# Weighted Automata and Logics on Infinite Graphs

Stefan Dück*

Institute of Computer Science, Leipzig University, D-04109 Leipzig, Germany
dueck@informatik.uni-leipzig.de

**Abstract.** We show a Büchi-like connection between graph automata and logics for infinite graphs. Using valuation monoids, a very general weight structure able to model computations like average or discounting, we extend this result to the quantitative setting. This gives us the first general results connecting automata and logics over infinite graphs in the qualitative and the quantitative setting.

**Keywords:** quantitative automata, infinite graphs, graphs, quantitative logic, valuation monoids

## 1 Introduction

The coincidence between the languages recognizable by a finite state machine and the languages definable in monadic second order theory is one of the most fruitful results in theoretical computer science. Since Büchi, Elgot, and Trakhtenbrot [6, 18, 36] established this fundamental result, it has not only been the corner stone of multiple applications, like verification of finite-state programs, but also lead to multiple extensions covering finite and infinite trees [28, 31], traces [32], pictures [22], (infinite) nested words [1], and texts [24]. A general result for finite graphs was given by Thomas [33].

It has remained an open question whether it is possible to get such a result for infinite graphs. In particular, this question is unanswered in the case of infinite pictures. The main contributions of this paper are the following:

- We show a Büchi-like equivalence between infinite graph acceptors and an EMSO-logic for infinite graphs.
- We establish a valuation-weighted automata model over graphs, which generalizes semiring-weighted automata and comprises previous automata models over special classes of graphs.
- Using methods of weighted logics, we extend our Büchi-like result from the qualitative to the quantitative setting, i.e., we show the equivalence of weighted infinite graph automata to a restricted weighted MSO-logic.

* supported by Deutsche Forschungsgemeinschaft (DFG) Graduiertenkolleg 1763 (QuantLA)

Formally introduced by Schützenberger [30], the study of quantitative questions (How often does an event arise?; What is the cost of this solution?; etc.) is another flourishing theory (see e.g. [17, 2] and the recent handbook [13]). Quantitative automata modeling the long-time average or discounted behavior of systems were investigated, e.g., by Chatterjee, Doyen, and Henzinger [7].

Recently, Bollig and Kuske [4] considered a logic $\mathrm{FO}^\infty$ featuring a first-order quantifier expressing that there are infinitely many elements satisfying a formula. In a different context than ours (for Muller message-passing automata), they were able to relate an extended Ehrenfeucht-Fraïssé game and $k$-equivalence of two formulas of $\mathrm{FO}^\infty$, thus developing a Hanf-like theorem [23] for this logic. We show how this result can be applied to infinite graphs to connect $\mathrm{EMSO}^\infty$ and infinite graph automata, yielding our first main result.

Using weighted MSO-logic [11], its extension to graphs [10], and valuation monoids [14], we generalize our graph automata model and our Büchi-like result to a quantitative setting. Here, one crucial part is the closure under the (restricted) weighted universal quantification (the *valuation-quantification*). An essential part of proving this closure is utilizing [4] to show that $\mathrm{FO}^\infty$ corresponds to one-state infinite graph acceptors.

To enhance readability, we first develop our weighted results in the finite case. Note that using valuation monoids, this model and the results are also new for finite graphs and enable us to consider examples using average or discounting in this general setting, as well as classical (possibly non-commutative) semirings. Furthermore, our approach is designed in an adaptable way, thereby facilitating the later extensions to infinite graphs.

## 2 Graphs and Graph Acceptors

In this section, we introduce the basic concepts around graphs and graph acceptors. Following [10, 34], we define a *(directed) unpointed graph* as a relational structure $G = (V, (P_a)_{a\in A}, (E_b)_{b\in B})$ over two finite alphabets $A$ and $B$, where $V$ is the set of *vertices*, the sets $P_a$, $a \in A$, form a partition of $V$, and the sets $E_b$, $b \in B$, are pairwise disjoint irreflexive binary relations on $V$, called *edges*. We denote by $E = \bigcup_{b\in B} E_b$ the set of all edges. Then the elements of $A$ are the vertex labels, and the elements of $B$ are the edge labels. A graph is *bounded by $t$* if every vertex has an (in- plus out-) degree less than or equal to $t$.

We call a class of graphs *pointed* if every graph $G$ of this class has a distinguished vertex. Formally, this assumption can be defined by adding a unary relation *root* to $G$ with $\mathrm{root} = \{u\}$.

We consider subgraphs of a pointed graph $(G, u)$ around a vertex $v$ as follows. We call $\tau = (H, v, w)$ a *tile* if $(H, v)$ is a pointed graph and either $w$ is an additionally distinguished vertex of $H$ or $w = \textit{empty}$. Let $r \geq 0$. We denote by $\mathrm{dist}(x, y) \leq r$ that there exists a path $(x = x_0, x_1, ..., x_j = y)$ with $j \leq r$ and $(x_i, x_{i+1}) \in E$ or $(x_{i+1}, x_i) \in E$ for all $i < j$. We call $(H, v, u)$ an *$r$-tile* if for every vertex $x$ of $H$, it holds that $\mathrm{dist}(x, v) \leq r$. We denote by $\mathrm{sph}^r((G, u), v)$ the unique $r$-tile $(H, v, w)$ consisting of all vertices $x$ of $G$ with $\mathrm{dist}(x, v) \leq r$

together with their edges and $w = u$ if $\mathrm{dist}(u, v) \leq r$ and $w = \mathit{empty}$, otherwise. We say $v$ is the *center* of $\tau = (H, v, w)$, resp. of $\tau = (H, v) = (H, v, \mathit{empty})$.

In this work, we assume all *graphs* to be pointed. We may omit the explicit root $u$ of a graph and the radius $r$ of a tile if the context is clear. Moreover, our results not explicitly utilizing the root also hold for unpointed graphs $G$.

