Universität Leipzig
Institut für Informatik

Diplomarbeit

Charakterisierung erkennbarer Baumreihen über starken Bimonoiden durch gewichtete MSO-Logik

Steffen Märcker

3. Dezember 2010

Betreuer: Prof. Dr. Manfred Droste, Dr. Ingmar Meinecke

Inhaltsverzeichnis

1 Einleitung **3**

2 Baumautomaten und Baumsprachen **5**
2.1 Bäume und Baumautomaten 5
2.2 Bekannte Konstruktionen 7

3 Gewichtete Baumautomaten über starken Bimonoiden **8**
3.1 Bimonoide 8
3.2 Gewichtete Baumautomaten und Baumreihen 12
3.2.1 Gewichtete Baumautomaten über starken Bimonoiden 12
3.2.2 Erkennbare Stufenfunktionen 16
3.2.3 Erkennbarkeit und schwache Erkennbarkeit von Baumreihen . . . 21
3.3 Abschlusseigenschaften erkennbarer Baumreihen 31

4 Erkennbarkeit und Definierbarkeit in wMSO **36**
4.1 MSO-Logik über Bäumen 36
4.2 Gewichtete MSO-Logik über Bäumen 39
4.3 Erkennbarkeit in wMSO definierbarer Baumreihen 49
4.3.1 Charakterisierung erkennbarer Stufenfunktionen 49
4.3.2 Erkennbarkeit der Semantik von wMSO-Formeln 51
4.4 Definierbarkeit erkennbarer Baumreihen in wMSO 61

5 Diskussion **67**

Literaturverzeichnis **69**

1 Einleitung

ENDLICHE Wortautomaten ermöglichen es, reguläre Wortsprachen sowohl zu erkennen als auch zu erzeugen. Julius Richard Büchi gelang es, diese erkennbaren Wortsprachen mithilfe der monadischen Logik zweiter Stufe, kurz MSO, zu charakterisieren [7, 19]. Dieses Ergebnis wurde dann auf erkennbare Baumsprachen, das heißt Mengen von geordneten Bäumen, die durch einen Aufwärtsbaumautomaten erkannt werden, erweitert [11, 28]. Anstelle der $<$-Relation auf den Positionen eines Wortes tritt dabei die Kindrelation $edge_i(x, y)$ für die Positionen eines Baumes. Die erkennbaren Wort- und Baumsprachen haben breite Anwendung in der Informatik gefunden. Zu den bekanntesten gehören beispielsweise reguläre Ausdrücke und Syntaxbäume vieler Programmiersprachen. Im Zusammenspiel mit XML ist die Schemasprache RelaxNG zur Dokumentvalidierung [9, 29], im Gegensatz zu XML-Schema, durch die reiche Theorie erkennbarer Baumsprachen fundiert.

Diese bekannten Theorien definieren erkennbare Sprachen als Mengen erkennbarer Worte beziehungsweise Bäume. Eine natürliche Erweiterung dieses Modells stellen gewichtete Automaten dar, welche klassisch auf einer Gewichtsstruktur, häufig einem Semiring, operieren [27]. Im Wortfall erkennen diese Automaten Funktionen von der Menge der Worte über einem Alphabet in die Menge der Gewichte. Gewichtete Baumautomaten beschreiben hingegen Funktionen von der Menge der Bäume in die Gewichtsmenge. Solche Funktionen nennt man formale Potenzreihen. Die gewichteten Automatenmodelle haben vielfältige Anwendungen hervorgebracht, beispielsweise in der Bildkompression und Sprachverarbeitung [8, 10].

Während die Sprachen ungewichteter Automaten durch zweiwertige Logiken charakterisiert werden, kann man formale Potenzreihen mithilfe gewichteter Logiken beschreiben. Manfred Droste, Paul Gastin und Heiko Vogler [12, 17] haben entsprechende Erweiterungen der monadischen Logik zweiter Stufe im Kontext von Wort- und Baumreihen untersucht. Die Semantik einer Formel der gewichteten monadischen Logik ist demnach eine formale Potenzreihe über einem Semiring. Die hierbei klassisch betrachteten Semiringe sind kommutativ, da in diesen die Interpretation von Konjunktion und Generalisierung unabhängig von der Reihenfolge der Teilformeln ist.

Mehrwertige Logiken, wie zum Beispiel die Gödel-Logiken oder die Quantenlogik [3, 21], basieren auf beschränkten Verbänden. Dies sind Verbände, deren gesamte Grundmenge ein Supremum und ein Infimum besitzt. Im Gegensatz zu Semiringen sind Verbände im Allgemeinen nicht distributiv. Droste und Vogler haben beschränkte Verbände als Gewichtsstrukturen gewichteter Wortautomaten betrachtet [18]. Dabei rückten sie starke Bimonoide, welche eine Verallgemeinerung sowohl der beschränkten Verbände als auch der Semiringe darstellen, in den Fokus. Sie konnten sowohl die Ergebnisse von Kleene zur Rationalität erkennbarer Wortsprachen auf formale Potenzreihen erweitern, als auch Büchis Charakterisierung durch die monadische Logik zweiter Stufe.

Wir[1] werden die Charakterisierung der formalen Potenzreihen von Worten über starken Bimonoiden durch die gewichtete monadische Logik zweiter Stufe auf formale Potenzreihen von Bäumen, sogenannten Baumreihen, übertragen. Hierfür wiederholen wir zunächst sowohl die Erkennbarkeit von Baumsprachen mittels der bekannten Baumautomaten wie auch die Definierbarkeit erkennbarer Baumsprachen durch Sätze der ungewichteten MSO.

Wir betrachten anschließend die Erkennbarkeit von Baumreihen in starken Bimonoiden. In diesen ist die Multiplikation im Allgemeinen nicht kommutativ. Da wir das Gewicht eines Automatenlaufes als Produkt von Transitionsgewichten bestimmen wollen, werden wir feststellen, dass für unsere Definition der Erkennbarkeit die Reihenfolge der Faktoren ausschlaggebend ist. Auf Worten gibt die Folge der Symbole eine natürliche Ordnung der Transitionsgewichte vor. Für Bäume gibt es hingegen verschiedene Möglichkeiten, die Positionen und damit auch die Transitionsgewichte aufzuzählen. Daher führen wir die Erkennbarkeit einer Baumreihe in Abhängigkeit einer gegebenen Aufzählung der Positionen ein.

Neben den Gewichten der Läufe verwenden unsere gewichteten Baumautomaten Finalgewichte zur Bestimmung des Gewichtes eines Baumes. Wir untersuchen zwei Einschränkungen unseres Automatenmodells. *Crisp-deterministische* gewichtete Baumautomaten schränken die Gewichte eines Laufes auf $\{0, 1\}$ ein und charakterisieren eine Teilmenge der formalen Potenzreihen, die erkennbaren Stufenfunktionen. Außerdem definieren wir *schwache* gewichtete Baumautomaten, in denen die Finalgewichte auf die Menge $\{0, 1\}$ beschränkt sind. Wir zeigen, dass für bestimmte Positionsaufzählungen alle erkennbaren Baumreihen ebenfalls mit diesen eingeschränkten Automaten erkennbar sind.

Des Weiteren betrachten wir sogenannte bi-lokal endliche Bimonoide. In diesen induzieren alle endlichen Teilmengen der Grundmenge für beide Operationen Untermonoide, welche ebenfalls endlich sind. Für zwei Beispielaufzählungen werden wir zeigen, dass die in diesen Strukturen erkennbaren Baumreihen stets erkennbare Stufenfunktionen sind. Einerseits handelt es sich dabei um eine Aufzählung, die durch Tiefensuche auf Bäumen bestimmt ist und zum anderen eine Aufzählung, welcher eine lineare Ordnung der Elemente des Bimonoides zugrunde liegt.

Analog dazu verwenden wir die Positionsaufzählungen zur Interpretation der gewichteten MSO über starken Bimonoiden. Wir führen drei Fragmente von MSO ein, da die Semantik von unbeschränkten Formeln der Logik im Allgemeinen nicht erkennbar ist. Mit dem *gestuften* Fragment charakterisieren wir zunächst die erkennbaren Stufenfunktionen.

Als erstes Hauptergebnis zeigen wir, dass sich mit den *eingeschränkten* Formeln genau die Baumreihen beschreiben lassen, welche durch schwache gewichtete Baumautomaten erkannt werden. Im zweiten Hauptergebnis betrachteten wir die Positionsaufzählungen mittels Tiefensuche und einer linearen Ordnung. Wir beweisen, dass in bi-lokal endlichen Bimonoiden das *existenzielle* Fragment genau die mit diesen beiden Positionsaufzählungen erkennbaren Baumreihen charakterisiert.

[1]Die Verwendung des *Pluralis Auctoris* dient in dieser Arbeit der Einbeziehung des Lesers und der Unterstreichung der objektiven Nachprüfbarkeit der Ergebnisse.

2 Baumautomaten und Baumsprachen

2.1 Bäume und Baumautomaten

Wir betrachten in dieser Arbeit endliche geordnete Bäume. In diesen sind die Kindknoten jedes Knotens geordnet. Die Beschriftung eines Knotens bestimmt dabei die Anzahl seiner Kinder.

Die Menge der natürlichen Zahlen bezeichnen wir mit $\mathbb{N} = \{1, 2, 3, \ldots\}$, die der natürlichen Zahlen inklusive 0 mit $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$. Die Positionen eines Baumes sind bestimmt durch seinen *Baumbereich* - eine endliche, nichtleere Menge $D \subseteq \mathbb{N}^*$ mit

- $uv \in D \implies u \in D$ für alle Positionen $u, v \in \mathbb{N}^*$,
- $u.(i+1) \in D \implies u.i \in D$ für alle $u \in \mathbb{N}^*, i \in \mathbb{N}$.

Für eine Position $u \in D$ ist der *Grad $grad(u)$ von u* durch $\max(\{i \mid u.i \in D\})$ bestimmt, wobei $\max(\emptyset) = 0$.

Wir beschriften die Knoten eines Baumes mit Symbolen aus einem *Rangalphabet* (Σ, rk_Σ). Hierbei ist Σ ein Alphabet, das heißt eine endliche, nichtleere Menge, und rk_Σ eine Abbildung $rk_\Sigma : \Sigma \to \mathbb{N}$, welche die Anzahl der Kinder festlegt. Für diese Abbildungen gebe es jeweils mindestens ein $\sigma \in \Sigma$ mit $rk_\Sigma(\sigma) = 0$. Der *maximale Rang* des Rangalphabetes (Σ, rk_Σ) sei $max_\Sigma := \max(\{rk_\Sigma(\sigma) \mid \sigma \in \Sigma\})$. Außerdem bezeichnen wir mit $\Sigma^{(i)} := \{\sigma \in \Sigma \mid rk_\Sigma(\sigma) = i\}$ die Menge der Symbole des Rangs $i \in \mathbb{N}$. Ist rk_Σ bekannt oder aus dem Kontext ersichtlich, sprechen wir im Folgenden verkürzend vom Rangalphabet Σ.

Ein endlicher *geordneter Baum* t ist nun eine partielle Abbildung $t : \mathbb{N}^* \dashrightarrow \Sigma$, wobei

- $t^{-1}(\Sigma) = \{u \in \mathbb{N}^* \mid \exists \sigma \in \Sigma : t(u) = \sigma\}$ ist ein Baumbereich und
- $rk_\Sigma(t(u)) = grad(u)$.

Den Baumbereich eines Baumes t bezeichnen wir mit $\mathrm{pos}(t) := t^{-1}(\Sigma)$. Die Menge aller endlichen Bäume über dem Rangalphabet Σ nennen wir T_Σ. Ein *Teilbaum* eines Baumes t an der Stelle $u \in dom(t)$ ist eine modifizierte partielle Abbildung $t|_u : \mathbb{N}^* \dashrightarrow \Sigma$ mit $t|_u(w) = t(u.w)$.

Abbildung 2.1 zeigt einen Beispielbaum $t \in T_\Sigma$ über dem dreielementigen Rangalphabet $\Sigma := \{\sigma_0, \sigma_1, \sigma_2\}$ mit $rk_\Sigma(\sigma_0) = 0$, $rk_\Sigma(\sigma_1) = 1$ und $rk_\Sigma(\sigma_2) = 2$. Sein Baumbereich ist die Menge $\mathrm{pos}(t) = \{\varepsilon, 1, 2, 1.1, 1.2, 2.1\}$. Ein Teilbaum von t wäre zum Beispiel $t|_2 := \{(\varepsilon, \sigma_1), (1, \sigma_0)\}$.

Endliche geordnete Bäume lassen sich auch mit alternativen Definitionen äquivalent formalisieren. Beispielsweise kann man einen Baum als Term über einer endlichen Menge an Funktionssymbolen, die mindestens ein nullstelliges Symbol enthält, auffassen. Zum Beispiel entspricht der Baum in Abbildung 2.1 dem Term $t = \sigma_2(\sigma_2(\sigma_0, \sigma_0), \sigma_1(\sigma_0))$.

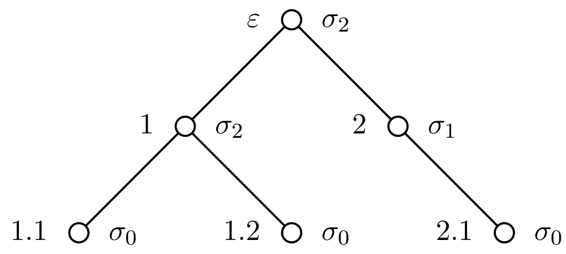


Abbildung 2.1: Beispielbaum

Als Grundlage dieser Arbeit dienen endliche Aufwärts-Baumautomaten [11, 28], deren Eigenschaften wir nun kurz wiederholen. Mit $\wp(M)$ bezeichnen wir die Potenzmenge einer Menge M. Ein *Aufwärts-Baumautomat* ist ein Quadrupel $A = (Q, \Sigma, \delta, F)$. Dabei ist

- Q eine endliche Zustandsmenge,
- Σ ein Rangalphabet,
- $\delta = (\delta_\sigma)_{\sigma \in \Sigma}$ eine Familie von Transitionsfunktionen $\delta_\sigma : Q^{rk_\Sigma(\sigma)} \to \wp(Q)$ und
- $F \subseteq Q$ die Menge der Finalzustände.

