Weight Assignment Logic[*]

Vitaly Perevoshchikov[**]

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

Abstract. We introduce a weight assignment logic for reasoning about quantitative languages of infinite words. This logic is an extension of the classical MSO logic and permits to describe quantitative properties of systems with multiple weight parameters, e.g., the ratio between rewards and costs. We show that this logic is expressively equivalent to unambiguous weighted Büchi automata. We also consider an extension of weight assignment logic which is expressively equivalent to nondeterministic weighted Büchi automata.

Keywords: quantitative omega-languages, quantitative logic, multi-weighted automata, Büchi automata, unambiguous automata

1 Introduction

Since the seminal Büchi theorem [6] about the expressive equivalence of finite automata and monadic second-order logic, a significant field of research investigates logical characterizations of language classes appearing from practically relevant automata models. In this paper we introduce a new approach to the logical characterization of quantitative languages of infinite words where every infinite word carries a value, e.g., a real number.

Quantitative languages of infinite words and various weighted automata for them were investigated by Chatterjee, Doyen and Henzinger in [7] as models for verification of quantitative properties of systems. Their weighted automata are automata with a single weight parameter where a computation is evaluated using measures like the limit average or discounted sum. Recently, the problem of analysis and verification of systems with multiple weight parameters, e.g. time, costs and energy consumption, has received much attention in the literature [2, 3, 5, 17, 18, 21]. For instance, the setting where a computation is evaluated as the ratio between accumulated rewards and costs was considered in [3, 5, 18]. Another example is a model of energy automata with several energy storages [17].

Related work. Droste and Gastin [9] introduced weighted MSO logic on finite words with constants from a semiring. In the semantics of their logic (which is a quantitative language of finite words) disjunction is extended by the sum operation of the semiring and conjunction is extended by the product. They show that weighted MSO

[*] This is the full version of the paper published at DLT 2015 [23]

[**] Supported by DFG Research Training Group 1763 (QuantLA)

logic is more expressive than weighted automata [10] (the unrestricted use of weighted conjunction and weighted universal quantifiers leads to unrecognizability) and provide a syntactically restricted fragment which is expressively equivalent to weighted automata. This result was extended in [15] to the setting of infinite words. A logical characterization of the quantitative languages of Chatterjee, Doyen and Henzinger was given in [12] (again by a restricted fragment of weighted MSO logic). In [14], a multi-weighted extension of weighted MSO logic of [12] with the multiset-based semantics was considered.

Our contributions. In this paper, we introduce a new approach to logic for quantitative languages, different from [9, 12, 14, 15]. We develop a so-called *weight assignment logic* (WAL) on infinite words, an extension of the classical MSO logic to the quantitative setting. This logic allows us to assign weights (or multi-weights) to positions of an ω-word. Using WAL, we can, for instance, express that whenever a position of an input word is labelled by letter a, then the weight of this position is 2. As a weighted extension of the logical conjunction, we use the *merging* of partially defined ω-words. In order to evaluate a partially defined ω-word, we introduce a *default weight*, assign it to all positions with undefined weight, and evaluate the obtained totally defined ω-word, e.g., as the reward-cost ratio or discounted sum.

As opposed to the weighted MSO logic of [9], the weighted conjunction-like operators of WAL capture recognizability by weighted Büchi automata. We show that WAL is expressively equivalent to *unambiguous* weighted Büchi automata where, for every input ω-word, there exists at most one accepting computation. Unambiguous automata are of considerable interest for automata theory as they can have better decidability properties. For instance, in the setting of finite words, the equivalence problem for unambiguous max-plus automata is decidable [19] whereas, for nondeterministic max-plus automata, this problem is undecidable [20].

We also consider an extended version of WAL which captures nondeterministic weighted Büchi automata. In extended WAL we allow existential quantification over first-order and second-order variables in the prefix of a formula. The structure of extended WAL is similar to the structure of unweighted logics for, e.g., timed automata [25] and data automata [4].

For the proof of our expressiveness equivalence result, we establish a Nivat decomposition theorem for nondeterministic and unambiguous weighted Büchi automata. Recall that Nivat's theorem [22] is one of the fundamental characterizations of rational transductions and shows a connection between rational transductions and rational language. Recently, Nivat's theorem was proved for semiring-weighted automata on finite words [11] and weighted multioperator tree automata [24]. We obtain similar decompositions for WAL and extended WAL and deduce our results from the classical Büchi theorem [6]. Our proof is constructive and hence decidability properties for WAL and extended WAL can be transferred into decidability properties of weighted Büchi automata. As a side application of our Nivat theorem, we can easily show that weighted Büchi automata and weighted Muller automata are expressively equivalent.

Outline. In Sect. 2 we introduce a general framework for weighted Büchi automata and consider several examples. In Sect. 3 we prove a Nivat decomposition theorem for weighted Büchi automata. In Sect. 4 we define weight assignment logic and its extension. In Sect. 5 we state our main result and give a sketch of its proof for the unambiguous and nondeterministic cases.

2 Weighted Büchi Automata

Let $\mathbb{N} = \{0, 1, ...\}$ denote the set of all natural numbers. For an arbitrary set X, an *ω-word* over X is an infinite sequence $(x_i)_{i \in \mathbb{N}}$ where $x_i \in X$ for all $i \in \mathbb{N}$. Let X^ω denote the set of all ω-words over X. Any set $\mathcal{L} \subseteq X^\omega$ is called an *ω-language* over X.

A *Büchi automaton* over an alphabet Σ is a tuple $\mathcal{A} = (Q, I, T, F)$ where Q is a finite set of states, Σ is an alphabet (i.e. a finite non-empty set), $I, F \subseteq Q$ are sets of initial resp. accepting states, and $T \subseteq Q \times \Sigma \times Q$ is a transition relation. A *run* $\rho = (t_i)_{i \in \mathbb{N}} \in T^\omega$ of $\mathcal{A}$ is defined as an infinite sequence of matching transitions which starts in an initial state and visits some accepting state infinitely often, i.e., $t_i = (q_i, a_i, q_{i+1})$ for each $i \in \mathbb{N}$, such that $q_0 \in I$ and $\{q \in Q \mid q = q_i \text{ for infinitely many } i \in \mathbb{N}\} \cap F \neq \emptyset$. Let $\mathrm{label}(\rho) := (a_i)_{i \in \mathbb{N}} \in \Sigma^\omega$, the *label* of ρ. We denote by $\mathrm{Run}_\mathcal{A}$ the set of all runs of $\mathcal{A}$ and, for each $w \in \Sigma^\omega$, we denote by $\mathrm{Run}_\mathcal{A}(w)$ the set of all runs ρ of $\mathcal{A}$ with $\mathrm{label}(\rho) = w$. Let $\mathcal{L}(\mathcal{A}) = \{w \in \Sigma^\omega \mid \mathrm{Run}_\mathcal{A}(w) \neq \emptyset\}$, the ω-language *accepted* by $\mathcal{A}$. We call an ω-language $\mathcal{L} \subseteq \Sigma^\omega$ *recognizable* if there exists a Büchi automaton $\mathcal{A}$ over Σ such that $\mathcal{L}(\mathcal{A}) = \mathcal{L}$.

We say that a monoid $\mathbb{K} = (K, +, \mathbb{0})$ is *complete* (cf., e.g., [15]) if it is equipped with infinitary sum operations $\sum_I : K^I \to K$ for any index set I, such that, for all I and all families $(k_i)_{i \in I}$ of elements of K, the following hold:

- $\sum_{i \in \emptyset} k_i = \mathbb{0}$, $\sum_{i \in \{j\}} k_i = k_j$, $\sum_{i \in \{p,q\}} k_i = k_p + k_q$ for $p \neq q$;
- $\sum_{j \in J} (\sum_{i \in I_j} k_i) = \sum_{i \in I} k_i$, if $\bigcup_{j \in J} I_j = I$ and $I_j \cap I_{j'} = \emptyset$ for $j \neq j'$.

Let $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$. Then, $\overline{\mathbb{R}}$ equipped with infinitary operations like *infinum* or *supremum* forms a complete monoid. Now we introduce an algebraic structure for weighted Büchi automata which is an extension of totally complete semirings [15] and valuation monoids [12] and covers various multi-weighted measures.

Definition 2.1. *A* valuation structure $\mathbb{V} = (M, \mathbb{K}, \mathrm{val})$ *consists of a non-empty set M, a complete monoid $\mathbb{K} = (K, +, \mathbb{0})$ and a mapping $\mathrm{val} : M^\omega \to K$ called henceforth a* valuation function.

In the definition of a valuation structure we have two weight domains M and K. Here M is the set of transition weights which in the multi-weighted examples can be tuples of weights (e.g., a reward-cost pair) and K is the set of weights of computations which can be single values (e.g., the ratio between rewards and costs).

Definition 2.2. *Let Σ be an alphabet and $\mathbb{V} = (M, (K, +, \mathbb{0}), \mathrm{val})$ a valuation structure. A* weighted Büchi automaton *(WBA) over $\mathbb{V}$ is a tuple $\mathcal{A} = (Q, I, T, F, \mathrm{wt})$ where (Q, I, T, F) is a Büchi automaton over Σ and $\mathrm{wt} : T \to M$ is a transition weight function.*

The behavior of WBA is defined as follows. Given a run ρ of this automaton, we evaluate the ω-sequence of transition weights of ρ (which is in M^ω) using the valuation function val and then resolve the nondeterminism on the weights of runs using the complete monoid $\mathbb{K}$. Formally, let $\rho = (t_i)_{i \in \mathbb{N}} \in T^\omega$ be a run of $\mathcal{A}$. Then, the *weight* of ρ is defined as $\mathrm{wt}_\mathcal{A}(\rho) = \mathrm{val}((\mathrm{wt}(t_i))_{i\in\mathbb{N}}) \in K$. The *behavior* of $\mathcal{A}$ is a mapping $[\![\mathcal{A}]\!] : \Sigma^\omega \to K$ defined for all $w \in \Sigma^\omega$ by $[\![\mathcal{A}]\!](w) = \sum(\mathrm{wt}_\mathcal{A}(\rho) \mid \rho \in \mathrm{Run}_\mathcal{A}(w))$. Note that the sum in the equation above can be infinite. Therefore we consider a complete monoid $(K, +, \mathbb{0})$. A mapping $\mathbb{L} : \Sigma^\omega \to K$ is called a *quantitative ω-language*. We say that $\mathbb{L}$ is (nondeterministically) *recognizable* over $\mathbb{V}$ if there exists a WBA $\mathcal{A}$ over $\mathbb{V}$ such that $[\![\mathcal{A}]\!] = \mathbb{L}$.