We denote by $\mathrm{Lab}_G(v)$ the label of the vertex $v$ of the graph $G$. We denote by $\mathrm{DG}_t(A, B)$ the class of all finite, directed, and pointed graphs over $A$ and $B$, bounded by $t$. We denote by $\mathrm{DG}_t^\infty(A, B)$ the class of all infinite, directed, and pointed graphs over $A$ and $B$, bounded by $t$. Note that $r$-tiles of finite or infinite graphs are finite structures, and there exist only finitely many non-isomorphic $r$-tiles since the degree of every considered graph is bounded.

**Definition 1 ([33, 34]).** *A* graph acceptor (GA) $\mathcal{A}$ *over* $\mathrm{DG}_t(A, B)$ *is defined as a quadruple* $\mathcal{A} = (Q, \Delta, \mathrm{Occ}, r)$ *where*

- $Q$ *is a finite set of states,*
- $r \in \mathbb{N}$ *is the* tile-radius*,*
- $\Delta$ *is a finite set of pairwise non-isomorphic $r$-tiles over $A \times Q$ and $B$,*
- Occ*, the* occurrence constraint*, is a boolean combination of formulas "*$\mathrm{occ}(\tau) \geq n$*", where $n \in \mathbb{N}$ and $\tau \in \Delta$.*

Note that Thomas (cf. [33, 34]) uses non-pointed graphs. Here, the pointing can be seen as optional additional information to distinguish tiles from each other.

Given a finite graph $G = (G, u)$ of $\mathrm{DG}_t(A, B)$ and a mapping $\rho : V \to Q$, we consider the graph $G_\rho = (G_\rho, u) \in \mathrm{DG}_t(A \times Q, B)$, which consists of the same vertexes and edges as $G$ and is additionally labeled with $\rho(v)$ at every vertex $v$.

We call $\rho$ a *run (or tiling) of $\mathcal{A}$ on $G$* if for every $v \in V$, $\mathrm{sph}^r(G_\rho, v)$ is isomorphic to a tile in $\Delta$. We say $G_\rho$ *satisfies* $\mathrm{occ}(\tau) \geq n$ if there exist at least $n$ distinct vertices $v \in V$ such that $\mathrm{sph}^r(G_\rho, v)$ is isomorphic to $\tau$. The semantics of "$G_\rho$ *satisfies* Occ" are then defined in the usual way.

We call a run $\rho$ *accepting* if $G_\rho$ satisfies Occ. We say that $\mathcal{A}$ *accepts* the graph $G \in \mathrm{DG}_t(A, B)$ if there exists an accepting run $\rho$ of $\mathcal{A}$ on $G$. We define $L(\mathcal{A}) = \{G \in \mathrm{DG}_t(A, B) \mid \mathcal{A} \text{ accepts } G\}$, the *language accepted by* $\mathcal{A}$. We call a language $L \subseteq \mathrm{DG}_t(A, B)$ *recognizable* if $L = L(\mathcal{A})$ for some GA $\mathcal{A}$.

Next, we introduce the logic $\mathrm{MSO}(\mathrm{DG}_t(A, B))$, short MSO, cf. [34]. We denote by $x, y, ...$ first-order variables ranging over vertices and by $X, Y, ...$ second order variables ranging over sets of vertices. The formulas of MSO are defined inductively by

$$\varphi ::= P_a(x) \mid E_b(x, y) \mid \mathrm{root}(x) \mid x = y \mid x \in X \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x.\varphi \mid \exists X.\varphi$$

where $a \in A$ and $b \in B$. An *FO-formula* is a formula of MSO without set quantifications, i.e., without using $\exists X$. An *EMSO-formula* is a formula of the form $\exists X_1, ..., \exists X_k.\varphi$ where $\varphi$ is an FO-formula.

The satisfaction relation $\models$ for graphs and MSO-sentences is defined in the natural way. Then for a sentence $\varphi \in \mathrm{MSO}$, we define the *language of* $\varphi$ as $L(\varphi) = \{G \in \mathrm{DG}_t(A, B) \mid G \models \varphi\}$. We call a language $L \subseteq \mathrm{DG}_t(A, B)$ *MSO-definable* (resp. *FO-definable)* if $L = L(\varphi)$ for some MSO-sentence (resp. FO-sentence) $\varphi$.

**Theorem 2 ([34]).** *Let $L \subseteq \mathrm{DG}_t(A, B)$ be a set of graphs. Then:*

1. *$L$ is recognizable by a one-state GA iff $L$ is definable by an FO-sentence.*
2. *$L$ is recognizable iff $L$ is definable by an EMSO-sentence.*

## 3 Infinite Graph Acceptors

In the following, we extend Theorem 2 to the infinite setting, thus showing a Büchi-like result for infinite graphs. We introduce infinite graph acceptors with an extended acceptance condition and an EMSO$^\infty$ logic featuring a first-order quantifier $\exists^\infty x.\varphi$ to express that there exist infinitely many vertices fulfilling $\varphi$.

Using the occurrence constraint as acceptance condition, the introduced graph acceptor for finite graphs could also be interpreted as a model for infinite graphs. However, every occurrence constraint only checks for occurrences up to a certain threshold, i.e., it cannot express that a tile occurs infinitely many often. This motivates the following definition.

**Definition 3.** *An* infinite graph acceptor (GA$^\infty$) $\mathcal{A}$ *over* $\mathrm{DG}_t^\infty(A, B)$ *is defined as a quadruple* $\mathcal{A} = (Q, \Delta, \mathrm{Occ}, r)$ *where*

- *$Q$, $\Delta$, and $r$ are defined as before, and*
- Occ*, the* extended occurrence constraint*, is a boolean combination of formulas "*$\mathrm{occ}(\tau) \geq n$*" and "*$\mathrm{occ}(\tau) = \infty$*", where $n \in \mathbb{N}$ and $\tau \in \Delta$.*

The notions of an *accepting run* $\rho$ of $\mathcal{A}$ on $G \in \mathrm{DG}_t^\infty(A, B)$ and a *recognizable language* $L = L(\mathcal{A}) \subseteq \mathrm{DG}_t^\infty(A, B)$ are defined as before.