Ein Baumautomat heißt *deterministisch*, falls für alle $\sigma \in \Sigma^{(m)}$ und $q_1, \dots, q_m \in Q$ die Menge $\delta_\sigma(q_1, \dots, q_m)$ einelementig ist. Anstatt $\delta_\sigma(q_1, \dots, q_m) = \{q\}$ schreiben wir dann verkürzend $\delta_\sigma(q_1, \dots, q_m) = q$. Diese Definition umfasst zugleich Vollständigkeit, das heißt $\delta_\sigma(q_1, \dots, q_m) \neq \emptyset$.

Ein *Lauf* des Baumautomaten A auf einem Baum $t \in T_\Sigma$ ordnet jeder Position des Baumes einen Zustand zu. Also ist ein Lauf eine Abbildung $r : \mathrm{pos}(t) \to Q$. Für jede Position $w \in \mathrm{pos}(t)$ gelte dabei $r(w) \in \delta_\sigma(r(w.1), \dots, r(w.m))$ mit $t(w) = \sigma \in \Sigma^{(m)}$. Die Menge aller Läufe des Baumautomaten A auf $t \in \Sigma$ bezeichnen wir mit $R_A(t)$. Ist A deterministisch, so ist $R_A(t)$ für alle $t \in T_\Sigma$ einelementig.

Ein Lauf r eines Baumautomaten A heißt *erfolgreich*, falls $r(\varepsilon) \in F$. Die *Baumsprache*, die ein Baumautomat A *erkennt*, ist die Menge $L(A) = \{t \in T_\Sigma \mid \exists r \in R_A(t) : r(\varepsilon) \in F\}$. Eine Baumsprache $L \subseteq T_\Sigma$ heißt *(deterministisch) erkennbar*, falls es einen (deterministischen) Baumautomaten gibt, der L erkennt. Analog zu den regulären Wortsprachen ist jede erkennbare Baumsprache auch deterministisch erkennbar [28, Theorem 1]. Es ist ebenfalls bekannt, dass die Menge der erkennbaren Baumsprachen T_Σ^{rec} (recognizeable tree languages) unter den booleschen Operationen Vereinigung, Komplement und somit auch Schnitt, abgeschlossen ist [28, Theorem 2].

2.2 Bekannte Konstruktionen

Einige Beweise dieser Arbeit greifen grundlegende Ideen bekannter Konstruktionen auf. Deshalb werden wir diese von uns benötigten Ideen kurz vorstellen.

Mithilfe der *Potenzmengenkonstruktion* kann man beweisen, dass jede nichtdeterministisch erkennbare Baumsprache auch von einem deterministischen Baumautomaten erkennbar ist [28, Theorem 1]. Sei dazu $A = (Q, \Sigma, \delta, F)$ ein Baumautomat. Wir konstruieren den deterministischen Baumautomaten $A' := (\wp(Q), \Sigma, \delta', F')$, wobei für alle $\sigma \in \Sigma^{(m)}$

$$\delta'_\sigma(Q_1, \ldots, Q_m) := \bigcup_{q_1 \in Q_1, \ldots, q_m \in Q_m} \delta_\sigma(q_1, \ldots, q_m) \ .$$

Wählen wir nun $F := \{P \in \wp(Q) \mid P \cap F \neq \emptyset\}$, so erhalten wir $L(A') = L(A)$. Setzen wir hingegen $F := \{P \in \wp(Q) \mid P \cap F = \emptyset\}$, so ist $L(A') = T_\Sigma \setminus L(A)$.

Die erkennbaren Baumsprachen sind unter Vereinigung abgeschlossen. In einem möglichen Beweis bildet man dazu den *Vereinigungsautomaten*. Seien also $A = (Q, \Sigma, \delta, F)$ und $A' = (Q', \Sigma, \delta', F')$ zwei Baumautomaten. Wir bilden den Vereinigungsautomaten $A'' := (Q \mathbin{\dot\cup} Q', \Sigma, \delta'', F \mathbin{\dot\cup} F')$ und definieren für alle $\sigma \in \Sigma^{(m)}$

$$\delta''_\sigma(q_1, \ldots, q_m) := \begin{cases} \delta_\sigma \cup \delta'_\sigma & \text{falls } m = 0 \ , \\ \delta_\sigma(q_1, \ldots, q_m) & \text{falls } q_1, \ldots, q_m \in Q \text{ und } m > 0 \ , \\ \delta'_\sigma(q_1, \ldots, q_m) & \text{falls } q_1, \ldots, q_m \in Q' \text{ und } m > 0 \text{ und} \\ \emptyset & \text{sonst.} \end{cases}$$

Nun gilt $L(A'') = L(A) \cup L(A')$.

Aus dem Abschluss unter Vereinigung und Komplement folgt der Abschluss unter Schnittmengenbildung. Dies lässt sich auch direkt durch einen *Produktautomaten* zeigen. Seien wieder $A = (Q, \Sigma, \delta, F)$ und $A' = (Q', \Sigma, \delta', F')$ zwei Baumautomaten. Der Produktautomat ist das Quadrupel $A'' := (Q \times Q', \Sigma, \delta'', F \times F')$. Für alle $\sigma \in \Sigma^{(m)}$ ist die Transitionsfunktion wie folgt definiert:

$$\delta_\sigma((q_1, p_1), \ldots, (q_m, p_m)) := \big\{(q, p) \bigm| q \in \delta_\sigma(q_1, \ldots, q_m) \ \wedge \ p \in \delta'_\sigma(p_1, \ldots p_m)\big\} \ .$$

Wir erhalten damit $L(A'') = L(A) \cap L(A')$.

3 Gewichtete Baumautomaten über starken Bimonoiden

3.1 Bimonoide

Wir werden die bekannten Baumautomaten zu gewichteten Baumautomaten erweitern. In diesem Abschnitt führen wir die dazu benötigten algebraischen Strukturen ein und geben charakteristische Beispiele an. Einige dieser Beispiele werden wir in späteren Beweisen wieder aufgreifen.

Ein *Monoid* ist eine algebraische Struktur $(K, \cdot, 1)$, bestehend aus einer Trägermenge K, einer binären Operation $\cdot$ über K und einem ausgezeichneten Element $1 \in K$. Die Operation $\cdot$ ist hierbei assoziativ und 1 bezüglich $\cdot$ ein neutrales Element (Einselement). Es gilt also $k \cdot (l \cdot m) = (k \cdot l) \cdot m$ und $1 \cdot k = k \cdot 1$ für alle für $k, l, m \in K$. Ein typisches Beispiel ist das, von den Wortsprachen bekannte, freie Monoid $(\Sigma^*, \cdot)$ über einer nicht notwendigerweise endlichen Menge Σ. Dabei ist $\cdot$ die Konkatenation und $\Sigma^* := \{w \mid w = \varepsilon \vee \exists v \in \Sigma^*, a \in \Sigma : w = v \cdot a\}$

Ein *Bimonoid* ist eine algebraische Struktur $(K, +, \cdot, 0, 1)$, wobei sowohl $(K, +, 0)$ als auch $(K, \cdot, 1)$ Monoide sind [15]. Wir nennen die Operation $+$ Addition und $\cdot$ Multiplikation. Ist das Bimonoid $(K, +, \cdot, 0, 1)$ fest oder aus dem Kontext bekannt, identifizieren wir es mit seiner Trägermenge K. Ein Bimonoid heißt *stark*, falls die Operation $+$ kommutativ ist und 0 bezüglich $\cdot$ ein absorbierendes Element ist (multiplikatives Nullelement), das heißt es gilt $0 \cdot k = 0 = k \cdot 0$ für alle $k \in K$. Ist K stark und die Multiplikation distributiv über der Addition, also $k \cdot (l + m) = (k \cdot l) + (k \cdot m)$ und $(k + l) \cdot m = (k \cdot m) + (l \cdot m)$ für $k, l, m \in K$, so ist K ein *Semiring*. Ist $\cdot$ kommutativ, so nennen wir K *kommutativ*.

Um gewichtete Logik später in Bimonoiden zu interpretieren, benötigen wir eine Negations-Operation. Ein Bimonoid *mit Negation* ist eine Struktur $(K, +, \cdot, \bar{\ }, 0, 1)$. Dabei bildet $(K, +, \cdot, 0, 1)$ ein Bimonoid und $\bar{\ } \colon K \to K$ ist eine Funktion derart, dass $\bar{0} = 1$ und $\bar{1} = 0$. Offensichtlich lässt sich jedes Bimonoid um eine solche Operation erweitern, indem man $\bar{k} \in K$ für $k \in K \setminus \{0, 1\}$ beliebig wählt.

Ist f eine Funktion mit dem Definitionsbereich $\mathrm{dom}(f)$ und $N \subseteq \mathrm{dom}(f)$, so bezeichnen wir mit $f|_N$ die Einschränkung von f auf N, wobei $f|_N(n) := f(n)$ für alle $n \in N$. Sei $(K, \cdot, 1)$ ein Monoid und $L \subseteq K$. Ein Untermonoid von K ist ein Monoid $(K', \cdot|_{K' \times K'}, 0, 1)$ mit $K' \subseteq K$. Eine Menge $L \subseteq K$ *induziert* das Untermonoid K', falls K' die kleinste Menge M ist, sodass $L \subseteq M$ und $(m \cdot|_{M \times M} n) \in M$ für alle $m, n \in M$ gilt. Ein Bimonoid K heißt *additiv lokal endlich*, falls für jede endliche Teilmenge $L \subseteq K$ das durch L induzierte Untermonoid von $(K, +, 0)$ endlich ist. Analog dazu betrachten wir ebenfalls *multiplikativ lokal endliche* Bimonoide. Ist ein Bimonoid sowohl additiv als auch multiplikativ lokal endlich, so nennen wir es *bi-lokal endlich*.

Falls $k + k = k$ in einem Bimonoid K für jedes $k \in K$ gilt, nennen wir K *additiv idempotent*. Gilt $k \cdot k = k$, so ist K *multiplikativ idempotent*. Ein Bimonoid heißt *bi-idempotent*, falls es sowohl additiv als auch multiplikativ idempotent ist.

Alle bi-idempotenten, kommutativen starken Bimonoide sind bi-lokal endlich. Sei dazu $L := \{k_1, \ldots, k_m\} \subseteq K$ endlich und L' das durch L induzierte Untermonoid bezüglich Multiplikation. Da $\cdot$ kommutativ ist, hat jedes Element l aus L' die Form $l = k_1^{n_1} \cdot \ldots \cdot k_m^{n_m}$ für $n_1, \ldots, n_m \in \mathbb{N}$. Aus der Idempotenz folgt, dass für jedes $l \in L'$ die Exponenten $n_1, \ldots, n_m$ tatsächlich aus $\{0, 1\}$ stammen. Also ist L' endlich. Da für die Addition die gleiche Argumentation anwendbar ist, muss K bi-lokal endlich sein.

Betrachten wir nun einige Beispiele für Bimonoide.

Beispiel 3.1.1. Sei $B_\omega := (\{0,1\}^* \cup \{1^\omega\}, \oplus, \cdot, 1^\omega, \varepsilon)$. Dabei ist die Multiplikation $\cdot$ die übliche Konkatenation und $w \cdot 1^\omega := 1^\omega \cdot w := 1^\omega$ für alle $w \in K_\omega$. Die Addition $\oplus$ entspricht einer stellenweisen Minimumbildung, von rechts beginnend. Wir definieren diese induktiv für $x, y \in \{0, 1\}$ und $v, w \in K_\omega$ durch

$$x \oplus \varepsilon := \varepsilon \oplus x := \varepsilon, \quad 0 \oplus 1 := 1 := 0 \oplus 0 \quad \text{und} \quad vx \oplus wy := (v \oplus w) \cdot (x \oplus y) \ .$$

Offensichtlich ist die Multiplikation $\cdot$ assoziativ, ε ihr neutrales Element und 1^ω absorbierend. Die induktive Definition der Addition $\oplus$ ist assoziativ, erhält die Kommutativität und 1^ω verhält sich neutral. Damit ist B_ω ein starkes Bimonoid. Tatsächlich lässt sich aus jeder Menge M und einer Minimumbildung mit einem maximalen Element in M ein starkes Bimonoid der Form

$$M_\omega = (M^* \cup \{\max(M)^\omega\}, \oplus, \cdot, \max(M)^\omega, \varepsilon)$$

konstruieren.

Beispiel 3.1.2. Ein weiteres Beispiel lässt sich mit den reellen Mengen $K_\varepsilon := \{0\} \cup [\varepsilon, 1]$ für $0 < \varepsilon < 1$ finden. Sei ε fix und das Biomonoid $K := (K_\varepsilon, \oplus, \odot, 0, 1)$, wobei für alle $k, k' \in K_\varepsilon$ die Operationen wir folgt definiert sind:

$$\begin{aligned} k \oplus k' &:= \min(1, k + k') \ , \\ k \odot k' &:= \begin{cases} k \cdot k' & \text{falls } k \cdot k' \in K_\varepsilon \text{ und} \\ 0 & \text{sonst.} \end{cases} \end{aligned}$$

Das Bimonoid K ist offensichtlich stark und kommutativ. Anwendung findet es beispielsweise bei der Modellierung von Rechenoperationen mit eingeschränkter Genauigkeit.

Nach der obigen Definition ist jeder Semiring ein starkes Bimonoid. Ein typisches Beispiel für einen Semiring sind die natürlichen Zahlen inklusive 0 mit Addition und Multiplikation $\mathbb{N}_0 = (\mathbb{N}_0, +, \cdot, 0, 1)$.

