operations.

Acknowledgement

I would like to thank Manfred Droste, Stefan Göller, Christian Mathissen and two anonymous referees for their helpful discussions and comments.

References

1. R. Alur and D. L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–235, 1994.
2. R. Alur, S. La Torre, and G. J. Pappas. Optimal paths in weighted timed automata. *Theoretical Computer Science*, 318:297–322, 2004.
3. R. Alur and P. Madhusudan. Decision problems for timed automata: A survey. In M. Bernardo and F. Corradini, editors, *SFM-RT*, volume 3185 of *LNCS*, pages 1–24. Springer, 2004.
4. G. Behrmann, A. Fehnker, T. Hune, K. Larsen, P. Pettersson, J. Romijn, and F. Vaandrager. Minimum-cost reachability for priced timed automata. In M. D. Di Benedetto and A. Sangiovanni-Vincentelli, editors, *HSCC*, volume 2034 of *LNCS*, pages 147–161. Springer, 2001.
5. B. Bollig and I. Meinecke. Weighted distributed systems and their logics. In S. N. Artëmov and A. Nerode, editors, *LFCS*, volume 4514 of *LNCS*, pages 54–68. Springer, 2007.
6. P. Bouyer, T. Brihaye, V. Bruyère, and J.-F. Raskin. On the optimal reachability problem on weighted timed automata. *Formal Methods in System Design*, 31(2):135–175, 2007.
7. P. Bouyer, T. Brihaye, and N. Markey. Improved undecidability results on weighted timed automata. *Information Processing Letters*, 98(5):188–194, 2006.
8. P. Bouyer, K. G. Larsen, and N. Markey. Model checking one-clock priced timed automata. *Logical Methods in Computer Science*, 4:1–28, 2008.
9. P. Bouyer and N. Markey. Costs are expensive! In J.-F. Raskin and P. S. Thiagarajan, editors, *FORMATS*, volume 4763 of *LNCS*, pages 53–68. Springer, 2007.
10. J. R. Büchi. On a decision method in restricted second order arithmetics. In E. Nagel et al., editor, *Proc. Intern. Congress on Logic, Methodology and Philosophy of Sciences*, pages 1–11, Stanford, 1960. Stanford University Press.
11. A. Chiplunkar, S. Narayanan Krishna, and C. Jain. Model checking logic WCTL with multi constrained modalities on one clock priced timed automata. In J. Ouaknine and F. W. Vaandrager, editors, *FORMATS*, volume 5813 of *LNCS*, pages 88–102. Springer, 2009.
12. M. Droste and P. Gastin. Weighted automata and weighted logics. In L. Caires, G. F. Italiano, L. Monteiro, C. Palamidessi, and M. Yung, editors, *ICALP*, volume 3580 of *LNCS*, pages 513–525. Springer, 2005.
13. M. Droste and P. Gastin. Weighted automata and weighted logics. In M. Droste, W. Kuich, and H. Vogler, editors, *Handbook of Weighted Automata*, EATCS Monographs in Theoretical Computer Science, pages 175–211. Springer, 2009.
14. M. Droste and K. Quaas. A Kleene-Schützenberger theorem for weighted timed automata. In R. M. Amadio, editor, *FoSSaCS*, volume 4962 of *LNCS*, pages 142–156. Springer, 2008.
15. M. Droste and G. Rahonis. Weighted automata and weighted logics on infinite words. In O. H. Ibarra and Z. Dang, editors, *Developments in Language Theory*, volume 4036 of *LNCS*, pages 49–58. Springer, 2006.

16. M. Droste and H. Vogler. Weighted tree automata and weighted logics. *Theoretical Computer Science*, 366(3):228–247, 2006.
17. D. D'Souza. A logical characterisation of event clock automata. *Int. J. Found. Comput. Sci.*, 14(4):625–640, 2003.
18. C. A. Furia and M. Rossi. MTL with bounded variability: Decidability and complexity. In F. Cassez and C. Jard, editors, *FORMATS*, volume 5215 of *LNCS*, pages 109–123. Springer, 2008.
19. C. Mathissen. Definable transductions and weighted logics for texts. In T. Harju, J. Karhumäki, and A. Lepistö, editors, *Developments in Language Theory*, volume 4588 of *LNCS*, pages 324–336. Springer, 2007.
20. C. Mathissen. Weighted logics for nested words and algebraic formal power series. In L. Aceto, I. Damgård, L. Ann Goldberg, M. M. Halldórsson, A. Ingólfsdóttir, and I. Walukiewicz, editors, *ICALP (2)*, volume 5126 of *LNCS*, pages 221–232. Springer, 2008.
21. I. Mäurer. Weighted picture automata and weighted logics. In B. Durand and W. Thomas, editors, *STACS*, volume 3884 of *LNCS*, pages 313–324. Springer, 2006.
22. I. Meinecke. Weighted logics for traces. In D. Grigoriev, J. Harrison, and E. A. Hirsch, editors, *CSR*, volume 3967 of *LNCS*, pages 235–246. Springer, 2006.
23. K. Quaas. *Kleene-Schützenberger and Büchi theorems for Weighted Timed Automata*. PhD thesis, Universität Leipzig, Institut für Informatik, Abteilung Automaten und Sprachen, 2009.
24. K. Quaas. Weighted timed MSO logics. In V. Diekert and D. Nowotka, editors, *DLT 2009, Proceedings*, volume 5583 of *LNCS*, pages 419–430. Springer, 2009.
25. R. Alur and T.A. Henzinger. Real-time logics: complexity and expressiveness. In *Fifth Annual IEEE Symposium on Logic in Computer Science*, pages 390–401, Washington, D.C., 1990. IEEE Computer Society Press, IEEE Computer Society Press.
26. W. Thomas. Automata on infinite objects. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics (B)*, pages 133–192. Elsevier and MIT Press, 1990.
27. W. Thomas. Languages, automata and logic. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, pages 389–485. Springer, 1997.
28. T. Wilke. *Automaten und Logiken zur Beschreibung zeitabhängiger Systeme*. PhD thesis, Christian-Albrecht-Universität Kiel, 1994.
29. T. Wilke. Specifying timed state sequences in powerful decidable logics and timed automata. In H. Langmaack, W.-P. de Roever, and J. Vytopil, editors, *Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 863 of *LNCS*, pages 694–715, Lübeck, Germany, 1994. Springer-Verlag.