Next, following [4], we introduce the logic MSO$^\infty$($\mathrm{DG}_t^\infty(A, B)$), short MSO$^\infty$, by the following grammar

$$\varphi ::= P_a(x) \mid E_b(x, y) \mid \mathrm{root}(x) \mid x = y \mid x \in X \mid \neg\varphi \mid \varphi \vee \varphi \mid \exists x.\varphi \mid \exists^\infty x.\varphi \mid \exists X.\varphi$$

We denote by FO$^\infty$, resp. EMSO$^\infty$, the usual first-order, resp. existential fragment. Defining an *assignment* $\sigma$ and an *update* $\sigma[x \to v]$ as usual, the satisfaction relation $\models$ is defined as before, together with $(G, \sigma) \models \exists^\infty x.\varphi$ iff $(G, \sigma[x \to v]) \models \varphi$ for infinitely many $v \in V$.

Using an extended Ehrenfeucht-Fraïssé game, Bollig and Kuske [4] succeeded in proving a Hanf-like result for these structures. It says that for a given $k \in \mathbb{N}$ and a fixed maximal degree, there exists a sufficiently large tile-radius $r$ and a threshold $h$ such that two graphs which cannot be distinguished by an extended occurrence constraint over $r$ and $h$ are also indistinguishable by any FO$^\infty$-formula up to quantifier depth $k$.

From this result, which was originally developed in a different context, namely Muller message-passing automata, we can deduce the following corollary.

**Corollary 4.** *Let $\varphi$ be an FO$^\infty$-sentence. Then there exists an extended occurrence constraint* Occ *such that $G \models \varphi$ iff $G \models \mathrm{Occ}$ for all $G \in \mathrm{DG}_t^\infty(A, B)$.*

This result provides us with the means to prove our first main theorem.

**Theorem 5.** *Let $L \subseteq \mathrm{DG}_t^\infty(A, B)$ be a set of infinite graphs. Then:*

1. *$L$ is recognizable by a one-state GA$^\infty$ iff $L$ is definable by an* FO$^\infty$*-sentence.*
2. *$L$ is recognizable iff $L$ is definable by an* EMSO$^\infty$*-sentence.*

## 4 Weighted Graph Automata

In this section, we introduce and investigate a quantitative version of graph acceptors for finite graphs. We follow the approach of [10], but use more general structures than semirings, the *(graph-) valuation monoid* (cf. [14] for valuation monoids over words), which are able to model aspects like average, discounting, and other long-time behaviors of automata.

By abuse of notation, we also consider finite graphs $\mathrm{DG}_t(M, B)$ over an infinite set $M$. Note that we use this notation only in our weight assignments of the weighted automaton and never as part of the input or within a tile.

**Definition 6.** *A* (graph-) valuation monoid $\mathbb{D} = (D, +, \mathrm{Val}, 0)$ *consist of a commutative monoid* $(D, +, 0)$ *together with an absorptive valuation function* $\mathrm{Val} : \mathrm{DG}_t(D, B) \to D$*, i.e.,* $\mathrm{Val}(G) = 0$ *if at least one vertex of* $G$ *is labeled* $0$.

*In the following,* $\mathbb{D}$ *will always refer to a valuation monoid* [1] .

Note that we do not enforce distributivity or another form of compatibility between $+$ and Val. The choice of valuation monoids is a natural one when you want to consider strictly more general structures than semirings and incorporate examples like average or discounting, as follows. In the context of trees, another closely related structure are multi-operator monoids (see e.g. [20]).

*Example 7.* Let $\mathrm{dia}(G)$ be the diameter of $G = (G, u) \in \mathrm{DG}_t(A, B)$. We define $\mathrm{avg}(G) = \dfrac{1}{|V|} \displaystyle\sum_{v \in V} \mathrm{Lab}_G(v)$ and $\mathrm{disc}_\lambda(G, u) = \displaystyle\sum_{r=0,\ldots,\mathrm{dia}(G)} \sum_{\mathrm{dist}(v,u)=r} \lambda^r \, \mathrm{Lab}_G(v)$. Then $\mathbb{D}_1 = (\mathbb{R} \cup \{-\infty\}, \sup, \mathrm{avg}, -\infty)$ and $\mathbb{D}_2 = (\mathbb{R} \cup \{-\infty\}, \sup, \mathrm{disc}_\lambda, -\infty)$ are two valuation monoids. Note that $\mathbb{D}_1$ does not use the root of the graph; therefore, we can omit it. In contrast, $\mathbb{D}_2$ is only utilizable for pointed graphs.

**Definition 8.** *A* weighted graph automaton (wGA) *over* $\mathrm{DG}_t(A, B)$ *and* $\mathbb{D}$ *is a tuple* $\mathcal{A} = (Q, \Delta, \mathrm{wt}, \mathrm{Occ}, r)$ *where*

- $\mathcal{A}' = (Q, \Delta, \mathrm{Occ}, r)$ *is a graph acceptor over* $\mathrm{DG}_t(A, B)$,
- $\mathrm{wt} : \Delta \to D$ *is the weight function assigning to every tile of* $\Delta$ *a value of* $D$.

An *accepting run* $\rho : V \to Q$ *of* $\mathcal{A}$ *on* $G \in \mathrm{DG}_t(A, B)$ is defined as an accepting run of $\mathcal{A}'$ on $G$. As in the unweighted case, the pointing of $G = (G, u)$ is optional.

For an accepting run $\rho$, we consider the graph $G_\rho^D$, where every vertex is labeled with the weight of the tile the run $\rho$ defines around this vertex. More precisely, for a vertex $v$ of $G$, let $\tau_\rho(v)$ be the $r$-tile of $\Delta$ which is isomorphic to $\mathrm{sph}^r(G_\rho, v)$. Then $G_\rho^D$ is defined as the unique graph over $\mathrm{DG}_t(D, B)$ resulting from the graph $G$ where for all vertices $v$, $\mathrm{Lab}_{G_\rho^D}(v) = \mathrm{wt}(\tau_\rho(v))$.