Beispiel 3.1.3. Sei $\mathbb{N}^*_\wp = (\wp(\mathbb{N}^*), \cup, \cdot, \emptyset, \{\varepsilon\})$, wobei $\cup$ die übliche Mengenvereinigung ist und $V \cdot W = \{vw \mid v \in V \wedge w \in W\}$. Bekanntlich ist $(\wp(\mathbb{N}^*), \cup, \emptyset)$ ein Monoid und $\cup$ kommutativ. Das Monoid $(\wp(\mathbb{N}^*), \cdot, \{\varepsilon\})$ ist eine Erweiterung des freien Monoides über $\mathbb{N}^*$, wobei die leere Menge $\emptyset$ das absorbierende Element darstellt. Tatsächlich ist $\mathbb{N}_\wp$ sogar ein Semiring, da die erweiterte Konkatenation über der Mengenvereinigung distributiv ist:

$$\begin{aligned}
U \cdot (V \cup W) &= \{uu' \mid u \in U \ \wedge\ (u' \in V \ \vee\ u' \in W)\} \\
&= \{uv \mid u \in U \ \wedge\ v \in V\} \cup \{uw \mid u \in U \ \wedge\ w \in W\} \\
&= (U \cdot V) \cup (U \cdot W) \\
(U \cup V) \cdot W &= \{ww' \mid (w \in U \ \vee\ w \in V) \ \wedge\ w' \in W\} \\
&= \{uw \mid u \in U \ \wedge\ w \in W\} \cup \{uv \mid v \in V \ \wedge\ w \in W\} \\
&= (U \cdot W) \cup (V \cdot W)
\end{aligned}$$

Auch hier lässt sich aus jeder Grundmenge M ein zu $\mathbb{N}^*_\wp$ analoger Semiring konstruieren, beispielsweise mit einem Alphabet Σ.

Ein *Verband* ist eine Struktur $(K, \vee, \wedge)$ [2, 23]. Die Operationen $\vee$ (Supremum) und $\wedge$ (Infimum) sind dabei assoziativ, kommutativ, idempotent und *absorptiv*, das heißt, für alle $k_1, k_2 \in K$ gilt $k_1 \wedge (k_1 \vee k_2) = k_1$ und $k_1 \vee (k_1 \wedge k_2) = k_1$. Man kann Verbände alternativ auch als Halbordnungen definieren, in welchen je zwei Elemente ein Infimum und ein Supremum besitzen.

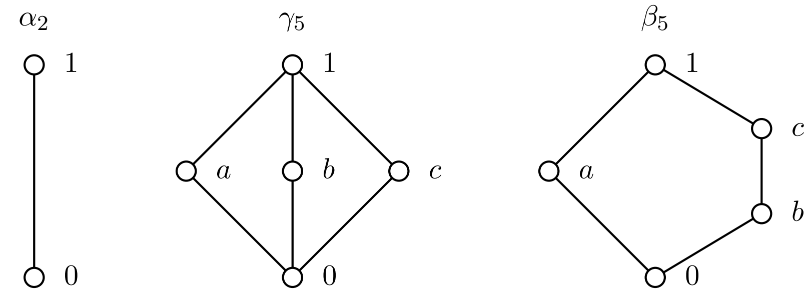


Abbildung 3.1: Beispielverbände

Ein Verband K heißt *nach unten beschränkt*, falls es ein Element $0 \in K$ gibt, das bezüglich $\vee$ neutral ist. 0 ist dann das Infimum von K, das heißt, es gilt $0 \wedge k = 0$ für alle $k \in K$. Wenn ein bezüglich $\wedge$ neutrales Element $1 \in K$ existiert, heißt der Verband K *nach oben beschränkt*. 1 ist dann das Supremum von K, das bedeutet, es gilt $1 \wedge k = 1$ für alle $k \in K$. Ein *beschränkter* Verband ist ein nach unten und oben beschränkter Verband. Jedes bi-idempotentes, kommutatives und starkes Bimonoid, dessen Operationen $+$ und $\cdot$ absorptiv sind und in dem 1 ein additives Nullelement ist, ist ein beschränkter Verband. Dabei entspricht die Addition der Supremumbildung und die Multiplikation der Infimumbildung. Da $\vee$ und $\wedge$ kommutativ und idempotent sind, ist jeder (beschränkte) Verband bi-lokal endlich.

Ist in einem Verband $\wedge$ $(\cdot)$ distributiv über $\vee$ $(+)$, dann ist $\vee$ auch distributiv über $\wedge$. Es ist bekannt, dass ein Verband genau dann distributiv ist, wenn er einen der beiden Beispielverbände γ_5 und β_5 aus Abbildung 3.1 enthält.

Typische Verbände sind beispielsweise $(\mathbb{N}, \min, \max)$, die (zweielementige) boolesche Algebra und der Teilmengenverband $(\wp(M), \cup, \cap, M, \emptyset)$. Ein weiteres interessantes Beispiel ist die Lindenbaum Algebra, deren Elemente aussagenlogische Formeln sind, welche durch die Folgerungsrelation $\models$ der intuitionistischen Logik geordnet werden. Die Verbandstheorie kennt somit viele natürliche Beispiele von Bimonoiden.

Abbildung 3.2 illustriert die Zusammenhänge zwischen Bimonoiden, Verbänden und Semiringen.

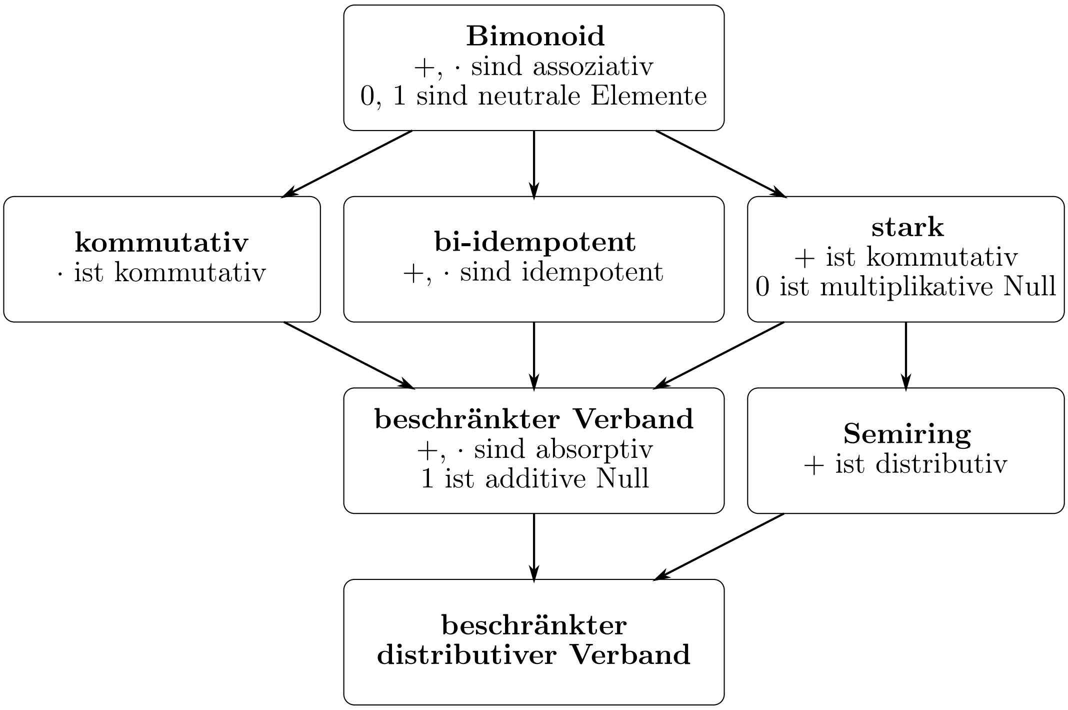


Abbildung 3.2: Zusammenhänge zwischen Bimonoiden, Verbänden und Semiringen

3.2 Gewichtete Baumautomaten und Baumreihen

In diesem Abschnitt führen wir gewichtete Baumautomaten über starken Bimonoiden ein. Diese entscheiden nicht, wie die gewöhnlichen Baumautomaten, die Zugehörigkeit eines Baumes zu einer Baumsprache, sondern definieren Funktionen, welche jedem Baum ein Element des zugrunde liegenden Bimonoides zuordnen. Hierfür führen wir den Begriff der Erkennbarkeit solcher Funktionen durch ein allgemeines Automatenmodell ein. Wir betrachten außerdem zwei Einschränkungen dieser Automaten und deren Auswirkungen auf die Erkennbarkeit. Es stellt sich heraus, dass das gewählte Bimonoid hierbei eine entscheidende Rolle spielt.

3.2.1 Gewichtete Baumautomaten über starken Bimonoiden

Im Folgenden bezeichnen wir die Menge aller partiellen Funktionen aus einer Menge M in eine Menge N mit $[M \dashrightarrow N]$.

Definition 3.2.1. Sei K ein starkes Bimonoid. Ein *endlicher gewichteter Aufwärts-Baumautomat* (weighted bottom-up final state tree automaton, wta) ist ein Quadrupel $M = (Q, \Sigma, \mu, \gamma)$, dabei sei

- Q eine endliche Zustandsmenge,
- Σ ein Rangalphabet,
- $\mu = (\mu_m)_{m \leq max_\Sigma}$ eine Familie von Transitionsfunktionen $\mu_m : \Sigma^{(m)} \to K^{Q^m \times Q}$ und
- $\gamma : Q \to K$ eine Funktion der Finalgewichte.

Ein *Lauf* eines gewichteten Baumautomaten M auf einem Baum t ist eine Abbildung $r : \mathrm{pos}(t) \to Q$. Da der Definitionsbereich $\mathrm{dom}(r)$ ein Baumbereich ist, können wir einen Lauf r als einen geordneten Baum über dem Alphabet Q auffassen. Dabei ist Q kein Rangalphabet. $R_M(t)$ bezeichnet die Menge der Läufe eines Baumautomaten M auf einem Baum t. Die Menge aller Läufe eines Baumautomaten auf allen Bäumen sei $R_M := \bigcup_{t \in T_\Sigma} R_M(t)$.

Für ein Symbol $\sigma \in \Sigma^{(m)}$ und Zustände $q_1, \ldots, q_m, q \in Q$ benutzen wir für den Funktionswert $(\mu_m(\sigma))((q_1, \ldots, q_m), q)$ die Abkürzung $\mu(\sigma)_{q_1 \ldots q_m, q}$. Sei $t \in T_\Sigma$ ein Baum. Ein gewichteter Baumautomat M ordnet durch einen Lauf $r \in R_M(t)$ jeder Position $w \in \mathrm{pos}(t)$ eines Baumes das Gewicht $\mu(\sigma)_{r(w.1) \ldots r(w.m), r(w)}$ zu, wobei $\sigma = t(w)$ und $m = rk(\sigma)$. Die daraus resultierende partielle Funktion $t_r : \mathbb{N}^* \dashrightarrow K$ nennen wir *Gewichtsbaum von r auf t in M*. Die Menge aller Gewichtsbäume über einem Bimonoid K sei $T_K := \{t : \mathbb{N}^* \dashrightarrow K \mid \mathrm{dom}(t) \text{ ist Baumbereich}\}$. Somit definiert ein Baumautomat für alle $t \in T_\Sigma$ und $r \in R_M(t)$ eine partielle Funktion $M : T_\Sigma \times R_M \dashrightarrow [\mathbb{N}^* \dashrightarrow K]$ mit $M(t, r)(w) := \mu(t(w))_{r(w.1) \ldots r(w.m), r(w)}$. Wir verallgemeinern mit der partiellen Funktion M die Bezeichnung $\mathrm{wt}(t, r, w)$, welche beispielsweise Droste und Vogler benutzen [17].

Das Gewicht eines Laufes auf einem Baum t wird üblicherweise als Produkt der Gewichte an den Positionen des Laufes berechnet. Da das Bimonoid K im Allgemeinen nicht kommutativ ist, hängt das Ergebnis von der Reihenfolge der Faktoren ab. Deshalb benötigen wir eine Methode, um die Gewichte eines Gewichtsbaumes anzuordnen, das heißt eine *Aufzählung der Positionen* des Gewichtsbaumes $M(t, r)$. Um die Vergleichbarkeit von Baumautomaten untereinander zu gewährleisten, muss diese Methode unabhängig vom betrachteten Baumautomaten sein.

Definition 3.2.2. Eine partielle Funktion $en : T_K \dashrightarrow [\mathbb{N} \dashrightarrow \mathbb{N}^*]$ heißt *Positionsaufzählung für Gewichtsbäume*, falls für alle $t \in T_K$ gilt:

- $en(t) \in \mathrm{pos}(t)^{[1,n]}$ mit $n = |\mathrm{pos}(t)|$ und
- $en(t)$ ist zwischen $[1, n]$ und $\mathrm{pos}(t)$ bijektiv.

Mithilfe einer Positionsaufzählung für Gewichtsbäume lässt sich nun das Gewicht der Läufe eines gewichteten Baumautomaten eindeutig definieren.

Definition 3.2.3. Sei K ein starkes Bimonoid, M ein gewichteter Baumautomat und en eine Positionsaufzählung für Gewichtsbäume. Sei weiterhin $t \in T_\Sigma$ und $r : \mathrm{pos}(t) \to Q$ ein Lauf von M auf t. Das *Gewicht des Laufes* r ist bestimmt durch:

$$\mathrm{wt}_M^{en}(t, r) := \prod_{i=1}^{|\mathrm{pos}(t)|} M(t, r)[en(M(t, r))(i)] \ .$$

Bei den erkennbaren Baumsprachen wird ein Baum $t \in T_\Sigma$ von einem Baumautomaten A genau dann erkannt, wenn es einen erfolgreichen Lauf von A auf t gibt. Ein gewichteter Baumautomat $M = (Q, \Sigma, \mu, \gamma)$ ordnet hingegen jedem Lauf $r \in R_M(t)$ an der Wurzel ein Finalgewicht $\gamma(r(\varepsilon))$ zu. In Anlehnung an ungewichtete Baumautomaten nennen wir einen Lauf $r \in R_M(t)$ genau dann *erfolgreich*, wenn $\gamma(r(\varepsilon)) \neq 0$. Wir lassen damit zu, dass ein Lauf r erfolgreich sein kann, doch sein Gewicht $\mathrm{wt}_M^{en}(t, r)$ trotzdem 0 ist.