We say that a WBA $\mathcal{A}$ over Σ and $\mathbb{V}$ is *unambiguous* if $|\mathrm{Run}_\mathcal{A}(w)| \leq 1$ for every $w \in \Sigma^\omega$. We call a quantitative ω-language $\mathbb{L} : \Sigma^\omega \to K$ *unambiguously recognizable* over $\mathbb{V}$ if there exists an unambiguous WBA $\mathcal{A}$ over Σ and $\mathbb{V}$ such that $[\![\mathcal{A}]\!] = \mathbb{L}$.

Example 2.3. (a) The ratio measure was introduced in [5], e.g., for the modeling of the average costs in timed systems. In the setting of ω-words, we consider the model with two weight parameters: the *cost* and the *reward*. The rewards and costs of transitions are accumulated along every finite prefix of a run and their ratio is taken. Then, the weight of an infinite run is defined as the limit superior (or limit inferior) of the sequence of the computed ratios for all finite prefixes. To describe the behavior of these double-priced ratio Büchi automata, we consider the valuation structure $\mathbb{V}^{\text{Ratio}} = (M, \mathbb{K}, \mathrm{val})$ where $M = \mathbb{Q} \times \mathbb{Q}_{\geq 0}$ models the reward-cost pairs, $\mathbb{K} = (\overline{\mathbb{R}}, \sup, -\infty)$ and $\mathrm{val} : M^\omega \to \overline{\mathbb{R}}$ is defined for every sequence $u = ((r_i, c_i))_{i\in\mathbb{N}} \in M^\omega$ by $\mathrm{val}(u) = \limsup_{n\to\infty} \frac{r_1 + \ldots + r_n}{c_1 + \ldots + c_n}$. Here, we assume that $\frac{r}{0} = -\infty$.

(b) *Discounting* [1, 7] is a well-known principle which is used in, e.g., economics and psychology. In this example, we consider WBA with transition-dependent discounting, i.e., are two weight parameters: the cost and the discounting factor (which is not fixed and depends on a transition). In order to define WBA with discounting formally, we consider the valuation structure $\mathbb{V}^{\text{Disc}} = (M, \mathbb{K}, \mathrm{val})$ where $M = \mathbb{Q}_{\geq 0} \times ((0, 1] \cap \mathbb{Q})$ models the pairs of a cost and a discounting factor, $\mathbb{K} = (\mathbb{R}_{\geq 0} \cup \{\infty\}, \inf, \infty)$, and val is defined for all $u = ((c_i, d_i))_{i\in\mathbb{N}} \in M^\omega$ as $\mathrm{val}(u) = c_0 + \sum_{i=1}^{\infty} c_i \cdot \prod_{j=0}^{i-1} d_j$.

(c) Now we consider the valuation structure for the model of multi-weighted automata which correspond to one-player energy games with lower bound considered in [17]. Let $n \geq 1$ and $s_1, ..., s_n$ be energy storages. We start with empty storages and, after taking a transition of a Büchi automaton, the energy level of each storage s_j ($1 \leq j \leq n$) can be increased (if we regain energy) or decreased (if we consume energy). The goal is to keep the energy level of every energy storage not less than zero. Consider the sequence $u = (u_i)_{u\in\mathbb{N}}$ where, for all $i \in \mathbb{N}$, $u_i = (u_i^1, ..., u_i^n) \in \mathbb{Z}^n$ is the vector of the energy level changes for each storage. We say that u is *correct* if $\sum_{k=0}^{i} u_k^j \geq 0$ for all $i \in \mathbb{N}$ and $j \in \{1, ..., n\}$. For this situation we consider the valuation structure $\mathbb{V}^{\text{Energy}} = (M, \mathbb{K}, \mathrm{val})$ where $M = \mathbb{Z}^n$, $\mathbb{K} = (\{0, 1\}, \vee, 0)$ and, for all $u \in M^\omega$, we let $\mathrm{val}(u) = 1$ if u is correct and $\mathrm{val}(u) = 0$ otherwise.

(d) Since a valuation monoid $(K, (K, +, \mathbb{0}), \mathrm{val})$ of Droste and Meinecke [12] is a special case of valuation structures, all examples considered there also fit into our framework. □

3 Decomposition of WBA

In this section, we establish a Nivat decomposition theorem for WBA. We will need it for the proof of our main result. However, it also could be of independent interest.

Let Σ be an alphabet and $\mathbb{V} = (M, (K, +, \mathbb{0}), \mathrm{val})$ a valuation structure. For a (possibly different from Σ) alphabet Γ, we introduce the following operations. Let Δ be an arbitrary non-empty set and $h : \Gamma \to \Delta$ a mapping called henceforth a *renaming*. For any ω-word $u = (\gamma_i)_{i \in \mathbb{N}} \in \Gamma^\omega$, we let $h(u) = (h(\gamma_i))_{i \in \mathbb{N}} \in \Delta^\omega$. Now let $h : \Gamma \to \Sigma$ be a renaming and $\mathbb{L} : \Gamma^\omega \to K$ a quantitative ω-language. We define the *renaming* $h(\mathbb{L}) : \Sigma^\omega \to K$ for all $w \in \Sigma^\omega$ by $h(\mathbb{L})(w) = \sum \big(\mathbb{L}(u) \mid u \in \Gamma^\omega \text{ and } h(u) = w\big)$. For a renaming $g : \Gamma \to M$, the *composition* $\mathrm{val} \circ g : \Gamma^\omega \to K$ is defined for all $u \in \Gamma^\omega$ by $(\mathrm{val} \circ g)(u) = \mathrm{val}(g(u))$. Given a quantitative ω-language $\mathbb{L} : \Gamma^\omega \to K$ and an ω-language $\mathcal{L} \subseteq \Gamma^\omega$, the intersection $\mathbb{L} \cap \mathcal{L} : \Gamma^\omega \to K$ is defined for all $u \in \mathcal{L}$ as $(\mathbb{L} \cap \mathcal{L})(u) = \mathbb{L}(u)$ and for all $u \in \Gamma^\omega \setminus \mathcal{L}$ as $(\mathbb{L} \cap \mathcal{L})(u) = \mathbb{0}$. Given a renaming $h : \Gamma \to \Sigma$, we say that an ω-language $\mathcal{L} \subseteq \Gamma^\omega$ is *h-unambiguous* if for all $w \in \Sigma^\omega$ there exists at most one $u \in \mathcal{L}$ such that $h(u) = w$.

Our Nivat decomposition theorem for WBA is the following.

Theorem 3.1. *Let Σ be an alphabet, $\mathbb{V} = (M, (K, +, \mathbb{0}), \mathrm{val})$ a valuation structure, and $\mathbb{L} : \Sigma^\omega \to K$ a quantitative ω-language. Then*

(a) $\mathbb{L}$ is unambiguously recognizable over $\mathbb{V}$ iff there exist an alphabet Γ, renamings $h : \Gamma \to \Sigma$ and $g : \Gamma \to M$, and a recognizable and h-unambiguous ω-language $\mathcal{L} \subseteq \Gamma^\omega$ such that $\mathbb{L} = h((\mathrm{val} \circ g) \cap \mathcal{L})$.

(b) $\mathbb{L}$ is nondeterministically recognizable over $\mathbb{V}$ iff there exist an alphabet Γ, renamings $h : \Gamma \to \Sigma$ and $g : \Gamma \to M$, and a recognizable ω-language $\mathcal{L} \subseteq \Gamma^\omega$ such that $\mathbb{L} = h((\mathrm{val} \circ g) \cap \mathcal{L})$.

3.1 Proof of Theorem 3.1

We start with part (b) of Theorem 3.1.

Lemma 3.2. *Let $\mathcal{A}$ be a WBA over Σ and $\mathbb{V}$. Then there exist an alphabet Γ, renaming $h : \Gamma \to \Sigma$ and $g : \Gamma \to M$, and a recognizable ω-language $\mathcal{L} \subseteq \Gamma^\omega$ such that $[\![\mathcal{A}]\!] = h((\mathrm{val} \circ g) \cap \mathcal{L})$.*

Proof. The idea is as in [11] to take the set of transitions as the extended alphabet Γ. Then, h maps every transition to its label and g maps every transition to its weight. Then, if in the underlying unweighted Büchi automaton we label every transition with itself, then we obtain the Büchi automaton accepting $\mathcal{L}$.

Formally, let $\mathcal{A} = (Q, I, T, F, \mathrm{wt})$. We may assume w.l.o.g. that $T \neq \emptyset$.We let $\Gamma = T$ and $h : \Gamma \to \Sigma$ be defined for all $t = (p, a, q) \in T$ as $h(t) = a$, and let $g : T \to M$ be defined for all $t \in T$ as $g(t) = \mathrm{wt}(t)$. We also define $\mathcal{L}$ by $\mathcal{L} = \{\rho = (t_i)_{i \in \mathbb{N}} \mid \rho \text{ is a run of } \mathcal{A}\}$.

First we show that $\mathcal{L}$ is recognizable. Indeed, consider the Büchi automaton $\mathcal{A}' = (Q, I, T', F)$ over Γ where $T' = \{(p, (p, a, q), q) \mid (p, a, q) \in T\}$. Then $\mathcal{L}(\mathcal{A}') = \mathcal{L}$ and hence $\mathcal{L}$ is recognizable. Finally we show that $[\![\mathcal{A}]\!] = h((\mathrm{val}^\omega \circ g) \cap \mathcal{L})$. Let $w \in \Sigma^\omega$. Then

$$h((\mathrm{val}^\omega \circ g) \cap \mathcal{L})(w) = \sum_{\substack{u \in \mathcal{L}, \\ h(u) = w}} \mathrm{val}^\omega(g(u)) = \sum_{\rho \in \mathrm{Run}_{\mathcal{A}}(w)} \mathrm{wt}_{\mathcal{A}}(\rho) = [\![\mathcal{A}]\!](w).$$

□

Now we turn to the implication ⇐.

Lemma 3.3. *Let Γ be an alphabet, $h : \Gamma \to \Sigma$ and $g : \Gamma \to M$ renamings, and $\mathcal{L} \subseteq \Gamma^\omega$ a recognizable ω-language. Then, the quantitative ω-language $h((\mathrm{val} \circ g) \cap \mathcal{L})$ is recognizable over $\mathbb{V}$.*

Proof. Since Büchi automata are not determinizable, the most challenging part in the proof is to show that recognizability of quantitative ω-languages is stable under intersection with recognizable ω-languages. Here we apply the result of [8] that recognizable ω-languages are recognizable by unambiguous Büchi automata.