We denote by $\mathrm{acc}_\mathcal{A}(G)$ the set of all accepting runs of $\mathcal{A}$ on $G$. The *behavior* $[\![\mathcal{A}]\!] : \mathrm{DG}_t(A, B) \to D$ of a wGA $\mathcal{A}$ is defined, for each $G \in \mathrm{DG}_t(A, B)$, as

$$[\![\mathcal{A}]\!](G) = \sum_{\rho \in \mathrm{acc}_\mathcal{A}(G)} \mathrm{Val}(G_\rho^D) \ .$$

[1] [14] enforced $\mathrm{Val}(d) = d$, which was later shown to be not required even in the word case, see e.g. [21]

We call any function $S : \mathrm{DG}_t(A, B) \to D$ a *series*. Then $S$ is *recognizable* if $S = [\![\mathcal{A}]\!]$ for some wGA $\mathcal{A}$. By the usual identification of languages with functions assuming values in $\{0, 1\}$, we see that graph acceptors are expressively equivalent to wGA over the Boolean semiring $\mathbb{B}$.

Following [14], we call $\mathbb{D}$ *regular* if all constant series of $D$ are recognizable, i.e., for every $d \in D$, there exists a wGA $\mathcal{A}_d$ with $[\![\mathcal{A}_d]\!](G) = d$ for every $G \in \mathrm{DG}_t(A, B)$.

*Example 9.* Let $A = \{a, b\}$ and $B = \{x\}$. For a given graph, we are interested in the value $\max_{a\in A} |V|_{a\&\text{no\_outgoing}}/|V|$ which is the maximal proportion of nodes which are labeled with the same symbol and have no outgoing edges. For instance, in a tree the numerator would refer to the number of leafs labeled with $a$. We can compute this value with the following wGA over $\mathbb{D}_1 = (\mathbb{R} \cup \{-\infty\}, \sup, \mathrm{avg}, -\infty)$.

Set $\mathcal{A} = (\{q_1, q_2\}, \Delta, \mathrm{wt}, \mathrm{Occ}, r)$, with $r = 1$, $\Delta = \{\tau \mid \tau \text{ is a 1-tile}\}$, and

$$\mathrm{Occ} = \bigwedge_{\{\tau \,\mid\, \mathrm{center}(\tau)\in\{(a,q_1),(b,q_2)\}\}} \mathrm{occ}(\tau) = 0 \vee \bigwedge_{\{\tau \,\mid\, \mathrm{center}(\tau)\in\{(a,q_2),(b,q_1)\}\}} \mathrm{occ}(\tau) = 0.$$

Furthermore, we define $\mathrm{wt}(\tau) = 1$ if the center $v$ of $\tau$ is labeled with $q_1$ and the center has no outgoing edges. Then $[\![\mathcal{A}]\!](G)$ is the desired proportion. □

*Example 10.* Let us assume our graph represents a social network. Now, we are interested into the affinity of a person to a certain characteristic (a hobby, a political tendency, an attribute, etc.) be it to use this information in a matching process or for personalized advertising. We assume that this affinity is closely related to the social environment of a person (e.g., I am more inclined to watch soccer if I play soccer myself, or I have friends who are interested into it).

We define a one-state wGA $\mathcal{A} = (\{q\}, \{\tau \mid \tau \text{ is a 1-tile}\}, \mathrm{wt}, \mathit{true}, 1)$ over $A = \{a, b\}$, $B = \{x\}$, and $\mathbb{D}_2 = (\mathbb{R} \cup \{-\infty\}, \sup, \mathrm{disc}_\lambda, -\infty)$, with $\mathrm{wt}(\tau) = \#_a(\tau)$, where $\#_a(\tau)$ is the number of vertices of $\tau$ labeled with $a$. Then depending on $\lambda$, $\mathcal{A}$ computes for a pointed graph $(G, u)$ the affinity of $u$ to the characteristic $a$.

Additionally introducing a nondeterministic choice for the center vertex $u$ into the wGA, modifying the valuation function accordingly, and taking the supremum of all resulting runs, we can construct a nondeterministic automaton computing the maximal affinity of all vertices of a non-pointed graph. □

In the following, we give some results using ideas of [10]. These statements utilize the following formula. Let $\tau^* = \{\tau_1, ..., \tau_m\}$ be a finite set of tiles. For $N \in \mathbb{N}$, we shall write

$$\Big(\sum_{\tau\in\tau^*} \mathrm{occ}(\tau)\Big) \geq N \quad \text{short for} \quad \bigvee_{\substack{\sum_{i=1}^m n_i = N\\ n_i\in\{0,\ldots,N\}}} \ \bigwedge_{i=1,\ldots,m} \mathrm{occ}(\tau_i) \geq n_i \ . \tag{1}$$

We can interpret $\tau^*$ as a set of tiles matching a certain pattern. Then this formula is true iff the occurrence number of all tiles matching this pattern is at least $N$.

Let $S : \mathrm{DG}_t(A, B) \to D$ be a series recognizable by a wGA $\mathcal{A}$ with tile-radius $s$. Then we can show that for all $r \geq s$, $S$ is recognizable by a wGA $\mathcal{B}$ with tile-radius $r$.

We extend the operation $+$ of our valuation monoid to series by means of point-wise definition, i.e., $(S+T)(G) = S(G) + T(G)$ for each $G \in \mathrm{DG}_t(A,B)$.

**Proposition 11.** *The class of recognizable series is closed under* $+$.

Let $S : \mathrm{DG}_t(A,B) \to D$ and $L \subseteq \mathrm{DG}_t(A,B)$. We define the *restriction* $S \cap L : \mathrm{DG}_t(A,B) \to D$ by letting $(S \cap L)(G) = S(G)$ if $G \in L$ and $(S \cap L)(G) = 0$, otherwise.

**Proposition 12.** *Let* $S : \mathrm{DG}_t(A,B) \to D$ *be a recognizable series and* $L \subseteq DG_t(A,B)$ *be recognizable by a one-state GA. Then* $S \cap L$ *is recognizable.*

*Proof (sketch).* We build the wGA recognizing $S \cap L$ as a product-automaton from the wGA $\mathcal{A}$ recognizing $S$ and the GA $\mathcal{B}$ recognizing $L$. The occurrence-constraint is combined by conjugating the projections to the constraints of $\mathcal{A}$ and $\mathcal{B}$ together with formula (1). Since $\mathcal{B}$ has exactly one state, we can control the number of runs of $\mathcal{C}$.