Das Verhalten eines gewichteten Baumautomatens definieren wir nun als Funktion von der Menge der Bäume T_Σ in das Bimonoid K. Derartige Funktionen nennt man *formale Potenzreihen* und die Menge aller formalen Potenzreihen von T_Σ in K bezeichnet man mit $K\langle\langle T_\Sigma\rangle\rangle$.

Definition 3.2.4. Sei K ein starkes Bimonoid und en eine Positionsaufzählung für Gewichtsbäume. Ein gewichteter Baumautomat M *erkennt mit en* die formale Potenzreihe $S_M^{en} \in K\langle\langle T_\Sigma\rangle\rangle$, welche für alle $t \in T_\Sigma$ bestimmt ist mit:

$$(S_M^{en}, t) := \sum_{r \in R_M(t)} \mathrm{wt}_M^{en}(t, r) \cdot \gamma(r(\varepsilon)) \ .$$

Gibt es zu einer formalen Potenzreihe $S \in K\langle\langle T_\Sigma\rangle\rangle$ einen wta, der S mit en erkennt, so heißt S *en-erkennbar*. Die Klasse aller en-erkennbaren Potenzreihen bezeichnen wir mit $K^{en-rec}\langle\langle T_\Sigma\rangle\rangle$. Offenbar ist die von einem gewichteten Baumautomaten erkannte formale Potenzreihe von der zugrunde liegenden Positionsaufzählung für Gewichtsbäume abhängig. Wir wollen nun verschiedene dieser Aufzählungen betrachten.

Eine Möglichkeit ist, die Positionsaufzählung für Gewichtsbäume aus einer gegebenen linearen Ordnung der Menge der Positionen $\mathbb{N}^*$ herzuleiten. Sei die Menge $(\mathbb{N}^*, <)$ der Positionen linear geordnet.

Wir nennen eine Positionsaufzählung en für Gewichtsbäume *strukturerhaltend*, wenn mit ihr die Implikation $i < j \implies en(i) < en(j)$ für alle Gewichtsbäume $t_w \in T_K$ und $i, j \in [1, |\mathrm{pos}(t_w)|]$ gültig ist. Offenbar sind diese Positionsaufzählungen unabhängig den Gewichten $t_w(u)$ an einer Position $u \in \mathrm{pos}(t_w)$.

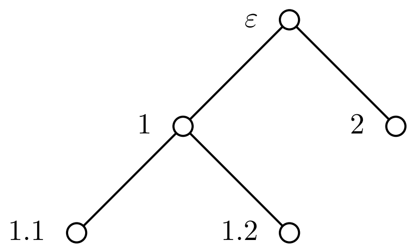
Die im Kontext von Aufwärts-Baumautomaten wohl am natürlichsten erscheinende Ordnung auf $\mathbb{N}^*$ wird durch die Tiefensuche in einem Baum induziert. Ihre Anwendung zur Gewichtsberechnung stimmt überein mit dem intuitiven Verständnis eines gewöhnlichen Baumautomatenlaufes, der an den Blättern beginnt und an der Wurzel endet.

Definition 3.2.5. Die *Ordnung nach Tiefensuche* $<_{TS}$ ist für alle $u, v \in \mathbb{N}^*$ bestimmt durch:

$$\begin{aligned} u <_{TS} v \iff & \exists x \in \mathbb{N}\ \exists u' \in \mathbb{N}^* : \\ & u = x.u' \land \\ & \forall y \in \mathbb{N}\ \forall v' \in \mathbb{N}^* : [v = y.v' \to (x < y \lor (x = y \land u' <_{TS} v'))]. \end{aligned}$$

Für alle $t_w \in T_K$ und $i, j \in [1, |\mathrm{pos}(t_w)|]$ ist die *Positionsaufzählung für Gewichtsbäume durch Tiefensuche* $TS : T_K \dashrightarrow [\mathbb{N} \dashrightarrow \mathbb{N}^*]$ eindeutig bestimmt durch:

$$i < j \iff TS(t_w)(i) <_{TS} TS(t_w)(j) .$$



$1.1 < 1.2 < 1 < 2 < \varepsilon$

Abbildung 3.3: Positionsaufzählung nach Tiefensuche

Abbildung 3.3 illustriert die Positionsaufzählung durch Tiefensuche an einem Beispielbaum. Die von uns gewählte Tiefensuche ordnet die Kinder $w.1, \ldots, w.m$ einer Position w aufsteigend von links nach rechts. Man kann natürlich andere Varianten der Tiefensuche betrachten, beispielsweise mit einer absteigenden Ordnung der Kinder.

Eine weitere Möglichkeit ist, die Positionsaufzählung für Gewichtsbäume aus einer linearen Ordnung des Bimonoids K zu konstruieren. Deren Existenz kann nach dem Wohlordnungsprinzip angenommen werden.

Definition 3.2.6. Eine *ordnungserhaltende Positionsaufzählung für Gewichtsbäume* ist eine partielle Funktion $en^< : T_K \dashrightarrow [\mathbb{N} \dashrightarrow \mathbb{N}^*]$ derart, dass für alle $t_w \in T_K$ und $i, j \in [1, |\mathrm{pos}(t_w)|]$ gilt:

$$i < j \implies t_w(en^<(t_w)(i)) \leq t_w(en^<(t_w)(j)) .$$

Da verschiedene Positionen eines Gewichtsbaumes das gleiche Gewicht tragen können, gibt es im Allgemeinen mehrere verschiedene Positionsaufzählungen für Gewichtsbäume durch $<$. Es stellt sich aber heraus, dass die Anordnung von Gewichten trotzdem ausschließlich von der linearen Ordnung $<$ abhängt.

Proposition 3.2.7. *Seien $t_w \in T_K$ ein Gewichtsbaum und $en_1^<$ und $en_2^<$ zwei ordnungserhaltende Positionsaufzählung für Gewichtsbäume. Dann gilt für alle $i \in [1, |\mathrm{pos}(t)|]$:*

$$t_w(en_1^<(t_w)(i)) = t_w(en_2^<(t_w)(i) \ .$$

Beweis. Für alle $i < j \in [1, \mathrm{pos}(t)]$ gilt sowohl

$$t_w(en_1^<(t_w)(i)) \leq t_w(en_1^<(t_w)(j)) \text{ als auch}$$
$$t_w(en_2^<(t_w)(i)) \leq t_w(en_2^<(t_w)(j)).$$

Da $en_1^<$ und $en_2^<$ zwischen $[1, |\mathrm{pos}(t)|]$ und $\mathrm{pos}(t)$ bijektiv sind, gilt für alle $i \in [1, |\mathrm{pos}(t)|]$

$$t_w(en_1^<(t_w)(i)) = t_w(en_2^<(t_w)(i)) \ .$$

□

Sei M ein gewichteter Baumautomat und $en_1^<$ und $en_2^<$ zwei ordnungserhaltende Positionsaufzählung für Gewichtsbäume. Für alle $t \in T_\Sigma$ und $r \in R_M(t)$ folgt aus der obigen Aussage sofort die Gleichheit der Laufgewichte:

$$\mathrm{wt}_M^{en_1^<}(t, r) = \mathrm{wt}_M^{en_2^<}(t, r) \ .$$

Somit sind alle ordnungserhaltenden Positionsaufzählungen für Gewichtsbäume äquivalent zueinander bezüglich der Erkennbarkeit formaler Potenzreihen in einem linear geordneten Bimonoid (K, $<$). Deshalb werden wir forthin von einer ordnungserhaltenden Positionsaufzählung $en^<$ sprechen, anstatt einen konkreten Repräsentanten der äquivalenten Positionsaufzählungen zu benennen.

Eine formale Potenzreihe S, die $en^<$-erkennbar ist, nennen wir abkürzend *$<$-erkennbar* (sprich: linear erkennbar) und $S_M^<$ steht für die formale Potenzreihe eines gewichteten Baumautomaten, der mit der Positionsaufzählung $en^<$ erkennt. Analog dazu bezeichnen wir die Klasse der $<$-erkennbaren Baumreihen auch mit $K^{<-rec}\langle\langle T_\Sigma\rangle\rangle$ anstelle von $K^{en^<-rec}\langle\langle T_\Sigma\rangle\rangle$.

Ist das zugrunde liegende Bimonoid kommutativ, so sind die Gewichtungen der Läufe unabhängig von den Positionsaufzählungen für Gewichtsbäume. Man erhält das gleiche Gewicht wie bei bekannten gewichteten Baumautomaten über kommutativen Semiringen.

Proposition 3.2.8. *Sei K ein kommutatives starkes Bimonoid, M ein gewichteter Baumautomat und $t \in T_\Sigma$ ein Baum. Dann gilt für alle Positionsaufzählungen en_1 und en_2:*

$$\mathrm{wt}_M^{en_1}(t, r) = \prod_{w \in \mathrm{pos}(t)} \mu(t(w))_{r(w.1)\ldots r(w.m), r(w)} = \mathrm{wt}_M^{en_2}(t, r) \ .$$

Somit ist in einem kommutativen Bimonoid $S_M^{en_1} = S_M^{en_2}$ für jeden Baumautomaten M und alle Aufzählungen en_1 und en_2. Es folgt sofort $K^{en_1-rec}\langle\langle T_\Sigma\rangle\rangle = K^{en_2-rec}\langle\langle T_\Sigma\rangle\rangle$.

3.2.2 Erkennbare Stufenfunktionen

Sei M eine Menge und $N \subseteq M$ beliebig. Die charakteristische Funktion der Menge N ist die Abbildung $\mathbb{1}_N : M \to \{0,1\}$, welche für alle $x \in M$ durch die Äquivalenz $\mathbb{1}_N(x) = 1 \iff x \in N$ beschrieben wird . Sei K ein starkes Bimonoid, $k \in K$ und $S : T_\Sigma \to K$ eine formale Potenzreihe. Die *Skalarprodukte* von k und S sind die formalen Potenzreihen $(k \cdot S, t) := k \cdot (S,t)$ und $(S \cdot k, t) := (S,t) \cdot k$ für alle $t \in T_\Sigma$.

Eine *erkennbare Stufenfunktion* ist eine formale Potenzreihe $S : T_\Sigma \to K$, für welche es eine Familie von erkennbaren Baumsprachen $L_1, \ldots, L_n \in \wp(T_\Sigma)$ und Elemente $k_1, \ldots, k_n \in K$ gibt, sodass $S = \sum_{i=1}^n k_i \cdot \mathbb{1}_{L_i}$. Wir fordern zusätzlich, dass die Baumsprachen $L_1, \ldots, L_n$ die Menge T_Σ partitionieren, und die Elemente $k_1, \ldots, k_n$ paarweise verschieden sind. Dies stellt keine Einschränkung dar, da die Menge der erkennbaren Baumsprachen unter Vereinigung und Schnitt abgeschlossen ist. Abbildung 3.4 verdeutlicht die Herkunft des Begriffs Stufenfunktion.

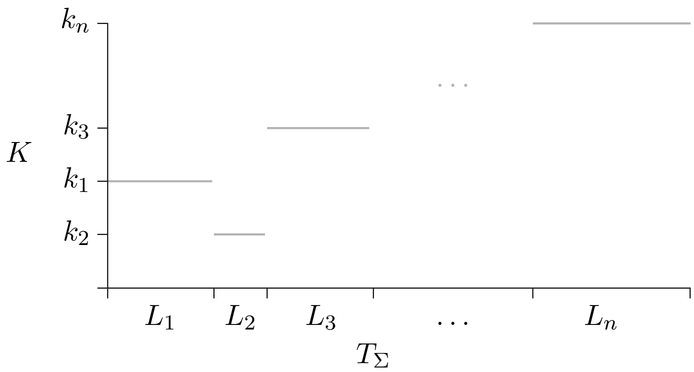


Abbildung 3.4: Graph einer Stufenfunktion über T_Σ

Im Folgenden nutzen wir häufig die Abbildung img, die jeder Funktion die Menge ihrer Bilder zuordnet; $\mathrm{img}(f) := \{y \mid \exists x : f(x) = y\}$. Ist M eine Menge oder Familie von Funktionen, so setzen wir $\mathrm{img}(M) := \bigcup_{f \in M} \mathrm{img}(f)$.

Die Menge der erkennbaren Stufenfunktionen lässt sich durch eine Einschränkung der gewichteten Baumautomaten charakterisieren. Ein *crisp-deterministischer Baumautomat* ist ein gewichteter Baumautomat $M = (Q, \Sigma, \mu, \gamma)$ mit folgenden Einschränkungen:

- Die Transitionsgewichte sind auf $\{0,1\} \subseteq K$ beschränkt (crisp), das heißt, für alle $\sigma \in \Sigma^{(m)}$ gilt
$$\mathrm{img}(\mu_m(\sigma)) \subseteq \{0,1\} \ .$$
- Die Transitionsfunktionen sind deterministisch, das heißt, für alle $\sigma \in \Sigma^{(m)}$ und $q_1, \ldots, q_m \in Q$ gilt
$$(\forall q : \mu_m(\sigma)_{q_1 \ldots q_m, q} = 0) \vee (\exists! q \in Q : \mu_m(\sigma)_{q_1 \ldots q_m, q} = 1) \ .$$

In einem Bimonoid kommutieren 0 und 1 mit allen Elementen. Da die Gewichte der Transitionen lediglich aus $\{0,1\}$ stammen, spielt die Reihenfolge der Multiplikation zur Bestimmung des Laufgewichtes keine Rolle. Somit gilt $S_M^{en_1} = S_M^{en_2}$ für alle

Positionsaufzählung für Gewichtsbäume en_1 und en_2. Bei einen crisp-deterministischen Baumautomaten M schreiben wir deshalb wt_M und S_M anstelle von wt_M^{en} und S_M^{en}. Eine formale Potenzreihe nennen wir *crisp-erkennbar*, falls es einen crisp-deterministischen gewichteten Baumautomaten M mit $S \in K\langle\langle T_\Sigma \rangle\rangle$ mit $S = S_M$ gibt.