Let $\mathcal{A}$ be an unambiguous Büchi automaton over Γ with $\mathcal{L}(\mathcal{A}) = \mathcal{L}$. If we associate with every transition (p, γ, q) of $\mathcal{A}$ the weight $g(\gamma) \in M$, then we obtain the WBA $\mathcal{B}$ over Γ and $\mathbb{V}$ with $[\![\mathcal{B}]\!] = (\mathrm{val} \circ g) \cap \mathcal{L}$. It remains to show that recognizable quantitative ω-languages are closed under renaming. For this, we apply the construction of Droste and Vogler [16]. Let $\mathcal{B} = (Q, I, T, F, \mathrm{wt})$. Then we construct the WBA $\mathcal{C} = (Q', I', T', F', \mathrm{wt}')$ over Σ and $\mathbb{V}$ defined as follows:

- $Q' = Q \times \Gamma$, $I' = I \times \{\gamma_0\}$ for some fixed $\gamma_0 \in \Gamma$, $F' = F \times \Gamma$;
- T' consists of all transitions $t = ((p, \gamma), a, (p', \gamma')) \in Q' \times \Sigma \times Q'$ such that $(p, \gamma', p') \in T$ and $h(\gamma') = a$. For such a transition t, we let $\mathrm{wt}'(t) = \mathrm{wt}(p, \gamma', p')$.

Then $h([\![\mathcal{B}]\!]) = [\![\mathcal{C}]\!]$. Hence the quantitative ω-language $h((\mathrm{val} \circ g) \cap \mathcal{L})$ is recognizable over $\mathbb{V}$. □

Then Theorem 3.1(b) follows immediately from Lemmas 3.2 and 3.3.

The proof of Theorem 3.1(a) relies on the same constructions as the proof of Theorem 3.1(b). Note that in the proof of Lemma 3.2, if $\mathcal{A}$ is unambiguous, the ω-language $\mathcal{L}$ is h-unambiguous. Note also that the WBA $\mathcal{B}$ in the proof of Lemma 3.3 is unambiguous but, in general, $\mathcal{C}$ is not. However, h-unambiguity of $\mathcal{L}$ guarantees that $\mathcal{C}$ is unambiguous.

3.2 Weighted Muller Automata

As a first application of Theorem 3.1 we show that WBA are expressively equivalent to *weighted Muller automata* which are defined as WBA with the difference that a set of accepting states $F \subseteq Q$ is replaced by a set $\mathcal{F} \subseteq 2^Q$ of sets of accepting states. Then, for an accepting run ρ, the set of all states, which are visited in ρ infinitely often, must be in $\mathcal{F}$.

Theorem 3.4. *Let Σ be an alphabet, $\mathbb{V} = (M, (K, +, \mathbb{0}), \mathrm{val})$ a valuation structure and $\mathbb{L} : \Sigma^\omega \to K$ a quantitative ω-language. Then $\mathbb{L} = [\![\mathcal{A}]\!]$ for some WBA $\mathcal{A}$ over Σ and $\mathbb{V}$ iff $\mathbb{L} = [\![\mathcal{A}']\!]$ for some weighted Muller automaton $\mathcal{A}'$ over Σ and $\mathbb{V}$.*

Theorem 3.4 extends the result of [15] for totally complete semirings. Whereas the proof of [15] was given by direct non-trivial automata transformation, our proof is based on the fact that weighted Muller automata permit the same decomposition as stated in Theorem 3.1 for WBA. The constructions for this case are much the same as the constructions of Theorem 3.1(b). We only have to replace $F \subseteq Q$ by $\mathcal{F} \subseteq 2^Q$ in the proofs and slightly modify the constructions of Lemma 3.3.

- It is well known that Muller automata are determinizable. Then, for the construction of the weighted Muller automaton for $(\mathrm{val} \circ g) \cap \mathcal{L}$ we use the fact that Muller and Büchi automata are expressively equivalent and take a deterministic Muller automaton recognizing $\mathcal{L}$.
- In the definition of $\mathcal{C}$ in the proof of Lemma 3.3 we replace F' by the Muller acceptance condition $\mathcal{F}'$ which consists of all sets $\{(q_1, \gamma_1), ..., (q_k, \gamma_k)\} \subseteq Q'$ such that $\{q_1, ..., q_k\} \in \mathcal{F}$ (a similar idea was used in [12]).

4 Weight Assignment Logic

4.1 Partial ω-words

Before we give a definition of the syntax and semantics of our new logic, we introduce some auxiliary notions about partial ω-words. Let X be an arbitrary non-empty set. A *partial ω-word* over X is a partial mapping $u : \mathbb{N} \dashrightarrow X$, i.e., $u : U \to X$ for some $U \subseteq \mathbb{N}$. Let $\mathrm{dom}(u) = U$, the *domain* of u. We denote by $X^\uparrow$ the set of all partial ω-words over X. Clearly, $X^\omega \subseteq X^\uparrow$. A *trivial ω-word* $\top \in X^\uparrow$ is the partial ω-word with $\mathrm{dom}(\top) = \emptyset$. For $u \in X^\uparrow$, $i \in \mathbb{N}$ and $x \in X$, the *update* $u[i/x] \in X^\uparrow$ is defined as $\mathrm{dom}(u[i/x]) = \mathrm{dom}(u) \cup \{i\}$, $u[i/x](i) = x$ and $u[i/x](i') = u(i')$ for all $i' \in \mathrm{dom}(u) \setminus \{i\}$. Let $\theta = (u_j)_{j \in J}$ be an arbitrary family of partial ω-words $u_j \in X^\uparrow$ where J is an arbitrary index set. We say that θ is *compatible* if, for all $j, j' \in J$ and $i \in \mathrm{dom}(u_j) \cap \mathrm{dom}(u_{j'})$, we have $u_j(i) = u_{j'}(i)$. If θ is compatible, then we define the *merging* $u := (\bigsqcap_{j \in J} u_j) \in X^\uparrow$ as $\mathrm{dom}(u) = \bigcup_{j \in J} \mathrm{dom}(u_j)$ and, for all $i \in \mathrm{dom}(u)$, $u(i) = u_j(i)$ whenever $i \in \mathrm{dom}(u_j)$ for some $j \in J$. Let $\theta = \{u_j\}_{j \in \{1,2\}}$ be compatible. Then, we write $u_1 \uparrow u_2$. Clearly, the relation $\uparrow$ is reflexive and symmetric. In the case $u_1 \uparrow u_2$, for $\bigsqcap_{j \in \{1,2\}} u_j$ we will also use notation $u_1 \sqcap u_2$.

Example 4.1. Let $X = \{a, b\}$ with $a \neq b$ and $u_1 = a^\omega \in X^\uparrow$. Let $u_2 \in X^\uparrow$ be the partial ω-word whose domain $\mathrm{dom}(u_2)$ is the set of all odd natural numbers and $u_2(i) = a$ for all $i \in \mathrm{dom}(u_2)$. Let $u_3 \in X^\uparrow$ be the partial ω-word such that $\mathrm{dom}(u_3)$ is the set of all even natural numbers and $u_3(i) = b$ for all $i \in \mathrm{dom}(u_3)$. Then $u_1 \uparrow u_2$ and $u_2 \uparrow u_3$, but $\neg(u_1 \uparrow u_3)$. This shows in particular that the relation $\uparrow$ is not transitive if X is not a singleton set. Then, $u_1 \sqcap u_2 = a^\omega$ and $u_2 \sqcap u_3 = (ba)^\omega$.

4.2 WAL: Syntax and Semantics

Let V_1 be a countable set of *first-order variables* and V_2 a countable set of *second-order variables* such that $V_1 \cap V_2 = \emptyset$. Let $V = V_1 \cup V_2$. Let Σ be an alphabet and $\mathbb{V} = (M, (K, +, \mathbb{0}), \mathrm{val})$ a valuation structure. We also consider a designated element $\mathbb{1} \in M$ which we call the *default weight*. We denote the pair $(\mathbb{V}, \mathbb{1})$ by $\mathbb{V}_{\mathbb{1}}$. The set $\mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$ of formulas of *weight assignment logic* over Σ and $\mathbb{V}_{\mathbb{1}}$ is given by the grammar

$$\varphi \ ::= \ P_a(x) \mid x = y \mid \ x < y \mid X(x) \mid x \mapsto m \mid \varphi \Rightarrow \varphi \mid \varphi \sqcap \varphi \mid \sqcap x.\varphi \mid \sqcap X.\varphi$$

where $a \in \Sigma$, $x, y \in V_1$, $X \in V_2$ and $m \in M$. Such a formula φ is called a *weight assignment formula*.

Let $\varphi \in \mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$. We denote by $\textsc{Const}(\varphi) \subseteq M$ the set of all weights $m \in M$ occurring in φ. The set $\textsc{Free}(\varphi) \subseteq V$ of *free variables* of φ is defined to be the set of all variables $\mathcal{X} \in V$ which appear in φ and are not bound by any quantifier $\sqcap\mathcal{X}$. We say that φ is a *sentence* if $\textsc{Free}(\varphi) = \emptyset$.

Note that the merging as defined before is a partially defined operation, i.e., it is defined only for compatible families of partial ω-words. In order to extend it to a totally defined operation, we fix an element $\bot \notin M^{\uparrow}$ which will mean the undefined value. Let $M^{\uparrow}_{\bot} = M^{\uparrow} \cup \{\bot\}$. Then, for any family $\theta = (u_j)_{j \in J}$ with $u_j \in M^{\uparrow}_{\bot}$, such that either $\theta \in (M^{\uparrow})^J$ is not compatible or $\theta \in (M^{\uparrow}_{\bot})^J \setminus (M^{\uparrow})^J$, we let $\bigsqcap_{j \in J} u_j = \bot$.

For any ω-word $w \in \Sigma^{\omega}$, a *w-assignment* is a mapping $\sigma : V \to \mathrm{dom}(w) \cup 2^{\mathrm{dom}(w)}$ mapping first-order variables to elements in $\mathrm{dom}(w)$ and second-order variables to subsets of $\mathrm{dom}(w)$. For a first-order variable x and a position $i \in \mathbb{N}$, the w-assignment $\sigma[x/i]$ is defined on $V \setminus \{x\}$ as σ, and we let $\sigma[x/i](x) = i$. For a second-order variable X and a subset $I \subseteq \mathbb{N}$, the w-assignment $\sigma[X/I]$ is defined similarly. Let Σ^{ω}_V denote the set of all pairs (w, σ) where $w \in \Sigma^{\omega}$ and σ is a w-assignment. We will denote such pairs (w, σ) by w_{σ}.