In the following, we show that recognizable series are closed under projection. Let $h : A' \to A$ be a mapping between two alphabets. Then $h$ naturally defines a relabeling of graphs from $\mathrm{DG}_t(A',B)$ into graphs from $\mathrm{DG}_t(A,B)$, also denoted by $h$. Let $S : \mathrm{DG}_t(A',B) \to D$ be a series. We define $h(S) : \mathrm{DG}_t(A,B) \to D$ by

$$h(S)(G) = \sum_{\substack{G' \in \mathrm{DG}_t(A',B) \\ h(G')=G}} S(G') \ . \tag{2}$$

**Proposition 13.** *Let* $S : \mathrm{DG}_t(A',B) \to D$ *be a recognizable series and* $h : A' \to A$. *Then* $h(S) : \mathrm{DG}_t(A,B) \to D$ *is recognizable.*

## 5 Weighted Logics for Graphs

In the following, we introduce a weighted MSO-Logic for finite graphs, following the approach of Droste and Gastin [11] for words. We also incorporate an idea of Bollig and Gastin [3] to consider unweighted MSO-formulas as explicit fragment of our logic. We utilize an idea of Gastin and Monmege [21] to consider formulas with an 'if..then..else'-operator $\beta?\varphi_1 : \varphi_2$ instead of a weighted conjunction $\varphi_1 \otimes \varphi_2$. This operator is able to model the essential step-functions (resp. the almost FO-boolean fragment) without the need to add a second operation to the valuation monoid (the product $\diamond$).

Note that our underlying structure may still provide a product (e.g. as in the case of semirings). In this case, it remains possible to enrich our logic with a second operation (previously denoted by $\otimes$), therefore getting a direct connection to previous works [11, 14, 10].

In both cases, we are able to prove a Büchi-like connection between our introduced weighted graph automata and the (restricted) weighted MSO logic. Since the second operation enforces additional technical restrictions, we omit the details for this case here.

**Definition 14.** *We define the weighted logic* $\mathrm{MSO}(\mathbb{D}, \mathrm{DG}_t(A,B))$*,* $\mathrm{MSO}(\mathbb{D})$*, as*

$$\beta ::= P_a(x) \mid E_b(x,y) \mid \mathrm{root}(x) \mid x = y \mid x \in X \mid \neg\beta \mid \beta \vee \beta \mid \exists x.\beta \mid \exists X.\beta$$
$$\varphi ::= d \mid \varphi \oplus \varphi \mid \beta ? \varphi : \varphi \mid \bigoplus_x \varphi \mid \bigoplus_X \varphi \mid \mathrm{Val}_x \varphi$$

*where* $d \in D$*;* $x, y$ *are first-order variables; and* $X$ *is a second order variable.*

Let $G \in \mathrm{DG}_t(A,B)$ and $\varphi \in \mathrm{MSO}(\mathbb{D})$. We follow classical approaches for logics and semantics. Let $\mathrm{free}(\varphi)$ be the set of all free variables in $\varphi$, and let $\mathcal{V}$ be a finite set of variables containing $\mathrm{free}(\varphi)$. A $(\mathcal{V}, G)$-*assignment* $\sigma$ is a function assigning to every first-order variable of $\mathcal{V}$ an element of $V$ and to every second order variable a subset of $V$. We define the *update* $\sigma[x \to v]$ as the $(\mathcal{V} \cup \{x\}, G)$-assignment mapping $x$ to $v$ and equaling $\sigma$ everywhere else. The assignment $\sigma[X \to I]$ is defined analogously.

We represent the graph $G$ together with the assignment $\sigma$ as a graph $(G, \sigma)$ over the vertex alphabet $A_{\mathcal{V}} = A \times \{0,1\}^{\mathcal{V}}$ where 1 denotes every position where $x$ resp. $X$ holds. A graph over $\mathcal{A}_{\mathcal{V}}$ is called *valid* if every first-order variable is assigned to exactly one position.

We define the *semantics* of $\varphi \in \mathrm{MSO}(\mathbb{D})$ as a function $[\![\varphi]\!]_{\mathcal{V}} : \mathrm{DG}_t(A_{\mathcal{V}}, B) \to D$ inductively for all valid $(G, \sigma) \in \mathrm{DG}_t(A_{\mathcal{V}}, B)$, as seen in Fig. 1. For not valid $(G, \sigma)$, we set $[\![\varphi]\!]_{\mathcal{V}}(G, \sigma) = 0$. We write $[\![\varphi]\!]$ for $[\![\varphi]\!]_{\mathrm{free}(\varphi)}$.

$$\begin{aligned}
[\![d]\!]_{\mathcal{V}}(G,\sigma) &= d \quad \text{for all } d \in D\\
[\![\varphi \oplus \psi]\!]_{\mathcal{V}}(G,\sigma) &= [\![\varphi]\!]_{\mathcal{V}}(G,\sigma) + [\![\psi]\!]_{\mathcal{V}}(G,\sigma)\\
[\![\beta ? \varphi : \psi]\!]_{\mathcal{V}}(G,\sigma) &= \begin{cases} [\![\varphi]\!]_{\mathcal{V}}(G,\sigma) \text{ , if } (G,\sigma) \models \beta \\ [\![\psi]\!]_{\mathcal{V}}(G,\sigma) \text{ , otherwise} \end{cases}\\
[\![\textstyle\bigoplus_x \varphi]\!]_{\mathcal{V}}(G,\sigma) &= \sum_{v \in V} [\![\varphi]\!]_{\mathcal{V} \cup \{x\}}(G, \sigma[x \to v])\\
[\![\textstyle\bigoplus_X \varphi]\!]_{\mathcal{V}}(G,\sigma) &= \sum_{I \subseteq V} [\![\varphi]\!]_{\mathcal{V} \cup \{X\}}(G, \sigma[X \to I])\\
[\![\mathrm{Val}_x \varphi]\!]_{\mathcal{V}}(G,\sigma) &= \mathrm{Val}((G,\sigma)_\varphi) \text{ where } (G,\sigma)_\varphi \text{ is the graph } (G,\sigma) \text{ where every}\\
&\quad\ \text{vertex } v \text{ is labeled with } [\![\varphi]\!]_{\mathcal{V} \cup \{x\}}(G, \sigma[x \to v])
\end{aligned}$$

**Fig. 1.** Semantics

Whether a graph is valid can be checked by an FO-formula, hence the language of all valid graphs over $A_{\mathcal{V}}$ is recognizable. For the Boolean semiring $\mathbb{B}$, the unweighted MSO is expressively equivalent to $\mathrm{MSO}(\mathbb{B})$.