Lemma 3.2.9. *Sei $S \in K\langle\langle T_\Sigma \rangle\rangle$. Dann ist S eine erkennbare Stufenfunktion genau dann, wenn S crisp-erkennbar ist.*

Beweis. ($\Rightarrow$) Weil S eine erkennbare Stufenfunktion ist, gibt es erkennbare Baumsprachen $L_1, \dots, L_n$, die T_Σ partitionieren, und paarweise verschiedene Elemente $k_1, \dots, k_n \in K$ so, dass $S = \sum_{i=1}^n k_i \cdot \mathbb{1}_{L_i}$. Da $1 \cdot k = k \cdot 1$ gilt $S = \sum_{i=1}^n \mathbb{1}_{L_i} \cdot k_i$.

Es gibt deterministische Baumautomaten $A_1, \dots, A_n$ mit $A_i = (Q_i, \Sigma, \delta_i, F_i)$ und $L(A_i) = L_i$. Wir konstruieren nun zu jedem A_i einen crisp-deterministischen wta mit $M_i := (Q_i, \Sigma, \mu_i, \gamma_i)$, wobei

$$\mu_i(\sigma)_{q_1 \dots q_m, q} := \begin{cases} 1 & \text{falls } \delta_{i,\sigma}(q_1, \dots, q_m) = q \ , \\ 0 & \text{sonst} \end{cases}$$

und

$$\gamma_i(q) := \begin{cases} k_i & \text{falls } q \in F_i \ , \\ 0 & \text{sonst.} \end{cases}$$

Jeder so konstruierte gewichtete Baumautomat M_i erkennt die formale Potenzreihe $S_{M_i} = \mathbb{1}_{L(A_i)} \cdot k_i = \mathbb{1}_{L_i} \cdot k_i$. Wir bilden nun den wta $M := (Q_M, \Sigma, \mu_M, \gamma_M)$ als Vereinigung der wta $M_1, \dots, M_n$, analog zur Vereingungsautomatenkonstruktion im ungewichteten Fall. Dabei sei

$$Q_M := \dot{\bigcup_{i \in [1,n]}} Q_i \ ,$$

$$\mu_M(\sigma)_{q_1 \dots q_m, q} := \begin{cases} \mu_i(\sigma)_{q_1 \dots q_m, q} & \text{falls } q_1, \dots, q_m, q \in Q_i \ , \\ 0 & \text{sonst,} \end{cases}$$

$$\gamma_M(q) := \begin{cases} \gamma_i(q) & \text{falls } q \in Q_i \text{ und} \\ 0 & \text{sonst.} \end{cases}$$

Da $Q_1, \dots, Q_n$ paarweise disjunkt sind, ist M ebenfalls crisp-deterministisch. Sei $t \in T_\Sigma$. Weil T_Σ durch $L_1, \dots, L_n$ partitioniert wird, gibt es genau ein $i \in [1, n]$ mit $t \in L_i$. Also gilt $(S_{M_j}, t) = 0$ für alle $j \neq i$ und es gibt genau einen erfolgreichen Lauf $r \in R_M(t)$, das heißt $\gamma(r(\varepsilon)) \neq 0$, mit $\mathrm{wt}_M(t, r) \neq 0$. Dies ist genau derjenige, welcher durch den wta M_i definiert ist. Deshalb gilt

$$(S_M, t) = (S_{M_i}, t) = \mathbb{1}_{L_i} \cdot k_i = (S, t) \ .$$

($\Leftarrow$) Sei $M = (Q, \Sigma, \mu, \gamma)$ crisp-deterministisch mit $S = S_M$. Für alle Bäume $t \in T_\Sigma$ gilt damit $(S_M, t) \neq 0 \implies \exists! r \in R_M : \mathrm{wt}_M(t, r) = 1$. Sei $\mathrm{img}(\gamma) \setminus \{0\} = \{k_1, \dots, k_n\}$. Wir bilden nun die Baumautomaten $A_i := (Q, \Sigma, \delta, F_i)$ mit den Transitionsfunktionen

$$\delta_\sigma(q_1, \dots, q_m) = q \iff \mu(\sigma)_{q_1 \dots q_m, q} = 1 \ .$$

Diese sind wohldefiniert, weil M crisp-deterministisch ist. Die Finalzustände bestimmen wir mit

$$F_i := \{q \in Q \mid \gamma(q) = k_i\} \ .$$

Sei $L_i = L(A_i)$. Damit gilt nun

$$t \in L_i \iff (S_M, t) = k_i \text{ , denn } k_i \neq 0 \ .$$

Da die erkennbaren Baumsprachen unter Vereinigung und Komplementbildung abgeschlossen sind, ist die Sprache $L_{n+1} := T_\Sigma \setminus \bigcup_{i=1}^{n} L_i$ erkennbar. Offensichtlich partitionieren die Baumsprachen $L_1, \ldots, L_n$ und L_{n+1} die Menge aller Bäume T_Σ. Weil M crisp-deterministisch ist, erhalten wir $S = S_M = \sum_{i=1}^{n+1} k_i \cdot \mathbb{1}_{L_i}$. Somit ist S eine erkennbare Stufenfunktion. □

Als Nächstes betrachten wir eine weitere Einschränkung der gewichteten Baumautomaten. Ein *schwacher* gewichteter Baumautomat (weak weighted tree automata, kurz wwta) ist ein wta $M = (Q, \Sigma, \mu, \gamma)$ mit einer eingeschränkten Finalgewichtsfunktion $\gamma : Q \to \{0, 1\} \subseteq K$. Man kann diese Automaten auch vollkommen äquivalent mit ausgezeichneten Finalzuständen anstatt einer Gewichtsfunktion der Finalzustände definieren. Die von schwachen gewichteten Baumautomaten erkannten Baumreihen nennen wir *schwach en-erkennbar* (weakly recognizeable). Ist $en = TS$ beziehungsweise $en = en^<$, so sprechen wir von *schwach TS-erkennbar* beziehungsweise *schwach <-erkennbar*. Es stellt sich heraus, dass jede erkennbare Stufenfunktion mit allen Positionsaufzählungen *en* schwach *en*-erkennbar ist.

Fortan werden wir die nachfolgenden Schreibweisen benutzen. Die Projektion eines n-Tupels $x = (x_1, \ldots, x_n)$ auf seine i-te Komponente bezeichnen wir mit $\pi_i(x) := x_i$, wobei $i \in [1, n]$. Seien $g : A \to B$ und $f : B \to C$ Funktionen, so schreiben wir $f \circ g$ für die Verkettung dieser Funktionen, wobei $(f \circ g)(a) := f(g(a))$ für alle $a \in A$.

Lemma 3.2.10. *Sei $S \in K\langle\langle T_\Sigma \rangle\rangle$ eine erkennbare Stufenfunktion. Dann ist S mit jeder Positionsaufzählung für Gewichtsbäume en schwach en-erkennbar.*

Beweis. Sei $S \in K\langle\langle T_\Sigma \rangle\rangle$ eine erkennbare Stufenfunktion. Dann ist $S = \sum_{i=1}^{n} k_i \cdot \mathbb{1}_{L_i}$ für eine Partition von erkennbaren Baumsprachen $L_1, \ldots, L_n$ und eine Menge von Elementen $\{k_1, \ldots, k_n\} \subseteq K$. Es gibt also deterministische Baumautomaten $A_i = (Q_i, \Sigma, \delta_i, F_i)$ mit $L(A_i) = L_i$ für $i \in [1, n]$.

Aus diesen konstruieren wir nun gewichtete Baumautomaten $M_i := (Q_i', \Sigma, \mu_i, \gamma_i)$ mit binären Finalgewichten. Die Idee ist, dass ein wwta M_i beim Übergang in einen Finalzustand an der Wurzel mit k_i multipliziert. Er rät hierfür nichtdeterministisch, ob eine Position die Wurzel ist. Falls nicht, wird das Gewicht aller übergeordneten Positionen auf 0 gesetzt. Die Zustände Q_i' tragen dabei ein Flag, welches signalisiert, ob forthin mit 0 multipliziert werden muss beziehungsweise ob bereits mit k_i multipliziert worden ist.

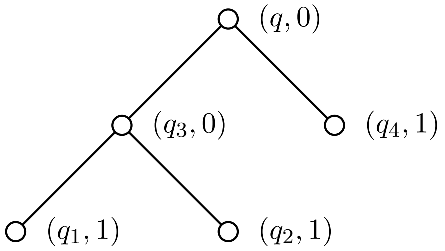
Es sei also

$$Q_i' := Q_i \times \{0,1\} \ ,$$

$$\mu_i(\sigma)_{(q_1,b_1)\dots(q_m,b_m),(q,b)} = \begin{cases} k_i & \text{falls } \delta_{i\sigma}(q_1,\dots,q_m) = q \text{ und} \\ & b_1 = \dots = b_m = 1 \text{ und } b = 0 \text{ und} \\ & q \in F_i \ , \\ 1 & \text{falls } \delta_{i\sigma}(q_1,\dots,q_m) = q \text{ und} \\ & b_1 = \dots = b_m = 1 \text{ und } b = 1 \ , \\ 0 & \text{sonst,} \end{cases}$$

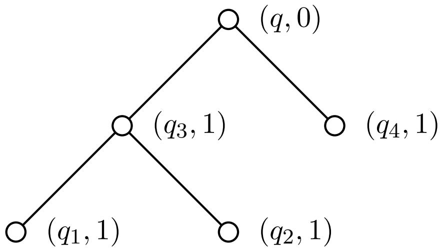
$$\gamma_i((q,b)) = \begin{cases} 1 & \text{falls } q \in F_i \text{ und } b = 0 \ , \\ 0 & \text{sonst.} \end{cases}$$

Abbildung 3.5, Abbildung 3.6 und Abbildung 3.7 illustrieren beispielhaft drei mögliche Läufe eines solchen Baumautomaten (mit Tiefensuche) auf einem Baum unter der Annahme, dass $F_i = \{q, q_3\}$.



$$\mathrm{wt}(t,r_1) \cdot \gamma((q,0)) = (1 \cdot 1 \cdot k_i \cdot 1 \cdot 0) \cdot 1$$

Abbildung 3.5: Lauf r_1



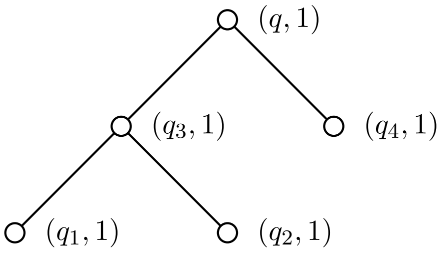
$$\mathrm{wt}(t,r_2) \cdot \gamma((q,0)) = (1 \cdot 1 \cdot 1 \cdot 1 \cdot k_i) \cdot 1$$

Abbildung 3.6: Lauf r_2

Für die wwta M_i und alle $t \in T_\Sigma$ gelten nun

1. $t \in L_i \implies (M_i, t) = k_i$ und
2. $t \notin L_i \implies (M_i, t) = 0$.

(1.) Sei $t \in L(A_i) = L_i$. Da A_i deterministisch ist, gibt es genau einen (erfolgreichen) Lauf $r \in R_{A_i}(t)$. Also muss $\pi_1 \circ r' = r$ für alle Läufe $r' \in R_{M_i}(t)$ mit $\mathrm{wt}^{en}_{M_i}(t,r') \neq 0$



$\mathrm{wt}(t, r_3) \cdot \gamma((q,1)) = (1 \cdot 1 \cdot 1 \cdot 1 \cdot 1) \cdot 0$

Abbildung 3.7: Lauf r_3

gelten. Seien $r_0, r_1 \in R_{M_i}(t)$ bestimmt durch

$$r_0(w) := \begin{cases} (r(w), 1) & \text{falls } w \neq \varepsilon \,, \\ (r(w), 0) & \text{sonst und} \end{cases}$$

$$r_1(w) := (r(w), 1) \,.$$

Für diese beiden Läufe gilt offenbar $\mathrm{wt}^{en}_{M_i}(t, r_0) = k_i$ beziehungsweise $\mathrm{wt}^{en}_{M_i}(t, r_1) = 1$. Angenommen ein weiterer Lauf $r' \in R_{M_i}$ mit $\mathrm{wt}^{en}_{M_i}(t, r') \neq 0$ und $r_0 \neq r' \neq r_1$ würde existieren. Dann gäbe es auch eine Position $w.l \in \mathrm{pos}(t)$ mit $1 \leq l \leq m = rk_\Sigma(t(w))$ derart, dass $r'(w.l) = (r(w.l), 0)$. Aber wegen $\mu_i(t(w))_{r'(w.1)\ldots r'(w.m), r'(w)} = 0$ erhalten wir $\mathrm{wt}^{en}_{M_i}(t, r') = 0$. Also sind r_0 und r_1 die einzigen Läufe von M_i auf t mit einem Gewicht verschieden von 0. Daher folgt

$$\begin{aligned} (S^{en}_{M_i}, t) &= \sum_{r \in R_{M_i}(t)} wt^{en}_{M_i}(t, r) \cdot \gamma_i(r(\varepsilon)) \\ &= \mathrm{wt}^{en}_{M_i}(t, r_0) \cdot \gamma_i(r_0(\varepsilon)) + \mathrm{wt}^{en}_{M_i}(t, r_1) \cdot \gamma_i(r_1(\varepsilon)) \\ &= (k_i \cdot 1) + (1 \cdot 0) \\ &= k_i \,. \end{aligned}$$

(2.) Da A_i deterministisch ist, gibt es genau einen Lauf $r \in R_{A_i}(t)$. Aus der Definition von μ folgt, dass $\mathrm{wt}^{en}_{M_i}(t, r') = 0$ für alle Läufe $r' \in R_{M_i}(t)$ mit $\pi_1 \circ r' \neq r$. Da r nicht erfolgreich ist, gilt $r(\varepsilon) \notin F$ und es folgt $\gamma(r'(\varepsilon)) = 0$ für alle Läufe $r' \in R_{M_i}(t)$ mit $\pi_1 \circ r' = r$. Also ist $(S^{en}_{M_i}, t) = 0$.