The semantics of $\mathbf{WAL}$-formulas is defined in two steps: by means of the auxiliary and proper semantics. Let $\varphi \in \mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$. The *auxiliary semantics* of φ is the mapping $\langle\langle \varphi \rangle\rangle : \Sigma^{\omega}_V \to M^{\uparrow}_{\bot}$ defined for all $w_{\sigma} \in \Sigma^{\omega}_V$ with $w = (a_i)_{i \in \mathbb{N}}$ as shown in Table 1. Note that the definition of $\langle\langle .. \rangle\rangle$ does not employ $+$ and val. The *proper semantics* $[\![\varphi]\!] : \Sigma^{\omega}_V \to K$ operates on the auxiliary semantics $\langle\langle \varphi \rangle\rangle$ as follows. Let $w_{\sigma} \in \Sigma^{\omega}_V$. If $\langle\langle \varphi \rangle\rangle(w_{\sigma}) \in M^{\uparrow}$, then we assign the default weight to all undefined positions in $\mathrm{dom}(\langle\langle \varphi \rangle\rangle(w_{\sigma}))$ and evaluate the obtained sequence using val. Otherwise, if $\langle\langle \varphi \rangle\rangle(w_{\sigma}) = \bot$, we put $[\![\varphi]\!](w_{\sigma}) = \mathbb{0}$. Note that if $\varphi \in \mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$ is a sentence, then the values $\langle\langle \varphi \rangle\rangle(w_{\sigma})$ and $[\![\varphi]\!](w_{\sigma})$ do not depend on σ and we consider the auxiliary semantics of φ as the mapping $\langle\langle \varphi \rangle\rangle : \Sigma^{\omega} \to M^{\uparrow}_{\bot}$ and the proper semantics of φ as the quantitative ω-language $[\![\varphi]\!] : \Sigma^{\omega} \to K$. Note that $+$ was not needed for the semantics of $\mathbf{WAL}$-formulas. This operation will be needed in the next section for the extension of $\mathbf{WAL}$. We say that a quantitative ω-language $\mathbb{L} : \Sigma^{\omega} \to K$ is $\mathbf{WAL}$-*definable* over $\mathbb{V}$ if there exist a default weight $\mathbb{1} \in M$ and a sentence $\varphi \in \mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$ such that $[\![\varphi]\!] = \mathbb{L}$.

Example 4.2. Consider a valuation structure $\mathbb{V} = (M, (K, +, \mathbb{0}), \mathrm{val})$ and a default weight $\mathbb{1} \in M$. Consider an alphabet $\Sigma = \{a, b, ...\}$ of actions. We assume that the

$$
\begin{aligned}
\langle\!\langle P_a(x)\rangle\!\rangle(w_\sigma) &= \begin{cases} \top, & a_{\sigma(x)} = a \\ \bot, & \text{otherwise} \end{cases} \\
\langle\!\langle x = y\rangle\!\rangle(w_\sigma) &= \begin{cases} \top, & \sigma(x) = \sigma(y) \\ \bot, & \text{otherwise} \end{cases} \\
\langle\!\langle x < y\rangle\!\rangle(w_\sigma) &= \begin{cases} \top, & \sigma(x) < \sigma(y) \\ \bot, & \text{otherwise} \end{cases} \\
\langle\!\langle X(x)\rangle\!\rangle(w_\sigma) &= \begin{cases} \top, & \sigma(x) \in \sigma(X) \\ \bot, & \text{otherwise} \end{cases}
\end{aligned}
\qquad
\begin{aligned}
\langle\!\langle x \mapsto m\rangle\!\rangle(w_\sigma) &= \top[\sigma(x)/m] \\
\langle\!\langle \varphi_1 \Rightarrow \varphi_2\rangle\!\rangle(w_\sigma) &= \begin{cases} \langle\!\langle \varphi_2\rangle\!\rangle(w_\sigma), & \langle\!\langle \varphi_1\rangle\!\rangle(w_\sigma) = \top \\ \top, & \text{otherwise} \end{cases} \\
\langle\!\langle \varphi_1 \sqcap \varphi_2\rangle\!\rangle(w_\sigma) &= \langle\!\langle \varphi_1\rangle\!\rangle(w_\sigma) \sqcap \langle\!\langle \varphi_2\rangle\!\rangle(w_\sigma) \\
\langle\!\langle \sqcap x.\varphi\rangle\!\rangle(w_\sigma) &= \textstyle\bigsqcap_{i\in\mathrm{dom}(w)} \langle\!\langle \varphi\rangle\!\rangle(w_{\sigma[x/i]}) \\
\langle\!\langle \sqcap X.\varphi\rangle\!\rangle(w_\sigma) &= \textstyle\bigsqcap_{I\subseteq\mathrm{dom}(w)} \langle\!\langle \varphi\rangle\!\rangle(w_{\sigma[X/I]})
\end{aligned}
$$

Table 1. The auxiliary semantics of $\mathbf{WAL}$-formulas

cost of a is $c(a) \in M$, the cost of b is $c(b) \in M$, and the costs of all other actions x in Σ are equal to $c(x) = \mathbb{1}$ (which can mean, e.g., that these actions do not invoke any costs). Then every ω-word w induces the ω-word of costs. We want to construct a sentence of our WAL which for every such an ω-word will evaluate its sequence of costs using val. The desired sentence $\varphi \in \mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$ is

$$\varphi = \sqcap x.([P_a(x) \Rightarrow (x \mapsto c(a))] \sqcap [P_b(x) \Rightarrow (x \mapsto c(b))]).$$

Then, for every $w = (a_i)_{i\in\mathbb{N}} \in \Sigma^\omega$, the auxiliary semantics $\langle\!\langle \varphi\rangle\!\rangle(w)$ is the partial ω-word over M where all positions $i \in \mathbb{N}$ with $a_i = a$ are labelled by $c(a)$, all positions with $a_i = b$ are labelled by $c(b)$, and the labels of all other positions are undefined. Then, the proper semantics $[\![\varphi]\!](w)$ assigns $\mathbb{1}$ to all positions with undefined labels and evaluates it by means of val.

4.3 WAL: Relation to MSO Logic

Let Σ be an alphabet. We consider monadic second-order logic $\mathbf{MSO}(\Sigma)$ over ω-words to be the set of formulas

$$\varphi \ ::= \ P_a(x) \mid x = y \mid x < y \mid X(x) \mid \varphi \wedge \varphi \mid \neg\varphi \mid \forall x.\varphi \mid \forall X.\varphi$$

where $a \in \Sigma$, $x, y \in V_1$ and $X \in V_2$. For $w_\sigma \in \Sigma_V^\omega$, the satisfaction relation $w_\sigma \models \varphi$ is defined as usual. The usual formulas of the form $\varphi_1 \vee \varphi_2$, $\exists\mathcal{X}.\varphi$ with $\mathcal{X} \in V$, $\varphi_1 \Rightarrow \varphi_2$ and $\varphi_1 \Leftrightarrow \varphi_2$ can be expressed using $\mathbf{MSO}$-formulas.

For any formula $\varphi \in \mathbf{MSO}(\Sigma)$, let $W(\varphi)$ denote the $\mathbf{WAL}$-formula obtained from φ by replacing $\wedge$ by $\sqcap$, $\forall\mathcal{X}$ (with $\mathcal{X} \in V$) by $\sqcap\mathcal{X}$, and every subformula $\neg\psi$ by $\psi \Rightarrow \mathbf{false}$. Here $\mathbf{false}$ can be considered as abbreviation of the sentence $\sqcap x.(x < x)$. Note that $W(\varphi)$ does not contain any assignment formulas $x \mapsto m$ and $\langle\!\langle W(\varphi)\rangle\!\rangle(w_\sigma) \in \{\top, \bot\}$ for every $w_\sigma \in \Sigma_V^\omega$. Moreover, it can be easily shown by induction on the structure of φ that, for all $w_\sigma \in \Sigma_V^\omega$: $w_\sigma \models \varphi$ iff $\langle\!\langle W(\varphi)\rangle\!\rangle(w_\sigma) = \top$. This shows that MSO logic on infinite words is subsumed by $\mathbf{WAL}$. For the formulas which do not contain any assignments of the form $x \mapsto m$, the merging $\sqcap$ can be considered as the usual conjunction and the merging quantifiers $\sqcap\mathcal{X}$ as the usual universal

$$
\begin{aligned}
[\![\sqcup x.\varphi]\!](w_\sigma) \; &= \sum \big([\![\varphi]\!](w_{\sigma[x/i]}) \mid i \in \mathrm{dom}(w)\big) \\
[\![\sqcup X.\varphi]\!](w_\sigma) &= \sum \big([\![\varphi]\!](w_{\sigma[X/I]}) \mid I \subseteq \mathrm{dom}(w)\big)
\end{aligned}
$$

Table 2. The semantics of $\mathbf{eWAL}$-formulas

quantifiers $\forall\mathcal{X}$. Moreover, $\top$ corresponds to the boolean true value and $\bot$ to the boolean false value.

For a $\mathbf{WAL}$-formula φ, we will consider $\neg\varphi$ as abbreviation for $\varphi \Rightarrow \mathbf{false}$.

4.4 Extended WAL

Here we extend $\mathbf{WAL}$ with weighted existential quantification over free variables in $\mathbf{WAL}$-formulas. Let Σ be an alphabet, $\mathbb{V} = (M, (K, +, \mathbb{0}), \mathrm{val})$ a valuation structure and $\mathbb{1} \in M$ a default weight. The set $\mathbf{eWAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$ of formulas of *extended weight assignment logic* over Σ and $\mathbb{V}_{\mathbb{1}}$ consists of all formulas of the form $\sqcup\mathcal{X}_1. \; ... \; \sqcup\mathcal{X}_k.\varphi$ where $k \geq 0$, $\mathcal{X}_1, ..., \mathcal{X}_k \in V$ and $\varphi \in \mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$. Given a formula $\varphi \in \mathbf{eWAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$, the *semantics* of φ is the mapping $[\![\varphi]\!] : \Sigma_V^\omega \to K$ defined inductively as follows. If $\varphi \in \mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$, then $[\![\varphi]\!]$ is defined as the proper semantics for $\mathbf{WAL}$. If φ contains a prefix $\sqcup x$ with $x \in V_1$ or $\sqcup X$ with $X \in V_2$, then, for all $w_\sigma \in \Sigma_V^\omega$, $[\![\varphi]\!](w_\sigma)$ is defined inductively as shown in Table 2. Again, if φ is a sentence, then we can consider its semantics as the quantitative ω-language $[\![\varphi]\!] : \Sigma^\omega \to K$. We say that a quantitative ω-language $\mathbb{L} : \Sigma^\omega \to K$ is $\mathbf{eWAL}$-*recognizable* over $\mathbb{V}$ if there exist a default weight $\mathbb{1} \in M$ and a sentence $\varphi \in \mathbf{eWAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$ such that $[\![\varphi]\!] = \mathbb{L}$.