The following lemma shows that for each finite set of variables containing $\mathrm{free}(\varphi)$, the semantics $[\![\varphi]\!]_{\mathcal{V}}$ are consistent with each other (cf. [11]).

**Lemma 15.** *Let* $\varphi \in \mathrm{MSO}(\mathbb{D})$ *and* $\mathcal{V}$ *be a finite set of variables with* $\mathcal{V} \supseteq \mathrm{free}(\varphi)$*. Then* $[\![\varphi]\!]_{\mathcal{V}}(G,\sigma) = [\![\varphi]\!](G, \sigma\restriction_{\mathrm{free}(\varphi)})$ *for each valid* $(G,\sigma) \in \mathrm{DG}_t(A_{\mathcal{V}}, B)$*. Furthermore, if* $[\![\varphi]\!]$ *is recognizable, then* $[\![\varphi]\!]_{\mathcal{V}}$ *is recognizable.*

Now, we show that recognizable series are closed under $\bigoplus_x$ and $\bigoplus_X$ quantification (in previous papers called the weighted existential quantification).

**Lemma 16.** *Let* $[\![\varphi]\!]$ *be recognizable. Then* $[\![\bigoplus_x \varphi]\!]$ *and* $[\![\bigoplus_X \varphi]\!]$ *are recognizable.*

The interesting case is the $\mathrm{Val}_x$-quantification (previously called the weighted universal quantification [11]). Similarly to [11], our unrestricted logic is strictly more powerful than our automata model. Therefore, inspired by [14] and [21], we introduce the following fragment.

We call a formula $\varphi \in \mathrm{MSO}(\mathbb{D})$ *almost FO-boolean* if $\varphi$ is built up inductively from the grammar, $\varphi ::= d \mid \beta?d : \varphi$ , where $d \in D$ and $\beta$ is an unweighted FO-formula.

This fragment is equivalent to all formulas $\varphi$ such that $[\![\varphi]\!]$ is an *FO-step function*, i.e., it takes only finitely many values and for each value its preimage is FO-definable. Denoting the constant series $d(G) = d$ for all $G \in DG_t(A, B)$ also with $d$, we get the following. If $\varphi$ is almost FO-boolean, then $[\![\varphi]\!]$ has a representation $[\![\varphi]\!] = \sum_{i=1}^{m} d_i \mathbb{1}_{L_i} = \sum_{i=1}^{m} d_i \cap L_i$, where $m \in \mathbb{N}$, $d_i \in D$, $L_i$ are languages definable by an unweighted FO-formula, and $(L_i)_{i=1\ldots m}$ form a partition of $\mathrm{DG}_t(A, B)$.

**Proposition 17.** *Let $\varphi \in \mathrm{MSO}(\mathbb{D})$ such that $[\![\varphi]\!]$ is an FO-step function. Then $[\![\mathrm{Val}_x\, \varphi]\!]$ is recognizable.*

*Proof (sketch).* Let $\mathcal{V} = \mathrm{free}(\mathrm{Val}_x\, \varphi)$ and $\mathcal{W} = \mathcal{V} \cup \{x\}$. Then $[\![\varphi]\!] = \sum_{i=1}^{m} d_i \mathbb{1}_{L_i}$, where $L_i$ are FO-definable languages forming a partition of all of $\mathrm{DG}_t(A_\mathcal{W}, B)$.

Now, we can encode the information in which language a given graph falls into an FO-formula $\tilde{L}$ over an extended alphabet. Using Theorem 2 yields a one-state GA $\tilde{\mathcal{A}}$ with $L(\tilde{\mathcal{A}}) = \tilde{L}$. Finally, we define a wGA $\mathcal{A}$ by adding weights to every tile depending on the state-label at its center and taking special care of the occurrence constraint. Then we can show that $[\![\mathcal{A}]\!] = [\![\mathrm{Val}_x\, \varphi]\!]$.

Let $\varphi \in \mathrm{MSO}(\mathbb{D})$. We call $\varphi$ *FO-restricted* if all unweighted subformulas $\beta$ are FO-formulas and for all subformulas $\mathrm{Val}_x\, \psi$ of $\varphi$, $\psi$ is almost FO-boolean.

These restrictions are motivated in [11] (restriction of $\mathrm{Val}_x\, \psi$) and [19] (restriction to FO) where it is shown that the unrestricted versions of the logic are strictly more powerful than weighted automata on words, resp. pictures. For graphs this is also true, even for the Boolean semiring. We summarize our results.

**Proposition 18.** *If $\mathbb{D}$ is regular, then for every FO-restricted $\mathrm{MSO}(\mathbb{D})$-sentence $\varphi$, there exists a wGA $\mathcal{A}$ with $[\![\mathcal{A}]\!] = [\![\varphi]\!]$.*

*Proof (sketch).* We use structural induction on $\varphi$. One new case is $[\![\beta?\varphi_1 : \varphi_2]\!] = [\![\varphi_1]\!] \cap L(\beta) + [\![\varphi_2]\!] \cap L(\neg\beta)$, which is recognizable by Proposition 11 and Proposition 12, because $L(\beta)$ and $L(\neg\beta)$ are recognizable by a one-state GA, since $\beta$ is an FO-formula. The other cases are covered by regularity of $\mathbb{D}$ and the proven closure results (Lemma 16 and Proposition 17 together with Lemma 15).

Now, we show that every wGA can be simulated by an MSO($\mathbb{D}$)-sentence.