Abschießend bilden wir einen Vereinigungsautomaten M aus den wta $M_1, \ldots, M_n$. Wegen 1. und 2. gilt für diesen $S^{en}_M = S$. □

Für eine Positionsaufzählung für Gewichtsbäume *en* wollen wir die Klasse der formalen Potenzreihen, die schwach *en*-erkennbar sind, mit $K^{en-wrec}\langle\langle T_\Sigma \rangle\rangle$ bezeichnen. Die Menge der erkennbaren Stufenfunktionen ist für alle Positionsaufzählungen *en* in $K^{en-wrec}\langle\langle T_\Sigma \rangle\rangle$ *echt enthalten*, wie das folgende Beispiel zeigt.

Beispiel 3.2.11. Sei $M := (Q, \Sigma, \mu, \gamma)$ ein wwta mit $Q := \{p_1, p_2\}$, $\gamma(p_1) := 1$ und $\gamma(p_2) := 0$. Für alle $\sigma \in \Sigma$, $m = rk_\Sigma(\sigma)$, $q_1, \ldots, q_m, q \in Q$ sei μ wie folgt definiert:

$$\mu(\sigma)_{q_1 \ldots q_m, q} := \begin{cases} 1 & \text{falls } \sigma \in \Sigma^{(0)} \text{ oder} \\ & q_1 = \ldots = q_m = q = p_2 \text{ oder} \\ & q = p_1 \text{ und } \exists! j : 1 \leq j \leq m \wedge q_j = p_1 \,, \\ 0 & \text{sonst.} \end{cases}$$

Wir betrachten einen Lauf von M auf einem Baum $t \in T_\Sigma$ mit einem Gewicht verschieden von 1 und einem Finalgewicht von ebenfalls 1. An der Wurzel muss der Zustand $r(\varepsilon) = p_1$ sein. An allen Positionen $w \in \mathrm{pos}(t)$ mit $r(w) = p_1$ gilt die Invariante, dass genau ein Nachfolgeknoten ebenfalls mit p_1 beschriftet ist oder w ein Blatt ist. Also zeichnet r genau einen Pfad von der Wurzel zu einem Blatt mit p_1 aus. Im Semiring der natürlichen Zahlen $(\mathbb{N}_0, \cdot, +, 1, 0)$ zählt M somit die Anzahl der Blätter jedes Baumes $t \in T_\Sigma$. Folglich ist $\mathrm{img}(S_M^{en})$ für alle Positionsaufzählungen en nicht endlich, weswegen S_M^{en} keine erkennbare Stufenfunktion sein kann.

3.2.3 Erkennbarkeit und schwache Erkennbarkeit von Baumreihen

Bei bestimmten Positionsaufzählungen für Gewichtsbäume en fallen Erkennbarkeit und schwache Erkennbarkeit zusammen. Das ist dann der Fall, wenn in en die Wurzel die letzte Position jedes Gewichtsbaumes ist.

Lemma 3.2.12. *Sei en eine Positionsaufzählung für Gewichtsbäume derart, dass für alle $t \in T_K$ stets $en(t)(|\mathrm{pos}(t)|) = \varepsilon$ gilt. Dann ist S genau dann schwach en-erkennbar, wenn $S \in K\langle\langle T_\Sigma\rangle\rangle$ en-erkennbar ist.*

Beweis. Sei $M = (Q, \Sigma, \mu, \gamma)$ ein gewichteter Baumautomat, der S en-erkennt. Wir konstruieren nun einen wta $M' := (Q', \Sigma, \mu', \gamma')$, der S schwach en-erkennt. Analog zur Beweisidee von Lemma 3.2.10 rät M', ob die aktuelle Position die Wurzel ist und multipliziert ggf. mit dem entsprechenden Finalgewicht aus M. Ist die Position nicht die Wurzel, so verwirft M' diesen Lauf, indem an allen übergeordneten Positionen das Gewicht auf 0 gesetzt wird. Wir definieren M' nun folgendermaßen:

$$Q' := Q \times \{0, 1\} \ ,$$

$$\mu'(\sigma)_{(q_1,b_1)\dots(q_m,b_m),(q,b)} := \begin{cases} \mu(\sigma)_{q_1\dots q_m,q} & \text{falls } b_1 = \ldots = b_m = 1 \text{ und } b = 1 \ , \\ \mu(\sigma)_{q_1\dots q_m,q} \cdot \gamma(q) & \text{falls } b_1 = \ldots = b_m = 1 \text{ und } b = 0 \ , \\ 0 & \text{sonst,} \end{cases}$$

$$\gamma'((q,b)) = \begin{cases} 1 & \text{falls } b = 0 \text{ und} \\ 0 & \text{sonst.} \end{cases}$$

Wir zeigen nun, dass für alle $t \in T_\Sigma$ die Gleichung $(S_M^{en}, t) = (S_{M'}^{en}, t)$ gilt.

Für alle Läufe $r' \in R_{M'}(t)$ mit $\mathrm{wt}_{M'}^{en}(t, r') \neq 0$ muss $w \neq \varepsilon \implies \pi_2(r'(w)) = 1$ gelten. Sei zunächst $r \in R_M(t)$ ein beliebiger Lauf. Von allen Läufen $r' \in R_{M'}(t)$ mit $r = \pi_1 \circ r'$ können nur Läufe der Form $r_0, r_1 \in R_{M'}(t)$ mit

$$r_0(w) := \begin{cases} (r(w), 1) & \text{falls } w \neq \varepsilon \ , \\ (r(w), 0) & \text{falls } w = \varepsilon \text{ und} \end{cases}$$

$$r_1(w) := (r(w), 1) \ .$$

einen Wert verschieden von 0 erhalten. Nach Definition von M' erhalten wir für jedes

$r \in R_M(t)$:

$$\begin{aligned}
\mathrm{wt}^{en}_M(t,r) \cdot \gamma(r(\varepsilon)) &= \mathrm{wt}^{en}_{M'}(t,r_0) \\
&= \mathrm{wt}^{en}_{M'}(t,r_0) \cdot \gamma'(r_0(\varepsilon)) + \mathrm{wt}^{en}_{M'}(t,r_1) \cdot \gamma'(r_1(\varepsilon)) \\
&= \sum_{\substack{r' \in R_{M'}(t) \\ r = \pi_1 \circ r'}} \mathrm{wt}^{en}_{M'}(t,r') \cdot \gamma'(r'(\varepsilon)) \ .
\end{aligned}$$

Sei andererseits $r' \in R_{M'}(t)$ ein erfolgreicher Lauf mit $\mathrm{wt}^{en}_{M'}(t,r') \neq 0$. Dann ist $r := \pi_1 \circ r'$ ein Lauf von M auf t mit $\mathrm{wt}^{en}_M(t,r) \neq 0$ und r' muss von der Form r_0 sein. Daraus folgt

$$\begin{aligned}
(S^{en}_M, t) &= \sum_{r \in R_M(t)} \mathrm{wt}^{en}_M(t,r) \cdot \gamma(r(\varepsilon)) \\
&= \sum_{r \in R_M(t)} \mathrm{wt}^{en}_{M'}(t,r_0) \quad \text{(wobei } r_0 \text{ jeweils wie oben)} \\
&= \sum_{r' \in R_{M'}(t)} \mathrm{wt}^{en}_M(t,r') \cdot \gamma'(r'(\varepsilon)) \\
&= (S^{en}_{M'}, t) \ .
\end{aligned}$$

□

Mit diesem Lemma folgt für die Positionsaufzählung TS sofort das folgende Korollar, denn für TS gilt stets $TS(|\mathrm{pos}(t)|) = \varepsilon$ für alle $t \in T_K$.

Korollar 3.2.13. *Sei K ein starkes Bimonoid. Es gilt:*

$$K^{TS-wrec}\langle\!\langle T_\Sigma \rangle\!\rangle = K^{TS-rec}\langle\!\langle T_\Sigma \rangle\!\rangle \ .$$

Wir betrachten nun das starke Bimonoid B_ω aus Beispiel 3.1.1.

Sei $S : T_\Sigma \to B_\omega$ die durch $(S,t) = 1^{|\mathrm{pos}(t)|}0$ bestimmte formale Potenzreihe. S ist TS-erkennbar, zum Beispiel durch den wta $M := (\{q\}, \Sigma, \mu, \gamma)$ mit $\mu(\sigma)_{q^m,q} := 1$ und $\gamma(q) := 0$ für $\sigma \in \Sigma^{(m)}$. Also ist S auch schwach TS-erkennbar.

Sei nun $<$ eine beliebige lineare Ordnung auf B_ω derart, dass $w0 < v1$ für alle Worte $w0, v1 \in B_\omega$ gilt. Wir können eine solche Ordnung beispielsweise aus jeder linearen Ordnung $\prec$ auf $\{0,1\}^*$ konstruieren. Sei dazu $v, w \in B_\omega$, dann definiere

$$\begin{aligned}
v < w \iff\ & v = \varepsilon \vee w = 1^\omega \vee \exists v', w' \in \{0,1\}^* \, \exists x, y \in \{0,1\} : \\
& (v = v'x \wedge w = w'y \wedge ((x = 0 \wedge y = 1) \vee (x = y \wedge v' \prec w'))) \ .
\end{aligned}$$

S ist $<$-erkennbar, unter anderem durch den obigen Automaten M. Jedoch ist S nicht schwach $<$-erkennbar. Angenommen es gäbe einen schwachen wta $M' = (Q', \Sigma, \mu', \gamma')$ mit $S = S^{<}_{M'}$. Sei $n := \max(\{|w| \mid w \in \mathrm{img}(\mu') \setminus \{1^\omega\}\})$ die Länge des längsten Gewichtes aus $\mathrm{img}(\mu')$. Sei weiterhin $t \in T_\Sigma$ ein Baum mit $|\mathrm{pos}(t)| > n$. Es gibt nun mindestens einen erfolgreichen Lauf $r \in R_{M'}(t)$ mit $\mathrm{wt}^{en<}_{M'}(t,r) \in \{0,1\}^* \, 0$. Wegen $<$ müssen alle Faktoren in $\mathrm{wt}^{en<}_{M'}(t,r)$ die Form $w0 \in \mathrm{img}(\mu')$ haben. Da aber $|\mathrm{pos}(t)| > n$ gilt, hat $\mathrm{wt}^{en<}_{M'}(t,r)$ tatsächlich die Form $\{0,1\}^* \, 0 \, \{0,1\}^* \, 0$. Also ist $(S^{<}_{M'}, t) \neq 1^{|\mathrm{pos}(t)|}0$. (Widerspruch)

Es gibt also Bimonoide K mit linearen Ordnungen $<$, für die $<$-Erkennbarkeit nicht mit schwacher $<$-Erkennbarkeit zusammenfällt. Für solche linearen Ordnungen gilt dann $K^{<-wrec}\langle\langle T_\Sigma\rangle\rangle \subset K^{<-rec}\langle\langle T_\Sigma\rangle\rangle$.

Dennoch gibt es auch lineare Ordnungen auf K, sodass die formale Potenzreihe S $<$-erkennbar mit diesen Ordnungen ist. Sei beispielsweise $\prec$ eine beliebige feste lineare Ordnung auf B_ω derart, dass $w1 \prec w0$ für alle $w0, w1 \in B_\omega$ gilt. Die Baumreihe S ist schwach $\prec$-erkennbar. Das leistet der wta $M := (\{p,q\}, \Sigma, \mu, \gamma)$, wobei

$$\mu(\sigma)_{r_1\ldots r_m,r} := \begin{cases} 1 & \text{falls } r_1 = \ldots = r_m = p \text{ und } r = p\ , \\ 10 & \text{falls } r_1 = \ldots = r_m = p \text{ und } r = q\ , \\ 1^\omega & \text{sonst,} \end{cases}$$

$$\gamma(r) := \begin{cases} \varepsilon & \text{falls } r = q \text{ und} \\ 1^\omega & \text{sonst.} \end{cases}$$

Wir haben gesehen, dass einige Baumreihen mit allen Positionsaufzählungen erkennbar sind. Wir werden jedoch als Nächstes zeigen, dass für zwei Positionsaufzählungen die Mengen der jeweils erkennbaren Baumreihen eine nichtleere symmetrische Differenz haben können.

Proposition 3.2.14. *Es gibt ein Bimonoid K mit einer linearen Ordnung $<$ und eine $<$-erkennbare Potenzreihe, die nicht TS-erkennbar ist.*

Beweis. Sei $\Sigma = (\{\sigma,\tau\}, rk_\Sigma)$ mit $rk_\Sigma(\sigma) = 0$ und $rk_\Sigma(\tau) := 1$. Wir betrachten das starke Bimonoid B_ω mit der obigen linearen Ordnung:

$$\begin{aligned} v < w \iff & v = \varepsilon \lor w = 1^\omega \lor \exists v', w' \in \{0,1\}^* \,\exists x, y \in \{0,1\} : \\ & (v = v'x \land w = w'y \land ((x = 0 \land y = 1) \lor (x = y \land v' \prec w')))\ . \end{aligned}$$

Dabei sei $\prec$ eine fest gewählte lineare Ordnung auf $\{0,1\}^*$. Wir definieren nun die Baumreihe S mit

$$(S,t) := \begin{cases} 0^n 1^n & \text{falls } \exists n \in \mathbb{N}_0 : |\mathrm{pos}(t)| = 2n \text{ und} \\ 0^{n+1} 1^n & \text{falls } \exists n \in \mathbb{N}_0 : |\mathrm{pos}(t)| = 2n+1\ . \end{cases}$$

Wir betrachten den wta $M := (\{p,q\}, \Sigma, \mu, \gamma)$ mit

$$\mu(\sigma)_{r_1,r} := \begin{cases} 0 & \text{falls } \sigma = \sigma \text{ und } r = p\ , \\ & \text{oder } \sigma = \tau, r_1 = q \text{ und } r = p\ , \\ 1 & \text{falls } \sigma = \tau, r_1 = p \text{ und } r = q\ , \\ 1^\omega & \text{sonst und} \end{cases}$$

$$\gamma(r) := \varepsilon \quad \text{für alle } r \in Q\ .$$

Sei nun $t \in T_\Sigma$. Der einzige Lauf $r \in R_M(t)$ mit $\mathrm{wt}_M^{en^<}(t,r) \neq 1^\omega$ ist bestimmt durch

$$r(w) := \begin{cases} p & \text{falls } w.1 \notin \mathrm{pos}(t) \text{ oder } r(w.1) = q \\ q & \text{falls } r(w.1) = p \end{cases}$$

Mit Beachtung von $<$ gilt offenbar $\mathrm{wt}_M^{en^<}(t,r) = (S,t)$, also $S_M^< = S$.