Example 4.3. Let $\Sigma = \{a\}$ be a singleton alphabet, $\mathbb{V} = \mathbb{V}^{\mathrm{DISC}}$ as defined in Example 2.3(b). Assume that, for every position of an ω-word, we can either assign to this position the cost 5 and the discounting factor 0.5 or we assign the cost the smaller cost 2 and the bigger discounting factor 0.75. After that we compute the discounted sum using the valuation function of $\mathbb{V}^{\mathrm{DISC}}$. We are interested in the infimal value of this discounted sum. We can express it by means of the $\mathbf{eWAL}$-formula

$$\varphi = \sqcup X.\sqcap x.([X(x) \Rightarrow (x \mapsto (5, 0.5))] \sqcap [(\neg X(x)) \Rightarrow (x \mapsto (2, 0.75))])$$

i.e. $[\![\varphi]\!](a^\omega)$ is the desired infimal value.

5 Expressiveness Equivalence Result

In this section we state and prove the main result of this paper.

Theorem 5.1. *Let Σ be an alphabet, $\mathbb{V} = (M, (K, +, \mathbb{0}), \mathrm{val})$ a valuation structure and $\mathbb{L} : \Sigma^\omega \to K$ a quantitative ω-language. Then*

(a) $\mathbb{L}$ is $\mathbf{WAL}$-definable over $\mathbb{V}$ iff $\mathbb{L}$ is unambiguously recognizable over $\mathbb{V}$.
(b) $\mathbb{L}$ is $\mathbf{eWAL}$-definable over $\mathbb{V}$ iff $\mathbb{L}$ is recognizable over $\mathbb{V}$.

5.1 Unambiguous Case: Definability Implies Recognizability

In this subsection, we prove part (a) of Theorem 5.1. First we show $\mathbf{WAL}$-definability implies unambiguous recognizability. We establish a decomposition of $\mathbf{WAL}$-formulas in a similar manner as it was done for unambiguous WBA in Theorem 3.1 (a), i.e., we separate weighted part of $\mathbf{WAL}$ from its unweighted part. Then applying the classical Büchi theorem and our Nivat Theorem 3.1(a), we obtain that $\mathbb{L}$ is recognizable over $\mathbb{V}$.

Lemma 5.2. *Let $\varphi \in \mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$ be a sentence. Then there exist an alphabet Γ, renamings $h : \Gamma \to \Sigma$ and $g : \Gamma \to M$, and a sentence $\beta \in \mathbf{MSO}(\Gamma)$ such that $[\![\varphi]\!] = h((\mathrm{val} \circ g) \cap \mathcal{L}(\beta))$.*

The proof of this lemma will be given in the rest of this subsection. Let $\# \notin M$ be a symbol which we will use to mark all positions whose labels are undefined in the auxiliary semantics of $\mathbf{WAL}$-formulas. Let $\Delta_\varphi = \textsc{Const}(\varphi) \cup \{\#\}$. Then our extended alphabet will be $\Gamma = \Sigma \times \Delta_\varphi$. We define the renamings h, g as follows. For all $u = (a, b) \in \Gamma$, we let $h(u) = a$, $g(u) = b$ if $b \in M$, and $g(u) = \mathbb{1}$ if $m = \#$. The main difficulty is to construct the sentence β. For any ω-word $w = (a_i)_{i \in \mathbb{N}} \in \Sigma^\omega$ and any partial ω-word $\eta \in (\textsc{Const}(\varphi))^\uparrow$, we encode the pair (w, η) as the ω-word $\mathrm{code}(w, \eta) = ((a_i, b_i))_{i \in \mathbb{N}} \in \Gamma^\omega$ where, for all $i \in \mathrm{dom}(\eta)$, $b_i = \eta(i)$ and, for all $i \in \mathbb{N} \setminus \mathrm{dom}(\eta)$, $b_i = \#$. In other words, we will consider ω-words of Γ as convolutions of ω-words over Σ with the encoding of the auxiliary semantics of φ.

The construction of β is based on the following technical lemma.

Lemma 5.3. *For every subformula ζ of φ, there exists a formula $\Phi(\zeta) \in \mathbf{MSO}(\Sigma \times \Delta_\varphi)$ such that $\textsc{Free}(\Phi(\zeta)) = \textsc{Free}(\zeta)$ and, for all $w_\sigma \in \Sigma_V^\omega$ and $\eta \in (\textsc{Const}(\varphi))^\uparrow$, we have: $\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \eta$ iff $(\mathrm{code}(w, \eta))_\sigma \models \Phi(\zeta)$.*

Note that $\langle\!\langle \varphi \rangle\!\rangle(w_\sigma) = \eta$ means in particular that $\langle\!\langle \varphi \rangle\!\rangle(w_\sigma) \neq \bot$.

Proof. Let $Y \in V_2$ be a fresh variable which does not occur in φ. First, we define inductively the formula $\Phi_Y(\zeta) \in \mathbf{MSO}(\Gamma)$ with $\textsc{Free}(\Phi_Y(\zeta)) = \textsc{Free}(\zeta) \cup \{Y\}$ which describes the connection between the input ω-word w and the output partial ω-word η; here the variable Y keeps track of the domain of η.

- For $\zeta = P_a(x)$, we let

  $$\Phi_Y(\zeta) = \bigvee_{b \in \Delta_\varphi} P_{(a,b)}(x) \wedge Y(\emptyset)$$

  where $Y(\emptyset)$ is abbreviation for $\forall y. \neg Y(y)$. Here we demand that the first component of the letter at position x is a and the second component is an arbitrary letter from Δ_φ and that the auxiliary semantics of ζ is the trivial partial ω-word $\top$.
- Let ζ be one of the formulas of the form $x = y$, $x < y$ or $X(x)$. Then, we let

  $$\Phi_Y(\zeta) = \zeta \wedge Y(\emptyset).$$

– For $\zeta = (x \mapsto m)$, we let

$$\Phi_Y(\zeta) = \bigvee_{a \in \Sigma} P_{(a,m)}(x) \wedge \forall y.(Y(y) \Leftrightarrow x = y).$$

This formula describes that position x of η must be labelled by m and all other positions are unlabelled.

– Let $\zeta = (\zeta_1 \Rightarrow \zeta_2)$. Let $Z \in V_2$ be a fresh variable. Consider the formula $\kappa = \exists Z.[\Phi_Z(\zeta_1) \wedge Z(\emptyset)]$ which checks whether the value of the auxiliary semantics of ζ_1 is $\top$. Then, we let

$$\Phi_Y(\zeta) = (\kappa \wedge \Phi_Y(\zeta_2)) \vee (\neg\kappa \wedge Y(\emptyset)).$$

– Let $\zeta = \zeta_1 \sqcap \zeta_2$. Let $Y_1, Y_2 \in V_2$ be two fresh distinct variables. Then we let

$$\Phi_Y(\zeta) = \exists Y_1.\exists Y_2.(\Phi_{Y_1}(\zeta_1) \wedge \Phi_{Y_2}(\zeta_2) \wedge [Y = Y_1 \cup Y_2]).$$

Here $Y = Y_1 \cup Y_2$ is considered as abbreviation for the MSO-formula $\forall y.(Y(y) \Leftrightarrow [Y_1(y) \vee Y_2(y)])$.

– The most interesting case is a formula of the form $\zeta = \sqcap\mathcal{X}.\zeta'$ with $\mathcal{X} \in V$. Here, every value of $\mathcal{X}$ induces its own value of $Y(\mathcal{X})$ and we have to merge infinitely many partial ω-words, i.e., to express that Y is the infinite union of $Y(\mathcal{X})$ over all sets $\mathcal{X}$. We can show that Y must be the minimal set which satisfies the formula $\xi(Y) = \forall\mathcal{X}.\exists Y'.(\Phi_{Y'}(\zeta') \wedge (Y' \subseteq Y))$ where $Y' \in V_2$ is a fresh variable. Then, we let

$$\Phi_Y(\zeta) = \xi(Y) \wedge \forall Z.(\xi(Z) \Rightarrow (Y \subseteq Z))$$

where $Z \in V_2$ is a fresh variable.

Let $w = (a_i)_{i \in \mathbb{N}} \in \Sigma^\omega$, σ be a w-assignment and $\eta \in (\textsc{Const}(\varphi))^\uparrow$. For $R \subseteq \mathbb{N}$, let $\eta|_R \in (\textsc{Const}(\varphi))^\uparrow$ be defined as $\mathrm{dom}(\eta|_R) = R \cap \mathrm{dom}(\eta)$ and $\eta|_R(i) = \eta(i)$ for all $i \in \mathrm{dom}(\eta|_R)$. Now we show by induction on the structure of ζ that

$$(\mathrm{code}(w,\eta))_\sigma \models \Phi_Y(\zeta) \;\text{ iff }\; \sigma(Y) \subseteq \mathrm{dom}(\eta) \text{ and } \langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \eta|_{\sigma(Y)}. \qquad (1)$$

– Let $\zeta = P_a(x)$.
  - Assume that $(\mathrm{code}(w,\eta))_\sigma \models \Phi_Y(\zeta)$. Then $a_{\sigma(x)} = a$ and $\sigma(Y) = \emptyset$. Hence $\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \top = \eta|_\emptyset$ and $\emptyset = \sigma(Y) \subseteq \mathrm{dom}(\eta)$.
  - Conversely, assume that $\sigma(Y) \subseteq \mathrm{dom}(\eta)$ and $\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \eta|_{\sigma(Y)}$. Then $\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \top$ which implies $a_{\sigma(x)} = a$ and $\sigma(Y) = \emptyset$. Then $(\mathrm{code}(w,\eta))_\sigma \models \Phi_Y(\zeta)$.

– Let ζ be one of the formulas $x < y$, $x = y$ and $X(x)$.
  - Assume that $(\mathrm{code}(w,\eta))_\sigma \models \Phi_Y(\zeta)$. Then $(\mathrm{code}(w,\eta))_\sigma \models \zeta$ and $\sigma(Y) = \emptyset$. Since $(\mathrm{code}(w,\eta))_\sigma \models \zeta$ implies $w_\sigma \models \zeta$, we obtain $\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \top = \eta|_\emptyset$ and $\emptyset = \sigma(Y) \subseteq \mathrm{dom}(\eta)$.
  - Conversely, assume that $\sigma(Y) \subseteq \mathrm{dom}(\eta)$ and $\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \eta|_{\sigma(Y)}$. Then $\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \top$ which implies $w_\sigma \models \zeta$ and $\sigma(Y) = \emptyset$. Then, $(\mathrm{code}(w,\eta))_\sigma \models \zeta$ and $\sigma(Y) = \emptyset$. Hence $(\mathrm{code}(w,\eta))_\sigma \models \Phi_Y(\zeta)$.

– Let $\zeta = (x \mapsto m)$ with $m \in \textsc{Const}(\varphi)$ (since ζ is a subformula of φ).