**Proposition 19.** *For every wGA $\mathcal{A}$, there exists an FO-restricted $\mathrm{MSO}(\mathbb{D})$-sentence $\varphi$ with $[\![\mathcal{A}]\!] = [\![\varphi]\!]$.*

Together with Proposition 18, this gives our second main result, a Büchi-like connection of the introduced weighted graph automata and the restricted weighted logic.

**Theorem 20.** *Let $\mathbb{D} = (D, +, \mathrm{Val}, 0)$ be a regular valuation monoid and let $S : \mathrm{DG}_t(A, B) \to D$ be a series. Then the following are equivalent:*

1. *$S$ is recognizable.*
2. *$S$ is definable by an FO-restricted $\mathrm{MSO}(\mathbb{D})$-sentence.*

Examples of a regular valuation monoid are the introduced valuation monoids using average or discounting and all semirings.

## 6 Weighted Automata and Logics for Infinite Graphs

In the following, we extend our results in the weighted setting to infinite graphs. We utilize $\infty$-*valuation monoids* to introduce weighted infinite graph automata.

We call a commutative monoid $(D, +, 0)$ *complete* if it has infinitary sum operations $\sum_I : D^I \to D$ for any index set $I$ such that

- $\sum_{i\in\emptyset} d_i = 0$, $\sum_{i\in\{k\}} d_i = d_k$, $\sum_{i\in\{j,k\}} d_i = d_j + d_k$ for $j \neq k$,
- $\sum_{j\in J}(\sum_{i\in I_j} d_i) = \sum_{i\in I} d_i$ if $\bigcup_{j\in J} I_j = I$ and $I_j \cap I_k = \emptyset$ for $j \neq k$.

**Definition 21.** *An* $\infty$-(graph)-valuation monoid $(D, +, \mathrm{Val}^\infty, 0)$ *consists of a complete monoid $(D, +, 0)$ together with an absorptive* $\infty$-valuation function $\mathrm{Val}^\infty : \mathrm{DG}_t^\infty(D, B) \to D$.

*Example 22.* Let $\bar{\mathbb{R}}_+ = \{x \in \mathbb{R} \mid x \geq 0\} \cup \{\infty, -\infty\}$. Let $t > 1$ be the maximal degree of our graphs and $0 < \lambda < \frac{1}{t-1}$. Then $\mathbb{D} = (\bar{\mathbb{R}}_+, \sup, \mathrm{disc}_\lambda^\infty, -\infty)$, with $\mathrm{disc}_\lambda^\infty(G, u) = \lim\limits_{n\to\infty} \sum\limits_{r=0}^{n} \sum\limits_{\mathrm{dist}(v,u)=r} \lambda^r \,\mathrm{Lab}_G(v)$, is an $\infty$-valuation monoid.

**Definition 23.** *A* weighted infinite graph automaton ($\mathrm{wGA}^\infty$) *over* $\mathrm{DG}_t^\infty(A, B)$ *and $\mathbb{D}$ is a tuple $\mathcal{A} = (Q, \Delta, \mathrm{wt}, \mathrm{Occ}, r)$ where*

- *$\mathcal{A}' = (Q, \Delta, \mathrm{Occ}, r)$ is an infinite graph acceptor over $\mathrm{DG}_t^\infty(A, B)$,*
- *$\mathrm{wt} : \Delta \to D$ is the weight function assigning to every tile of $\Delta$ a value of $D$.*

We transfer the previous notions of *accepting run* and *recognizable series.*

The weighted $\mathrm{MSO}^\infty$-logic for infinite graphs and its fragments is defined as extensions of $\mathrm{MSO}^\infty$ as in the finite case (using $\mathrm{Val}^\infty$ instead of Val) and is denoted by $\mathrm{MSO}^\infty(\mathbb{D})$. Again, the significant difference is that we have the operator $\exists^\infty x$ in our underlying unweighted fragment. Adapting our previous notations and results to the infinite setting, we get our third main result.

**Theorem 24.** *Let $\mathbb{D}$ be a regular $\infty$-valuation monoid and let $S : \mathrm{DG}_t^\infty(A, B) \to D$ be a series. Then $S$ is recognizable by a $wGA^\infty$ if and only if $S$ is definable by an $FO^\infty$-restricted $\mathrm{MSO}^\infty(\mathbb{D})$-sentence.*

The proof mainly follows the proof of Theorem 20. A notably difference is found in the closure under $\mathrm{Val}_x\, \varphi$ (in previous papers the weighted universal quantification). Since we have to deal with the additional quantifier $\exists^\infty x$, we cannot apply Theorem 2. However, Theorem 5 gives us one-state infinite graph acceptors $\mathcal{A}_i$ recognizing $L_i$. Then the automata constructions of Proposition 17 give us a $\mathrm{wGA}^\infty$ $\mathcal{A}$ with $[\![\mathcal{A}]\!] = [\![\mathrm{Val}_x\, \varphi]\!]$.

## 7 Conclusion

Utilizing Bollig and Kuske [4] and a Hanf-like theorem for a first-order logic together with an infinity operator, we have proven a Büchi-like theorem for infinite graphs.

We introduced a weighted automata model over graphs which is robust enough to compute very general weight functions and is adaptable to infinite graphs. We gave new examples for this model, employing average and discounting. Introducing a suitable weighted MSO-logic, we successfully generalized Büchi-like results from the unweighted setting [35] to the weighted setting, from words [11] to graphs and from finite graphs [10] to infinite graphs.

Similar to [10], it can be shown that weighted word, tree, picture, and nested word automata are special instances of these weighted graph automata, which gives us, e.g., results of [11, 16, 19, 26] and [14, 12] as corollaries. Note that these lists are not exhaustive, as graphs are a very general structure comprising many other structures like traces [27], texts [25], distributed systems [5], and others.

Infinite graphs cover for example infinite words [15], infinite trees [29], infinite traces [8], and infinite nested words [9] and it would be interesting to study the expressive power of weighted infinite graph automata over these special classes.
**Acknowledgments:** I want to thank Manfred Droste and Tobias Weihrauch for helpful discussions and insightful remarks on earlier drafts of this paper.