Es gibt jedoch keinen wta, der S (schwach) TS-erkennt. Wir verwenden nun ein Pumping-Argument um dies zu zeigen.

Angenommen S ist schwach TS-erkennbar, dann gäbe es einen wwta $M := (Q, \Sigma, \mu, \gamma)$ mit $S_M^{TS} = S$. Sei $m := |Q|$ und $k_{max} := \max(\{|k| \mid k \in \mathrm{img}(\mu) \setminus \{1^\omega\}\})$ die Länge des längsten Gewichtes von μ. Wir betrachten nun für $x := (m+1) \cdot k_{max}$ den Baum

$$s(w) := \begin{cases} \sigma & \text{falls } w = 1^{2x-1} & \text{(das einzige Blatt) und} \\ \tau & \text{falls } w \in \left\{\varepsilon, 1, \ldots, 1^{2x-2}\right\} & \text{(innere Knoten).} \end{cases}$$

Da s nur einen Pfad einhält, können wir uns s wie folgt veranschaulichen:

$$s := \quad \overbrace{\sigma - \tau - \ldots - \tau}^{x} - \overbrace{\tau - \ldots - \tau}^{x} \ .$$

Es existiert nun ein erfolgreicher Lauf $r_s \in R_M(s)$ mit einem Gewicht der Form $\{0,1\}^+ \cdot 0 \cdot 1^x$, denn $(S_M^{TS}, s) = 0^x 1^x$ entsteht mittels $\oplus$, das heißt stellenweiser Minimumbildung von rechts, aus dem Gewicht aller Läufe. Dieser Lauf r_s enthält an den ersten $m+1$ Positionen (vom Blatt an Position 1^{2x-1} beginnend) einen Zyklus. Das bedeutet, es gibt zwei Positionen $v, v.1^n \in \left\{1^{2x-m}, \ldots, 1^{2x-1}\right\}$ mit $n \in \mathbb{N}$ und $r(v) = r(v.1^n)$. Wir wollen diesen Zyklus mit (v,n) bezeichnen und sein Gewicht sei

$$\mathrm{wt}_{(v,n)} := M(t,r)(v.1^{n-1}) \cdot M(t,r)(v.1^{n-2}) \cdot \ldots \cdot M(t,r)(v) \ .$$

Es gilt $n \le m$ und mit $x = (m+1) \cdot k_{max}$ und deshalb für die Länge von $\mathrm{wt}_{(v,n)}$ auch $\left|\mathrm{wt}_{(v,n)}\right| < x$. Folglich ist $\mathrm{wt}_M^{TS}(s, r_s) \in \{0,1\}^* \cdot \mathrm{wt}_{(v,n)} \cdot \{0,1\}^* \cdot 1^x$, das heißt, das Gewicht des Zyklus hat keinen Einfluss auf die letzten x Stellen vom Gewicht des Laufes r_s. Nun pumpen wir den Zyklus und erhalten den Baum s':

$$s'(w) := \begin{cases} \sigma & \text{falls } w = 1^{2x-1+n} & \text{(das einzige Blatt) und} \\ \tau & \text{falls } w \in \left\{\varepsilon, 1, \ldots, 1^{2x-2+n}\right\} & \text{(innere Knoten).} \end{cases}$$

Wir können uns s' wieder wie folgt vorstellen:

$$s' := \quad \overbrace{\sigma - \tau - \ldots - \tau}^{x+n} - \overbrace{\tau - \ldots - \tau}^{x} \ .$$

Wir pumpen auch r_s entsprechend auf:

$$r_s'(w) := \begin{cases} r(w) & \text{falls } w \in \left\{\varepsilon, 1, \ldots, v.1^{n-1}\right\} \text{ und} \\ r(u) & \text{falls } w \in \left\{v.1^n, \ldots, 1^{2x-1+n}\right\} \text{ mit } w = u.1^n \ . \end{cases}$$

Damit sieht r_s' wie folgt aus:

$$r_s' := \quad \overbrace{\underset{r_s(1^{2x-1})}{\overset{1^{2x-1+n}}{\sigma}} - \ldots - \underset{r_s(v.1)}{\overset{v.1^{n+1}}{\tau}} - \underset{r_s(v)}{\overset{v.1^n}{\tau}} - \underset{r_s(v.1^{n-1})}{\overset{v.1^{n-1}}{\tau}} - \ldots}^{x+n} - \overbrace{\underset{r_s(1^{x-1})}{\overset{1^{x-1}}{\tau}} - \ldots - \underset{r_s(\varepsilon)}{\overset{\varepsilon}{\tau}}}^{x} \ .$$

Mit dieser Konstruktion ist auch r_s' ein erfolgreicher Lauf auf s'. Da der Zyklus (v,n) auf die letzten x Stellen des Gewichtes von r_s keinen Einfluss hat, gilt für r_s' ebenfalls $\mathrm{wt}_M^{TS}(r_s', s') \in \{0,1\}^+ \cdot 0 \cdot 1^x$. Damit ist aber auch $(S_M^{TS}, s') \in \{0,1\}^+ \cdot 0 \cdot 1^x$, doch $(S, s') = 0^{x+n} 1^{x+n}$. (Widerspruch) $\square$

Somit gilt $K^{<-rec}\langle\langle T_\Sigma\rangle\rangle \nsubseteq K^{TS-rec}\langle\langle T_\Sigma\rangle\rangle$ für einige Bimonoide K und lineare Ordnungen $<$. Es stellt sich nun heraus, dass die umgekehrte Inklusion ebenfalls nicht gilt.

Proposition 3.2.15. *Es gibt ein Bimonoid K und eine TS-erkennbare Potenzreihe, die mit keiner linearen Ordnung $<$ über K auch $<$-erkennbar ist.*

Beweis. Sei $\mathbb{N}^*_\wp$ das starke Bimonoid, welches wir in Beispiel 3.1.3 aus dem freien Monoid über $\mathbb{N}^*$ konstruiert haben und Σ ein Rangalphabet mit $max_\Sigma > 1$. Wir betrachten die formale Potenzreihe $S : T_\Sigma \to K$ mit

$$(S,t) = \overleftarrow{\mathrm{pos}}(t) := \{\varepsilon\} \cup \{n_m \dots n_1 \mid n_1, \dots, n_m \in \mathbb{N} \wedge n_1 \dots n_m \in \mathrm{pos}(t)\} \ .$$

Dann ist S TS-erkennbar, doch es gibt keine lineare Ordnung $<$ auf $\mathbb{N}^*_\wp$, sodass die Baumreihe S $<$-erkennbar ist. S wird beispielsweise durch den wta $M := (\{p,q\}, \Sigma, \mu, \gamma)$ TS-erkannt, wobei μ für jedes $\sigma \in \Sigma$ mit $m := rk_\Sigma(\sigma)$ bestimmt ist durch

$$\mu(\sigma)_{\vec{r},r} := \begin{cases} \{\varepsilon\} & \text{falls } \vec{r} = q^m \ , \\ \{1\} & \text{falls } r = p \text{ und } \vec{r} = pq^{m-1} \ , \\ \{2\} & \text{falls } r = p \text{ und } \vec{r} = qpq^{m-2} \ , \\ \vdots & \vdots \\ \{m\} & \text{falls } r = p \text{ und } \vec{r} = q^{m-1}p \ , \\ \emptyset & \text{sonst und} \end{cases}$$

$$\gamma(r) := \{\varepsilon\} \quad \text{für alle } r \in Q \ .$$

Wir stellen nun zunächst einige Beobachtungen auf dem freien Monoid über $\mathbb{N}^*$ an und erweitern diese im Anschluss auf $\mathbb{N}_\wp$. Sei $\pi \in \mathbb{N}^\omega$ aperiodisch, das heißt, es gibt keine Zerlegung von π, sodass $\pi = \pi_1\pi_2^\omega$ mit $\pi_1, \pi_2 \in \mathbb{N}^*$. Sei weiterhin $n_1, \dots, n_m$ eine beliebige Folge von Worten aus $\mathbb{N}^*$. Wir zeigen nun, dass die Menge

$$P_m := \left\{ n_1^{k_1} \dots n_m^{k_m} \;\middle|\; k_1, \dots, k_m \in \mathbb{N}_0 \right\}$$

nur endlich viele Präfixe von π enthält. Induktiv: Für $m = 1$ gilt die Aussage offensichtlich für alle aperiodischen $\pi \in \mathbb{N}^\omega$. Sei die Aussage nun für m bewiesen und $n_1, \dots, n_m, n_{m+1}$ eine Folge von Worten aus $\mathbb{N}^*$. Wir bezeichnen die endliche Menge $\{\alpha \in P_m \mid \exists\omega \in \mathbb{N}^\omega : \alpha\omega = \pi\}$ der Präfixe von π aus P_m mit A_m. Dann ist die Menge der zugehörigen Suffixe $\Omega_m := \{\omega \in \mathbb{N}^\omega \mid \exists\alpha \in A_m : \alpha\omega = \pi\}$ ebenfalls endlich. Für jeden dieser Suffixe enthält $B_m := \left\{ n_{m+1}^k \;\middle|\; k \in \mathbb{N}_0 \right\}$ nur endlich viel Präfixe. Also ist

$$\begin{aligned} A_{m+1} &:= \{\alpha\beta \mid \alpha \in A_m \wedge \beta \in B_m \wedge \exists\omega \in \mathbb{N}^* : \beta\omega \in \Omega_M\} \\ &= \{\alpha \in P_{m+1} \mid \exists\omega \in \mathbb{N}^* : \alpha\omega = \pi\} \end{aligned}$$

endlich. Da A_{m+1} die Menge der Präfixe von π ist, die in P_{m+1} enthalten sind, folgt die Behauptung für $m+1$. Nachfolgend sagen wir auch, die Folge $n_1, \dots, n_m$ *bildet (endlich viele) Präfixe von π.*

Angenommen es gäbe eine Ordnung $<$ auf $\mathbb{N}^*_\wp$ und einen Automaten $M = (Q, \Sigma, \mu, \gamma)$, welcher S mit $<$ erkennt. Sei $N_1 < \dots < N_m$ die Folge aller Gewichte aus $\mathrm{img}(\mu)$ und

$N := \bigcup \mathrm{img}(\gamma)$. Dann gibt es höchstens $l := |N_1| \cdot \ldots \cdot |N_m| \cdot |N|$ viele verschiedene Folgen $w_i := n_1, \ldots, n_m, n$ mit $i \in [1, l]$, $n_j \in N_j$ für $j \in [1, m]$ und $n \in N$. Sei nun $\pi \in [1, max_\Sigma]^\omega$ aperiodisch. Dann bildet jede der Folgen w_i nur endliche viele Präfixe von π. Sei α der längste dieser Präfixe von π. Dann bildet keine der Folgen w_i Präfixe von π der Länge $|\alpha| + 1$. Folglich können Präfixe von π dieser Länge nicht in einem Gewicht $W \in \mathrm{img}(S_M)$ enthalten sein. Es gibt jedoch Bäume $t \in T_\Sigma$, für die $(S, t) = \overleftarrow{\mathrm{pos}}(t)$ einen solchen Präfix enthält. (Widerspruch) □

Somit wäre $K^{TS-rec}\langle\langle T_\Sigma \rangle\rangle \nsubseteq K^{<-rec}\langle\langle T_\Sigma \rangle\rangle$ für einige Bimonoide K und lineare Ordnungen $<$ gezeigt. Da erkennbare Stufenfunktionen jedoch mit jeder Positionsaufzählung erkennbar sind, ist der Schnitt dieser Mengen nicht leer. Folglich hängt die Erkennbarkeit einer formalen Potenzreihe entscheidend von der Wahl der Positionsaufzählung für die Gewichtsbäume ab.

Die folgende Aussage wird uns den Weg zum Hauptergebnis dieses Abschnitts ebnen.

Lemma 3.2.16. *Sei en eine Positionsaufzählung für Gewichtsbäume und $S \in \mathbb{N}_0\langle\langle T_\Sigma \rangle\rangle$ en-erkennbar. Dann ist für alle $c, n \in \mathbb{N}_0$ mit $c > 0$*

i) $S^{-1}(n) \in T_\Sigma$ *erkennbar und*

ii) $S^{-1}(n + c\mathbb{N}_0) \in T_\Sigma$ *erkennbar.*

Beweis. Sei $S \in K\langle\langle T_\Sigma \rangle\rangle$ *en*-erkennbar, dann gibt es einen wta M mit $S_M^{en} = S$. Da $\mathbb{N}_0$ kommutativ ist, spielt die Reihenfolge der Gewichte bei der Bestimmung des Gewichtes eines Laufs nach Proposition 3.2.8 keine Rolle. Also ist S von M ebenfalls TS-erkennbar.

(i) Wir definieren den Semiring $\mathbb{N}_{n+1} := ([0, n+1], \oplus, \odot, 0, 1)$. Für $a, b \in [0, n+1]$ seien

$$\begin{aligned} a \oplus b &:= \min(a + b, n + 1) \text{ und} \\ a \odot b &:= \min(a \cdot b, n + 1) \ . \end{aligned}$$

Dieser entspricht einer Beschränkung des Semirings $(\mathbb{N}_0, +, \cdot, 0, 1)$ auf die ersten $n + 2$ Elemente. Sei nun $h : \mathbb{N}_0 \to \mathbb{N}_{n+1}$ ein Homomorphismus mit $h(a) := \min(a, n+1)$ für alle $a \in \mathbb{N}_0$. Nach [5], [20, Theorem 3.9] wird Erkennbarkeit durch Semiringhomomorphismen erhalten, also ist $h(S)$ erkennbar.