    - Assume that $(\mathrm{code}(w,\eta))_\sigma \models \varPhi_Y(\zeta)$. Then $\sigma(x) \in \mathrm{dom}(\eta)$, $\eta(\sigma(x)) = m$ and $\sigma(Y) = \{\sigma(x)\}$. Hence $\langle\!\langle \psi \rangle\!\rangle(w_\sigma) = \top[\sigma(x)/m] = \eta|_{\sigma(Y)}$ and $\{\sigma(x)\} = Y \subseteq \mathrm{dom}(\eta)$.
    - Conversely, assume that the right hand side of (1) holds true. Then $\eta|_{\sigma(Y)} = \top[\sigma(x)/m]$. Since $\sigma(Y) \subseteq \mathrm{dom}(\eta)$, we have $\sigma(Y) = \{\sigma(x)\}$. Moreover, $\eta(\sigma(x)) = m$. Then the left hand side of (1) also holds true.
- Let $\zeta = (\zeta_1 \Rightarrow \zeta_2)$.
    - Assume that the left hand side of (1) holds true. Then one of the following cases is possible.
        - ∗ $(\mathrm{code}(w,\eta))_\sigma \models \kappa \wedge \varPhi_Y(\zeta_2)$. Then, $(\mathrm{code}(w,\eta))_{\sigma[Y/\emptyset]} \models \varPhi_Y(\zeta_1)$ and $(\mathrm{code}(w,\eta))_\sigma \models \varPhi_Y(\zeta_2)$. Then by induction hypothesis for ζ_1 and ζ_2 we have: $\langle\!\langle \zeta_1 \rangle\!\rangle(w_\sigma) = \eta|_\emptyset = \top$, $\sigma(Y) \subseteq \mathrm{dom}(\eta)$ and $\langle\!\langle \zeta_2 \rangle\!\rangle(w_\sigma) = \eta|_{\sigma(Y)}$. This implies $\sigma(Y) \subseteq \mathrm{dom}(\eta)$ and $\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \langle\!\langle \zeta_2 \rangle\!\rangle(w_\sigma) = \eta|_{\sigma(Y)}$. Hence the right hand side of (1) holds true.
        - ∗ $(\mathrm{code}(w,\eta))_\sigma \models \neg\kappa \wedge Y(\emptyset)$. Then, $(\mathrm{code}(w,\eta))_{\sigma[Y/\emptyset]} \not\models \varPhi_Y(\zeta_1)$ and $\sigma(Y) = \emptyset$. Then by induction hypothesis for ζ_1 we have $\langle\!\langle \eta_1 \rangle\!\rangle(w_\sigma) \neq \eta|_\emptyset = \top$. Then $\emptyset = \sigma(Y) \subseteq \mathrm{dom}(\eta)$ and $\langle\!\langle \zeta \rangle\!\rangle = \top = \eta|_{\sigma(Y)}$. Then the right hand side of (1) holds true.
    - Now assume that the right hand side of (1) holds true. Then one of the following cases is possible.
        - ∗ $\langle\!\langle \zeta_1 \rangle\!\rangle(w_\sigma) = \top = \eta|_\emptyset$. Then by induction hypothesis for ζ_1 we have $(\mathrm{code}(w,\eta))_{\sigma[Y/\emptyset]} \models \varPhi_Y(\zeta_1)$ and hence $(\mathrm{code}(w,\eta))_\sigma \models \kappa$. Moreover, $\eta|_{\sigma(Y)} = \langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \langle\!\langle \zeta_2 \rangle\!\rangle(w_\sigma)$ and $\sigma(Y) \subseteq \mathrm{dom}(\eta)$. Then by induction hypothesis for ζ_2 we obtain $(\mathrm{code}(w,\eta))_\sigma \models \varPhi_Y(\zeta_1)$. Then we have $(\mathrm{code}(w,\eta))_\sigma \models \kappa \wedge \varPhi_Y(\zeta_1)$ and hence $(\mathrm{code}(w,\eta))_\sigma \models \varPhi_Y(\zeta)$.
        - ∗ $\langle\!\langle \zeta_1 \rangle\!\rangle(w_\sigma) \neq \top = \eta|_\emptyset$. Then by induction hypothesis for ζ_1 we have $(\mathrm{code}(w,\eta))_{\sigma[Y/\emptyset]} \not\models \varPhi_Y(\zeta_1)$ and hence $(\mathrm{code}(w,\eta))_\sigma \not\models \kappa$. Moreover, $\eta|_{\sigma(Y)} = \langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \top = \eta|_\emptyset$ and $\sigma(Y) \subseteq \mathrm{dom}(\eta)$ which implies $\sigma(Y) = \emptyset$. Then $(\mathrm{code}(w,\eta))_\sigma \models \neg\kappa \wedge Y(\emptyset)$ and hence $(\mathrm{code}(w,\eta))_\sigma \models \varPhi_Y(\zeta)$.
- Let $\zeta = \zeta_1 \sqcap \zeta_2$.
    - Assume that the left hand side of (1) holds. Then there exist subsets $R_1, R_2 \subseteq \mathrm{dom}(w)$ such that:
        - ∗ $\sigma(Y) = R_1 \cup R_2$,
        - ∗ $(\mathrm{code}(w,\eta))_{\sigma[Y/R_1]} \models \varPhi_Y(\zeta_1)$,
        - ∗ $(\mathrm{code}(w,\eta))_{\sigma[Y/R_2]} \models \varPhi_Y(\zeta_2)$.

      Then by induction hypothesis for ζ_1 and ζ_2 we have:
        - ∗ $R_1 \subseteq \mathrm{dom}(\eta)$ and $\langle\!\langle \zeta_1 \rangle\!\rangle(w_\sigma) = \eta|_{R_1}$,
        - ∗ $R_2 \subseteq \mathrm{dom}(\eta)$ and $\langle\!\langle \zeta_2 \rangle\!\rangle(w_\sigma) = \eta|_{R_2}$.

      Then $\sigma(Y) \subseteq \mathrm{dom}(\eta)$ and, since $\eta|_{R_1}$ and $\eta|_{R_2}$ are compatible partial ω-words, we have $\langle\!\langle \zeta_1 \sqcap \zeta_2 \rangle\!\rangle(w_\sigma = \eta|_{R_1} \sqcap \eta|_{R_2} = \eta|_{\sigma(Y)}$. This shows that the right hand side of (1) also holds true.
    - Conversely, assume that the right hand side of (1) holds. Let $\eta_1 = \langle\!\langle \zeta_1 \rangle\!\rangle(w_\sigma)$ and $\eta_2 = \langle\!\langle \zeta_2 \rangle\!\rangle(w_\sigma)$. Then $\eta|_{\sigma(Y)} = \sigma_1 \sqcap \sigma_2$. Moreover, there exist $R_1, R_2 \subseteq \mathrm{dom}(w)$ such that:
        - ∗ $R_1 \cup R_2 = \sigma(Y)$,

    - ∗ $\eta_1 = \eta|_{R_1}$ and $\eta = \eta|_{R_2}$.

      Since $R_1, R_2 \subseteq \sigma(Y) \subseteq \mathrm{dom}(\eta)$, by induction hypothesis we have $(\mathrm{code}(w,\eta))_{\sigma[Y_i/R_i]} \models \varPhi_Y(\zeta_i)$ for $i \in \{1,2\}$. Since Y_2 does not occur in $\varPhi_Y(\zeta_1)$ and Y_1 does not occur in $\varPhi_Y(\zeta_2)$, we have $\mathrm{code}(w,\eta)_{\sigma[Y_1/R_1][Y_2/R_2]} \models \varPhi_Y(\zeta_i)$ for $i \in \{1,2\}$. Then the left hand side of (1) holds.

- Let $\zeta = \sqcap x.\zeta'$ with $x \in V_1$.
  - Assume that $(\mathrm{code}(w,\eta))_\sigma \models \varPhi_Y(\zeta)$. Then $(\mathrm{code}(w,\eta))_\sigma \models \xi(Y)$. This means that for all $i \in \mathrm{dom}(w)$ there exists a subset $R_i \subseteq \sigma(Y)$ such that $(\mathrm{code}(w,\eta))_{\sigma[x/i][Y'/R_i]} \models \varPhi_Y(\zeta')$. Then by induction hypothesis for all $i \in \mathrm{dom}(w)$ we have: $R_i \subseteq \mathrm{dom}(\eta)$ and $\langle\!\langle \zeta' \rangle\!\rangle(w_{\sigma[x/i]}) = \eta|_{R_i}$. Let $R = \bigcup_{i \in \mathrm{dom}(w)} R_i$. Then, $(\mathrm{code}(w,\eta))_{\sigma[Z/R]} \models \xi(Z)$. Since $(\mathrm{code}(w,\eta))_\sigma \models \forall Z.(\xi(Z) \Rightarrow (Y \subseteq Z))$, we obtain $\sigma(Y) \subseteq R$. Hence $R = \sigma(Y)$ and

    $$\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \bigsqcap_{i \in \mathrm{dom}(w)} \eta|_{R_i} = \eta|_R = \eta|_{\sigma(Y)}.$$

    Finally, $\sigma(Y) = \bigcup_{i \in \mathrm{dom}(w)} R_i \subseteq \mathrm{dom}(\eta)$. This shows that the right hand side of (1) holds true.
  - Conversely, assume that the right hand side of (1) holds. Then there exists a family $(R_i)_{i \in \mathrm{dom}(w)}$ of subsets $R_i \subseteq \mathrm{dom}(Y) \subseteq \mathrm{dom}(\eta)$ such that $\bigcup_{i \in \mathrm{dom}(w)} R_i = \sigma(Y)$ and, for all $i \in \mathrm{dom}(w)$, $\langle\!\langle \zeta' \rangle\!\rangle(w_{\sigma[x/i]}) = \eta|_{R_i}$. Then it is easy to see by induction hypothesis that, for all $i \in \mathrm{dom}(w)$, $(\mathrm{code}(w,\eta))_{[x/i][Y'/R_i]} \models \varPhi_{Y'}(\zeta')$. Then $(\mathrm{code}(w,\eta))_\sigma \models \xi(Y)$. It remains to show that

    $$(\mathrm{code}(w,\sigma))_\sigma \models \forall Z.(\xi(Z) \Rightarrow (Y \subseteq Z)).$$

    Indeed, let $Q \subseteq \mathrm{dom}(w)$ with $(\mathrm{code}(w,\eta))_{\sigma[Z/Q]} \models \xi(Z)$. Then for all $i \in \mathrm{dom}(w)$ there exists a subset $Q_i \subseteq Q$ with $(\mathrm{code}(w,\eta))_{\sigma[x/i][Y'/Q_i]} \models \varPhi_{Y'}(\zeta')$. Then by induction hypothesis for all $i \in \mathrm{dom}(w)$ we have $Q_i \subseteq \mathrm{dom}(\eta)$ and

    $$\eta|_{Q_i} = \langle\!\langle \zeta' \rangle\!\rangle(w_{\sigma[x/i]}) = \eta|_{R_i}.$$

    Hence $Q_i = R_i$ for all $i \in \mathrm{dom}(w)$, and

    $$\sigma(Y) = \bigcup_{i \in \mathrm{dom}(w)} R_i = \bigcup_{i \in \mathrm{dom}(w)} Q_i \subseteq Q.$$

- The proof for $\zeta = \sqcap X.\zeta'$ with $X \in V_2$ is completely analogous to the proof of the previous case. The difference is that we consider ”for all $I \subseteq \mathrm{dom}(w)$” instead of ”for all $i \in \mathrm{dom}(w)$”.