## References

1. Alur, R., Madhusudan, P.: Adding nesting structure to words. J. ACM 56(3), 16:1–16:43 (2009)
2. Berstel, J., Reutenauer, C.: Rational Series and Their Languages, EATCS Monographs in Theoretical Computer Science, vol. 12. Springer (1988)
3. Bollig, B., Gastin, P.: Weighted versus probabilistic logics. In: Developments in Language Theory. LNCS, vol. 5583, pp. 18–38. Springer (2009)
4. Bollig, B., Kuske, D.: Muller message-passing automata and logics. In: LATA 2007. vol. Report 35/07, pp. 163–174. Universitat Rovira i Virgili, Tarragona (2007)
5. Bollig, B., Meinecke, I.: Weighted distributed systems and their logics. In: Logical Foundations of Computer Science. LNCS, vol. 4514, pp. 54–68. Springer (2007)
6. Büchi, J.R.: Weak second-order arithmetic and finite automata. Z. Math. Logik und Grundlagen Math. 6, 66–92 (1960)
7. Chatterjee, K., Doyen, L., Henzinger, T.A.: Quantitative languages. In: Computer Science Logic. LNCS, vol. 5213, pp. 385–400. Springer (2008)
8. Diekert, V., Gastin, P.: LTL is expressively complete for mazurkiewicz traces. In: ICALP 2000. LNCS, vol. 1853, pp. 211–222. Springer (2000)
9. Droste, M., Dück, S.: Weighted automata and logics for infinite nested words. Inf. Comput. In press, a short version appeared in LATA 2014, LNCS 8370, pp. 192-204
10. Droste, M., Dück, S.: Weighted automata and logics on graphs. In: Math. Foundations of Comp. Science, Part I. LNCS, vol. 9234, pp. 192–204. Springer (2015)
11. Droste, M., Gastin, P.: Weighted automata and weighted logics. Theor. Comput. Sci. 380(1-2), 69–86 (2007)
12. Droste, M., Götze, D., Märcker, S., Meinecke, I.: Weighted tree automata over valuation monoids and their characterization by weighted logics. In: Algebraic Foundations in Computer Science. pp. 30–55 (2011)

13. Droste, M., Kuich, W., Vogler, H. (eds.): Handbook of Weighted Automata. EATCS Monographs in Theoretical Computer Science, Springer (2009)
14. Droste, M., Meinecke, I.: Weighted automata and weighted MSO logics for average and long-time behaviors. Inf. Comput. 220, 44–59 (2012)
15. Droste, M., Rahonis, G.: Weighted automata and weighted logics on infinite words. In: Developments in Language Theory. LNCS, vol. 4036, pp. 49–58. Springer (2006)
16. Droste, M., Vogler, H.: Weighted tree automata and weighted logics. Theor. Comput. Sci. 366(3), 228–247 (2006)
17. Eilenberg, S.: Automata, Languages, and Machines, Pure and Applied Mathematics, vol. 59-A. Academic Press (1974)
18. Elgot, C.C.: Decision problems of finite automata design and related arithmetics. Trans. Am. Math. Soc. 98(1), 21–52 (1961)
19. Fichtner, I.: Weighted picture automata and weighted logics. Theory Comput. Syst. 48(1), 48–78 (2011)
20. Fülöp, Z., Stüber, T., Vogler, H.: A Büchi-like theorem for weighted tree automata over multioperator monoids. Theory Comput. Syst. 50(2), 241–278 (2012)
21. Gastin, P., Monmege, B.: A unifying survey on weighted logics and weighted automata. Soft Computing (2015), http://dx.doi.org/10.1007/s00500-015-1952-6
22. Giammarresi, D., Restivo, A., Seibert, S., Thomas, W.: Monadic second-order logic over rectangular pictures and recognizability by tiling systems. Inf. Comput. 125(1), 32–45 (1996)
23. Hanf, W.: Model-theoretic methods in the study of elementary logic. In: Addison, Henkin, Tarski (eds.) The Theory of Models. pp. 132–145. North-Holland (1965)
24. Hoogeboom, H.J., ten Pas, P.: Monadic second-order definable text languages. Theory Comput. Syst. 30(4), 335–354 (1997)
25. Mathissen, C.: Definable transductions and weighted logics for texts. Theor. Comput. Sci. 411(3), 631–659 (2010)
26. Mathissen, C.: Weighted logics for nested words and algebraic formal power series. Logical Methods in Computer Science 6(1) (2010), selected papers of ICALP 2008
27. Meinecke, I.: Weighted logics for traces. In: International Computer Science Symposium in Russia. LNCS, vol. 3967, pp. 235–246. Springer (2006)
28. Rabin, M.O.: Decidability of second order theories and automata on infinite trees. Trans. Am. Math. Soc. 141, 1–35 (1969)
29. Rahonis, G.: Weighted muller tree automata and weighted logics. Journal of Automata, Languages and Combinatorics 12(4), 455–483 (2007)
30. Schützenberger, M.P.: On the definition of a family of automata. Inf. Control 4(2-3), 245–270 (1961)
31. Thatcher, J.W., Wright, J.B.: Generalized finite automata theory with an application to a decision problem of second-order logic. Math. Syst. Theory 2(1), 57–81 (1968)
32. Thomas, W.: On logical definability of trace languages. In: Diekert, V. (ed.) Proc. workshop ASMICS 1989. pp. 172–182. Technical University of Munich (1990)
33. Thomas, W.: On logics, tilings, and automata. In: Int. Coll. on Automata, Languages, and Programming, ICALP 1991. LNCS, vol. 510, pp. 441–454. Springer (1991)
34. Thomas, W.: Elements of an automata theory over partial orders. In: Proc. DIMACS Workshop POMIV '96. pp. 25–40. AMS Press, Inc., New York, NY, USA (1996)
35. Thomas, W.: Handbook of formal languages, vol. 3. chap. Languages, Automata, and Logic, pp. 389–455. Springer, New York, NY, USA (1997)
36. Trakhtenbrot, B.A.: Finite automata and logic of monadic predicates (in Russian). Doklady Akademii Nauk SSR 140, 326–329 (1961)