Finally, we construct $\varPhi(\zeta)$ from $\varPhi_Y(\zeta)$ by labelling all positions not in Y by $\#$:

$$\varPhi(\zeta) = \exists Y.(\varPhi_Y(\zeta) \wedge \forall x.(Y(x) \vee \bigvee_{a \in \varSigma} P_{(a,\#)}(x))).$$

Assume that $\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \eta$. Let $R = \mathrm{dom}(\eta)$ and consider $\sigma' = \sigma[Y/R]$. Then $\sigma'(Y) \subseteq \mathrm{dom}(\eta)$ and $\langle\!\langle \zeta \rangle\!\rangle(w_\sigma) = \eta|_{\sigma(Y)}$. Then by (1) we have $(\mathrm{code}(w,\eta))_{\sigma'} \models$

$\Phi_Y(\zeta)$. Moreover, for all $i \in \mathrm{dom}(w) \setminus \sigma'(Y)$, the value $\eta(i)$ is undefined and hence $(\mathrm{code}(w, \eta))_{\sigma'} \models \forall x.(Y(x) \vee \bigvee_{a \in \Sigma} P_{(a,\#)}(x))$ which implies $(\mathrm{code}(w, \eta))_{\sigma} \models \Phi(\zeta)$. □

Now we continue the proof of Lemma 5.2. We apply Lemma 5.3 to the case $\zeta = \varphi$. Then, $\Phi(\varphi)$ is a sentence and $\mathcal{L}(\Phi(\varphi)) = \{\mathrm{code}(w, \eta) \mid \langle\!\langle \varphi \rangle\!\rangle(w) = \eta \neq \bot\}$. Note that $\mathcal{L}(\Phi(\varphi))$ is h-unambiguous, since for every $w \in \Sigma^\omega$ there exists at most one $u \in \mathcal{L}(\Phi(\varphi))$ with $h(u) = w$. If we let $\beta = \Phi(\varphi)$, then we obtain the desired decomposition $[\![\varphi]\!] = h((\mathrm{val} \circ g) \cap \mathcal{L}(\beta))$. Indeed, let $w \in \Sigma^\omega$. Then we distinguish between the following two cases:

- $\langle\!\langle \varphi \rangle\!\rangle(w) = \bot$. Then $[\![\varphi]\!](w) = \mathbb{0}$. On the other side, there exists no η with $\mathrm{code}(w, \eta) \in \mathcal{L}(\beta)$ and hence no $u \in \mathcal{L}(\beta)$ with $h(u) = w$. Then $h((\mathrm{val} \circ g) \cap \mathcal{L}(\beta))(w) = \mathbb{0} = [\![\varphi]\!](w)$.
- $\langle\!\langle \varphi \rangle\!\rangle(w) \in M^{\uparrow}$. Then, since the mapping g assigns the default weight $\mathbb{1}$ to the undefined positions of $\langle\!\langle \varphi \rangle\!\rangle(w) \in M^{\uparrow}$ and $\mathcal{L}(\beta)$ is h-unambiguous, we also have $h((\mathrm{val} \circ g) \cap \mathcal{L}(\beta))(w) = [\![\varphi]\!](w)$.

This finishes the proof of Lemma 5.2. Hence **WAL**-definability implies unambiguous recognizability.

5.2 Unambiguous Case: Recognizability Implies Definability

Now we show the converse part of Theorem 5.1(a), i.e., we show that unambiguous recognizability implies **WAL**-definability.

Lemma 5.4. *Let $\mathcal{A}$ be an unambiguous WBA over Σ and $\mathbb{V}$. Then, the quantitative ω-language $[\![\mathcal{A}]\!]$ if* **WAL***-definable over $\mathbb{V}$.*

Proof. Let $\mathcal{A} = (Q, I, T, F, \mathrm{wt})$ be an unambiguous WBA over Σ and $\mathbb{V}$. First, using the standard approach, we describe runs of $\mathcal{A}$ by means of MSO-formulas. For this, we fix an enumeration $(t_i)_{1 \leq i \leq m}$ of T and associate with every transition t_i a second-order variable X_i which keeps track of positions where t is taken. Then, a run of $\mathcal{A}$ can be described using a formula $\beta \in \mathbf{MSO}(\Sigma)$ with $\textsc{Free}(\beta) = \{X_1, ..., X_m\}$ which demands that values of the variables $X_1, ..., X_m$ form a partition of the domain of an input word, the transitions of a run are matching, the labels of transitions of a run are compatible with an input word, a run starts in I and visits some state in F infinitely often. Let $\mathbb{1} \in M$ be an arbitrary default weight. Consider the $\mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$-sentence

$$\varphi = W(\exists X_1 ... \exists X_m.\beta) \sqcap \big(\sqcap X_1 ... \sqcap X_m.[W(\beta) \Rightarrow \sqcap x. \textstyle\bigsqcap_{i=1}^{m} X_i(x) \Rightarrow (x \mapsto \mathrm{wt}(t_i))]\big).$$

Now we show that $[\![\varphi]\!] = [\![\mathcal{A}]\!]$. Let $w \in \Sigma^\omega$. We distinguish between the following two cases.

- $\mathrm{Run}_{\mathcal{A}}(w) = \emptyset$. Then $[\![\mathcal{A}]\!](w) = \mathbb{0}$. On the other side, $w \not\models \exists X_1 ... \exists X_m.\beta$ which implies $\langle\!\langle W(\exists X_1 ... \exists X_m.\beta) \rangle\!\rangle(w) = \bot$. Then $\langle\!\langle \varphi \rangle\!\rangle(w) = \bot$ and hence $[\![\varphi]\!](w) = \mathbb{0} = [\![\mathcal{A}]\!](w)$.

- $\mathrm{Run}_{\mathcal{A}}(w) \neq \emptyset$. Since $\mathcal{A}$ is unambiguous, we have $\mathrm{Run}_{\mathcal{A}}(w) = \{\rho\}$. Let $\rho = (\tau_i)_{i\in\mathbb{N}}$ and σ be a fixed w-assignment. Then, there exists exactly one tuple $(I_1, ..., I_m) \in (2^{\mathrm{dom}(w)})^m$ such that $w_{\sigma[X_1/I_1]...[X_m/I_m]} \models \beta$. Then $\langle\!\langle W(\exists X_1 ... \exists X_m.\beta)\rangle\!\rangle(w) = \top$. Moreover,

  $$\langle\!\langle W(\beta) \Rightarrow \sqcap x. \bigsqcap_{i=1}^{m} X_i(x) \Rightarrow (x \mapsto \mathrm{wt}(t_i))\rangle\!\rangle(w_{\sigma[X_1/I_1]...[X_m/I_m]}) = (\mathrm{wt}(\tau_i))_{i\in\mathbb{N}}$$

  and, for all $(J_1, ..., J_m) \in (2^{\mathrm{dom}(w)})^m$ with $(J_1, ..., J_m) \neq (I_1, ..., I_m)$, we have

  $$\langle\!\langle W(\beta) \Rightarrow \sqcap x. \bigsqcap_{i=1}^{m} X_i(x) \Rightarrow (x \mapsto \mathrm{wt}(t_i))\rangle\!\rangle(w_{\sigma[X_1/J_1]...[X_m/J_m]}) = \top.$$

  Then $\langle\!\langle \varphi \rangle\!\rangle(w) = (\mathrm{wt}(\tau_i))_{i\in\mathbb{N}}$ and hence $[\![\varphi]\!](w) = \mathrm{wt}_{\mathcal{A}}(\rho) = [\![\mathcal{A}]\!](w)$.

Hence $[\![\mathcal{A}]\!]$ is $\mathbf{WAL}$-definable over $\mathbb{V}$. □

5.3 Nondeterministic Case: Definability Implies Recognizability

Now we turn to the proof of Theorem 5.1(b). First we show that $\mathbf{eWAL}$-definability implies nondeterministic recognizability.

Lemma 5.5. *Let $\mathbb{1} \in M$ be a default weight and $\psi \in \mathbf{eWAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$. Then the quantitative ω-language $[\![\varphi]\!]$ is recognizable over $\mathbb{V}$.*

Proof. The idea of our proof is similar to the unambiguous case, i.e., via a decomposition of the $\mathbf{eWAL}$-sentence ψ. We show that there exist an extended alphabet Γ, renamings $h : \Gamma \to \Sigma$ and $g : \Gamma \to M$, and a sentence $\beta \in \mathbf{MSO}(\Gamma)$ such that $[\![\varphi]\!] = h((\mathrm{val} \circ g) \cap \mathcal{L}(\beta))$. Note that, as opposed to the unambiguous case, the ω-language $\mathcal{L}(\beta)$ is not necessarily h-unambiguous.

We may assume that $\psi = \sqcup x_1 ... \sqcup x_k . \sqcup X_1 ... \sqcup X_l . \varphi$ where $\varphi \in \mathbf{WAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$ and $x_1, ..., x_k$, X_1, ..., X_l are pairwise distinct variables.

As opposed to the unambiguous case, the extended alphabet Γ must also keep track of the values of the variables $x_1, ..., x_k, X_1, ..., X_l$. Let $\mathcal{V} = \{x_1, ..., x_k, X_1, ..., X_l\}$ and Δ_φ be defined as in the unambiguous case. Then we let $\Gamma = \Sigma \times \Delta_\varphi \times 2^{\mathcal{V}}$ and define h, g for all $u = (a, b, S) \in \Gamma$ with $a \in \Sigma$, $b \in \Delta_\varphi$ and $S \subseteq \mathcal{V}$ by $h(u) = a$ and $g(u) = b$ if $b \in M$ and $g(u) = \mathbb{1}$ otherwise. Finally we construct the MSO-sentence β over Γ. The construction of β will be based on Lemma 5.3. Let $\Phi(\varphi) \in \mathbf{MSO}(\Sigma \times \Delta_\varphi)$ be the formula constructed in Lemma 5.3 for $\zeta = \varphi$. Let $\overline{\Phi(\varphi)} \in \mathbf{MSO}(\Gamma)$ be the formula obtained from $\Phi(\varphi)$ by replacing every predicate $P_{(a,b)}(x)$ occurring in $\Phi(\varphi)$ by the formula $\bigvee(P_{(a,b,U)}(x) \mid U \subseteq \mathcal{V})$. Using the standard Büchi encoding technique we construct the formula $\phi \in \mathbf{MSO}(\Gamma)$ which encodes the values of $\mathcal{V}$-variables in the $2^{\mathcal{V}}$-component of an ω-word over Γ. We let $\phi = \forall y.(\phi_1 \wedge \phi_2)$ where

$$\phi_1 = \bigwedge_{x \in \mathcal{V} \cap V_1} ([R_{x,1}(y) \wedge (y = x)] \vee [R_{x,0}(y) \wedge (y \neq x)]),$$

$$\phi_2 = \bigwedge_{X \in \mathcal{V} \cap V_2} ([R_{X,1}(y) \wedge X(y)] \vee [R_{X,0}(y) \wedge \neg X(y)])$$

and, for $\mathcal{X} \in V$ and $i \in \{0, 1\}$, $R_{\mathcal{X},i}(y)$ denotes the formula

$$\bigvee(P_{(a,b,S)}(y) \mid a \in \Sigma, b \in \Delta_\varphi \text{ and } S \subseteq \mathcal{V} \text{ with } \mathcal{X} \lhd_i S)$$

where $\lhd_1 = \in$ and $\lhd_0 = \notin$.

Then we let $\beta = \exists x_1 ... \exists x_k . \exists X_1 ... \exists X_l .(\phi \wedge \overline{\Phi(\varphi)})$. It remains to show that $[\![\psi]\!] = h((\mathrm{val} \circ g) \cap \mathcal{L}(\beta))$.

Let $w = (a_i)_{i \in \mathbb{N}} \in \Sigma^\omega$. For any $u = (b_i)_{i \in \mathbb{N}} \in \Delta_\varphi^\omega$ we will abuse notation and write (w, u) for $((a_i, b_i))_{i \in \mathbb{N}}$. For $w \in \Sigma^\omega$, let $\mathcal{V}_w$ denote the set of all mappings $\mathcal{J} : \mathcal{V} \to \mathrm{dom}(w) \cup 2^{\mathrm{dom}(w)}$ such that $\mathcal{J}(\mathcal{V} \cap V_1) \subseteq \mathrm{dom}(w)$ and $\mathcal{J}(\mathcal{V} \cap V_2) \subseteq 2^{\mathrm{dom}(w)}$. For a w-assignment σ and $\mathcal{J} \in \mathcal{V}_w$, let $\sigma' := \sigma[\mathcal{V}/\mathcal{J}]$ denote the w-assignment such that $\sigma'|_{\mathcal{V}} = \mathcal{J}$ and $\sigma'_{V \setminus \mathcal{V}} = \sigma|_{V \setminus \mathcal{V}}$. Then

$$\begin{aligned} h((\mathrm{val} \circ g) \cap \mathcal{L}(\beta))(w) &= \sum(\mathrm{val}(g(u)) \mid \mathcal{J} \in \mathcal{V}_w \text{ and } (w, u)_{\sigma[\mathcal{V}/\mathcal{J}]} \models \Phi(\varphi)) \\ &\overset{(!)}{=} \sum_{\mathcal{J} \in \mathcal{V}_w} [\![\varphi]\!](w_{\sigma[\mathcal{V}/\mathcal{J}]}) \\ &= [\![\psi]\!](w). \end{aligned}$$

Then, the quantitative ω-language $[\![\psi]\!]$ is recognizable over $\mathbb{V}$ by Theorem 3.1 (b) and the classical Büchi theorem (which states that $\mathcal{L}(\beta)$ is a recognizable ω-language). □

6 Nondeterministic Case: Recognizability implies Definability

Now we show the converse direction of Theorem 5.1(b), i.e., that recognizability implies $\mathbf{eWAL}$-definability.

Lemma 6.1. *Let $\mathcal{A}$ be a WBA over Σ and $\mathbb{V}$. Then the quantitative ω-language $[\![\mathcal{A}]\!]$ is $\mathbf{eWAL}$-definable over $\mathbb{V}$.*

Proof. Our proof is a slight modification of our proof of Lemma 5.4. Let $\mathcal{A} = (Q, I, T, F, \mathrm{wt})$ be a nondeterministic WBA. Adopting the notations from the proof of Lemma 5.1, we construct the $\mathbf{eWAL}(\Sigma, \mathbb{V}_{\mathbb{1}})$-sentence

$$\psi = \sqcup X_1 ... \sqcup X_m . \big(W(\beta) \Rightarrow \sqcap x . \textstyle\bigsqcap_{i=1}^m X_i(x) \Rightarrow (x \mapsto \mathrm{wt}(t_i))\big).$$

(where $\mathbb{1}$ is irrelevant for the definition of ψ). Now we show that $[\![\varphi]\!] = [\![\mathcal{A}]\!]$. Let $w \in \Sigma^\omega$. Then, using the correspondence between the values of $X_1, ..., X_m$ and the runs in $\mathrm{Run}_\mathcal{A}(w)$, we obtain

$$[\![\psi]\!](w) = \sum_{\rho = (\tau_i)_{i \in \mathbb{N}} \in \mathrm{Run}_\mathcal{A}(w)} \mathrm{val}(\mathrm{wt}(\tau_i)) = \sum_{\rho \in \mathrm{Run}_\mathcal{A}(w)} \mathrm{wt}_\mathcal{A}(\rho) = [\![\mathcal{A}]\!](w).$$

This shows that $[\![\psi]\!]$ is $\mathbf{eWAL}$-definable over $\mathbb{V}$. □

7 Discussion

In this paper we introduced a weight assignment logic which is a simple and intuitive logical formalism for reasoning about quantitative ω-languages. Moreover, it works with arbitrary valuation functions whereas in weighted logics of [12], [14] some additional restrictions on valuation functions were added. We showed that WAL is expressively equivalent to unambiguous weighted Büchi automata. We also considered an extension of WAL which is equivalent to nondeterministic Büchi automata. Our expressiveness equivalence results can be helpful to obtain decidability properties for our new logics. The future research should investigate decidability properties of nondeterministic and unambiguous weighted Büchi automata with the practically relevant objectives. Although the weighted ω-automata models [7] do not have a Büchi acceptance condition, it seems likely that their decidability results about the threshold problems hold for Büchi acceptance condition as well. It could be also interesting to study our weight assignment technique in the context of temporal logic like LTL.

Our results obtained for ω-words can be easily adopted to the structures like finite words and trees. We have also extended the results of this paper to the timed setting and obtained a logical characterization of *multi-weighted timed automata* (cf., e.g., [5], [21]). For the proof of this result we applied a Nivat decomposition theorem for weighted timed automata [14]. Due to space constraints we cannot present this result here.

References

[1] Andersson, D.: Improved combinatorial algorithms for discounted payoff games. Master's thesis, Uppsala University, Department of Information Technology (2006).

[2] Bauer, S., Juhl, L., Larsen, K., Legay, A., Srba, J.: A logic for accumulated-weight reasoning on multiweighted modal automata. In: TASE 2012, pp. 77–84. IEEE (2012).

[3] Bloem, R., Greimel, K., Henzinger, T.A., Jobstmann, B.: Synthesizing robust systems. In: FMCAD 2009, pp. 85–92. IEEE (2009).

[4] Bouyer, P.: A logical characterization of data languages. Inf. Process. Lett. 84(2), 75–85 (2002).

[5] Bouyer, P., Brinksma, E., Larsen, K.G.: Optimal infinite scheduling for multi-priced timed automata. Formal Methods in System Design 32, 3–23 (2008).

[6] Büchi, J.R.: Weak second-order arithmetic and finite automata. Z. Math. Logik und Grundl. Math. 6, 66–92 (1960).

[7] Chatterjee, K., Doyen, L., Henzinger, T.A.: Quantitative languages. In: CSL 2008. LNCS, vol. 5213, pp. 385–400. Springer (2008).

[8] Carton, O., Michel, M.: Unambiguous Büchi automata. In: LATIN 2000. LNCS, vol. 1776, pp. 407–416. Springer (2000).

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

[10] Droste, M., Kuich, W., Vogler, H. (eds.): Handbook of Weighted Automata. EATCS Monographs on Theoretical Computer Science. Springer (2009).

[11] Droste, M., Kuske, D.: Weighted automata. In: Pin, J.-E. (ed.) Handbook: "Automata: from Mathematics to Applications", European Math- ematical Society, to appear.

[12] Droste, M., Meinecke, I.: Weighted automata and weighted MSO logics for average and long-time behaviors. Inf. Comput. 220-221, 44–59 (2012).

[13] Droste, M., Perevoshchikov, V.: Multi-weighted automata and MSO logic. In: CSR 2013. LNCS, vol. 7913, pp. 418–430. Springer (2013).

[14] Droste, M., Perevoshchikov, V.: A Nivat theorem for weighted timed automata and relative distance logic. In: ICALP 2014. LNCS, vol. 8573, pp. 171–182. Springer (2009).

[15] Droste, M., Rahonis, G.: Weighted automata and weighted logics on infinite words. In: DLT 2006. LNCS, vol. 4036, pp. 49–58. Springer (2006).

[16] Droste, M., Vogler, H.: Weighted automata and multi-valued logics over arbitrary bounded lattices. Theoret. Comp. Science 418, 14–36 (2012).

[17] Fahrenberg, U., Juhl, L., Larsen, K.G., Srba, J.: Energy games in multiweighted automata. In: ICTAC 2011. LNCS, vol. 6916, pp. 95–115. Springer (2011).

[18] Filiot, E., Gentilini, R., Raskin, J.-F.: Quantitative languages defined by functional automata. In: CONCUR 2012. LNCS, vol. 7454, pp. 132–146. Springer (2012).

[19] Hashiguchi, K., Ishiguro, K., Jimbo, S.: Decidability of the equivalence problem for finitely ambiguous finance automata. Int. Journal of Algebra and Computation 12(3), 445–461 (2002).

[20] Krob, D.: The equality problem for rational series with multiplicities in the tropical semiring is undecidable. International Journal of Algebra and Computation 4(3), 405–425 (1994).

[21] Larsen, K.G., Rasmussen, J.I.: Optimal conditional reachability for multi-priced timed automata. In: FoSSaCS 2005. LNCS, vol. 3441, pp. 234–249. Springer (2005).

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

[23] Perevoshchikov, V.: Weight assignment logic. In: DLT 2015. LNCS, vol. 9168, pp. 413–425. Springer (2015).

[24] Stüber, T., Vogler, H., Fülöp, Z.: Decomposition of weighted multioperator tree automata. Int. J. Foundations of Computer Sci. 20(2), 221–245 (2009).

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