In lokal endlichen Semiringen ist nach [6], [20, Theorem 3.15] das Urbild jeder Teilmenge unter einer erkennbaren formalen Potenzreihe ebenfalls erkennbar. Da $\mathbb{N}_{n+1}$ endlich ist, ist somit $h(S)^{-1}(n)$ erkennbar. Nun gilt aber

$$\begin{aligned} h(S)^{-1}(n) &= \{t \in T_\Sigma \mid h(S(t)) = n\} \\ &= \{t \in T_\Sigma \mid S(t) = n\} \\ &= S^{-1}(n) \ . \end{aligned}$$

Deshalb ist auch $S^{-1}(n)$ erkennbar.

(ii) Analog dazu betrachten wir den endlichen Semiring der Restklassen modulo c; $\mathbb{N}_c := (\mathbb{N}_{/c\mathbb{N}}, \oplus, \odot, 0, 1)$. Für $a, b \in [0, n+1]$ sind $\oplus$ und $\odot$ die üblichen Erweiterungen von $+$ und $\cdot$ auf die Restklassen:

$$\begin{aligned} a \oplus b &= a + b \pmod{c} \quad \text{und} \\ a \odot b &= a \cdot b \pmod{c} \quad . \end{aligned}$$

Sei nun $h : \mathbb{N}_0 \to \mathbb{N}_{/c\mathbb{N}}$ der natürliche Homomorphismus in die Menge der Restklassen mit $h(a) := [a]_c$. Es gilt wiederum

$$\begin{aligned} h(S)^{-1}([n]) &= \{t \in T_\Sigma \mid h(S(t)) = [n]\} \\ &= \{t \in T_\Sigma \mid S(t) = n \pmod{c}\} \\ &= S^{-1}(n + c\mathbb{N}) \, . \end{aligned}$$

Also ist auch $S^{-1}(n + c\mathbb{N}_0)$ erkennbar. □

Diesen Beweis kann man auch führen, indem man Baumautomaten, die $S^{-1}(n)$ beziehungsweise $S^{-1}(n + c\mathbb{N}_0)$ erkennen, direkt konstruiert. Im ersten Fall kann man analog zur Potenzmengenkonstruktion mithilfe von Multimengen die Anzahl der Läufe, mit einem Gewicht aus $[1, n]$, in den Zuständen bis maximal n zählen. Im zweiten Fall zeigt man zunächst, dass $S^{-1}(\{x \mid x = n \pmod{c}\})$ erkennbar ist. Dazu wandelt man die Konstruktion des ersten Falls derart um, dass die Transitionsfunktionen modulo c rechnen. Dann folgt die Aussage sofort aus dem ersten Fall.

Im Folgenden wird für alle $m \in \mathbb{N}$ und $k \in K$ der Term $\sum_{i=1}^{m} k$ mit mk abgekürzt und mit k^m der Term $\prod_{i=1}^{m} k$. Ist $m = 0$, so sei $mk := 0$ und $k^m := 1$.

Sei $(K, +, \cdot, 0, 1)$ ein bi-lokal endliches Bimonoid. Für jedes $k \in K$ ist das *additive Untermonoid* $\langle k \rangle^+$, welches durch $\{k\}$ induzierte Untermonoid von $(K, +, 0)$. Analog dazu heißt $\langle k \rangle^*$ *multiplikatives Untermonoid*. Sowohl $\langle k \rangle^+$ wie auch $\langle k \rangle^*$ sind endlich, da K bi-lokal endlich ist.

Wir können nun $\langle k \rangle^+$ als endliche Menge von Vielfachen von k schreiben, denn die Folge aller Vielfachen von k muss aufgrund der Endlichkeit von $\langle k \rangle^+$ schließlich in eine Schleife laufen. Um die kürzeste dieser Schleifen zu bestimmen, wählen wir das kleinste $m_k^+ \in \mathbb{N}_0$ mit $\exists y \in \mathbb{N} : m_k^+ k = (m_k^+ + y)k$ aus. Nun sei c_k^+ das Minimum dieser y und $d_k^+ := m_k^+ + c_k^+ - 1$. Damit lässt sich $\langle k \rangle^+$ als $\{0, k, 2k, \dots, d_k^+ k\}$ schreiben.

$$\begin{aligned} \langle k \rangle^+ &= \{0 \, , \, k \, , \, 2k \, , \, \dots \, , \, \overbrace{m_k^+ k \, , \, \dots \, , \, d_k^+ k \, , \, (m_k^+ + c_k^+)k}^{\text{kürzeste Schleife mit } m_k^+ k = (m_k^+ + c_k^+)k} \, , \, \dots\} \\ &= \{0 \, , \, k \, , \, 2k \, , \, \dots \, , \, m_k^+ k \, , \, \dots \, , \, d_k^+ k\} \end{aligned}$$

Das multiplikative Untermonoid $\langle k \rangle^*$ können wir analog zu $\langle k \rangle^+$ als endliche Menge von Potenzen von k, sogenannte Repräsentanten, schreiben. Wir wählen wieder das kleinste $m_k \in \mathbb{N}_0$ mit $\exists y \in \mathbb{N} : k^{m_k} = k^{m_k + y}$ aus. Nun sei c_k das Minimum dieser y und $d_k := m_k + c_k - 1$. Damit lässt sich $\langle k \rangle^*$ als $\{1, k, k^2, \dots, k^{d_k}\}$ schreiben.

$$\begin{aligned} \langle k \rangle^* &= \{1 \, , \, k \, , \, k^2 \, , \, \dots \, , \, \overbrace{k^{m_k} \, , \, \dots \, , \, k^{d_k} \, , \, k^{m_k + c_k}}^{\text{kürzeste Schleife mit } m_k k = k^{m_k + c_k}} \, , \, \dots\} \\ &= \{1 \, , \, k \, , \, k^2 \, , \, \dots \, , \, k^{m_k} \, , \, \dots \, , \, k^{d_k}\} \end{aligned}$$

Wir definieren für $\langle k \rangle^*$ nun eine Funktion $rep_k : \mathbb{N}_0 \to [0, d_k]$, die uns für jede Potenz von k ihren Repräsentanten liefert. Diese ist durch $rep_k(n) = n' \in [0, d_k] \iff k^n = k^{n'}$ definiert.

Satz 3.2.17. *Sei K bi-lokal endlich und $<$ eine lineare Ordnung auf K. Sei weiterhin $S \in K\langle\langle T_\Sigma \rangle\rangle$. Ist S TS-erkennbar oder $<$-erkennbar, so ist S eine erkennbare Stufenfunktion.*

Beweis. Sei S eine TS-erkennbare oder $<$-erkennbare Baumreihe und $M := (Q, \Sigma, \mu, \gamma)$ ein gewichteter Baumautomat mit $S_M^{TS} = S$ beziehungsweise $S_M^< = S$. Außerdem sei $B := \{\mathrm{img}(\mu(\sigma)) \mid \sigma \in \Sigma\} \cup \mathrm{img}(\gamma)$ und der Produktabschluss von B ist die Menge $Y := \{b_1 \cdot \ldots \cdot b_n \mid n \in \mathbb{N} \wedge b_1, \ldots, b_n \in B\}$. Da K multiplikativ lokal endlich ist, ist auch Y eine endliche Menge. Für alle Bäume $t \in T_\Sigma$, alle Positionsaufzählungen en und alle Läufe $r \in R_M(t)$ gilt $\mathrm{wt}_M^{en}(t,r) \in Y$.

Für jedes $\alpha \in Y$ definieren wir nun auf dem Semiring der natürlichen Zahlen $\mathbb{N}_0$ einen crisp wta, welcher für jeden Baum $t \in T_\Sigma$ die Anzahl der Läufe $r \in R_M(t)$ mit $\mathrm{wt}_M^{TS}(t,r) \cdot \gamma(r(\varepsilon)) = \alpha$ beziehungsweise $\mathrm{wt}_M^{en^<}(t,r) \cdot \gamma(r(\varepsilon)) = \alpha$ zählt. Wird S durch M TS-erkannt, so sei $M_\alpha^{TS} := (Q^{TS}, \Sigma, \mu^{TS}, \gamma_\alpha^{TS})$ mit

$$Q^{TS} := Q \times Y \ ,$$

$$\mu^{TS}(\sigma)_{(q_1,k_1)\ldots(q_m,k_m),(q,k)} := \begin{cases} 1 & \text{falls } k = (\prod_{i=1}^m k_i) \cdot \mu(\sigma)_{q_1\ldots q_m,q} \ , \\ 0 & \text{sonst,} \end{cases}$$

$$\gamma_\alpha^{TS}((q,k)) := \begin{cases} 1 & \text{falls } k \cdot \gamma(q) = \alpha \text{ und} \\ 0 & \text{sonst.} \end{cases}$$

Da μ^{TS} die Multiplikation der Gewichte genau in der Reihenfolge der Positionsaufzählung TS ausführt, gilt für jeden Baum $t \in T_\Sigma$ und alle Läufe $r \in R_{M_\alpha^{TS}}(t)$ die Gleichung $\pi_2(r(\varepsilon)) = \mathrm{wt}_M^{TS}(t, \pi_1 \circ r)$. Somit gilt für die von M erkannte Baumreihe und alle $t \in T_\Sigma$

$$(S_{M_\alpha^{TS}}^{TS}, t) = \left|\left\{r \in R_M(t) \;\middle|\; \mathrm{wt}_M^{TS}(t,r) \cdot \gamma(r(\varepsilon)) = \alpha\right\}\right| \ .$$

Wird S durch M hingegen $<$-erkannt, so zählen wir die Anzahl der Vorkommen aller Faktoren in den Zuständen mithilfe beschränkter Multimengen. Da für alle $k \in Y$ das multiplikative Untermonoid $\langle k \rangle^*$ endlich ist, nutzen wir dazu die rep_k-Funktion. Wir definieren dazu den crisp wta $M_\alpha^< := (Q^<, \Sigma, \mu^<, \gamma_\alpha^<)$ mit der Zustandsmenge

$$Q^< := Q \times [0,z]^Y \quad \text{wobei} \quad z := \max\left(\{rep_k(n) \mid k \in Y \wedge n \in \mathbb{N}_0\}\right) \ .$$

Für $Y = \{k_1, \ldots, k_n\}$ mit $n = |Y|$ und $k_1 < \ldots < k_n$ sei die Funktion der Finalgewichte

$$\gamma_\alpha^<((q,f)) := \begin{cases} 1 & \text{falls } k_1^{f(k_1)} \cdot \ldots \cdot k_n^{f(k_n)} = \alpha \ , \\ 0 & \text{sonst} \end{cases}$$

und die Transitionsfunktion

$$\mu^<(\sigma)_{(q_1,f_1)\ldots(q_m,f_m),(q,f)} := \begin{cases} 1 & \text{falls } f = sum_{\sigma,q}((q_1,f_1), \ldots, (q_m,f_m)) \text{ und} \\ 0 & \text{sonst.} \end{cases}$$

Für jedes $\sigma \in \Sigma$, $q \in Q$ sei dabei $sum_{\sigma,q} : (Q^<)^m \to [0,z]^Y$ mit $m := rk_\Sigma(\sigma)$ jeweils definiert durch:

$$sum_{\sigma,q}((q_1,f_1), \ldots, (q_m,f_m))(k) := \begin{cases} rep_k\left(\sum_{i=1}^m f_i(k) + 1\right) & \text{falls } k = \mu(\sigma)_{q_1\ldots q_m,q} \text{ und} \\ rep_k\left(\sum_{i=1}^m f_i(k)\right) & \text{falls } k \neq \mu(\sigma)_{q_1\ldots q_m,q} \ . \end{cases}$$

Die Summenfunktion $sum_{\sigma,q}((q_1, f_1), \ldots, (q_m, f_m))$ vereinigt somit die Multimengen $f_1, \ldots, f_m$, fügt k hinzu und setzt dann $f(k)$ auf den entsprechenden Repräsentanten aus Y. Somit gilt für jeden Baum $t \in T_\Sigma$ und alle Läufe $r \in R_{M_\alpha^<}$ die Gleichung

$$k_1^{f(k_1)} \cdot \ldots \cdot k_n^{f(k_n)} = \mathrm{wt}_M^{en^<}(t, \pi_1 \circ r), \text{ wobei } f = \pi_2(r(\varepsilon)) \;.$$

Also erhalten wir ebenfalls $(S^<_{M_\alpha^<}, t) = \left|\left\{ r \in R_M(t) \;\middle|\; \mathrm{wt}_M^{en^<}(t, r) \cdot \gamma(r(\varepsilon)) = \alpha \right\}\right|$.

Nun gilt für die wta M_α^{TS} und $M_\alpha^<$:

$$(S_M^{TS}, t) = \sum_{r \in R_M(t)} \mathrm{wt}_M^{TS}(t, r) \cdot \gamma(r(\varepsilon)) = \sum_{\alpha \in Y} (S^{TS}_{M_\alpha^{TS}}, t)\alpha \quad \text{und}$$

$$(S_M^<, t) = \sum_{r \in R_M(t)} \mathrm{wt}_M^{en^<}(t, r) \cdot \gamma(r(\varepsilon)) = \sum_{\alpha \in Y} (S^<_{M_\alpha^<}, t)\alpha \quad .$$

Sei im Folgenden die Positionsaufzählung en entweder TS oder $en^<$. Mit S_α^{en} bezeichnen wir die formale Potenzreihe, welche für alle Bäume $t \in T_\Sigma$ definiert ist durch $(S_\alpha^{en}, t) := (S^{en}_{M_\alpha^{en}}, t)\alpha$. Wir zeigen, dass die Baumreihen S_α^{en} für alle $\alpha \in Y$ erkennbare Stufenfunktionen sind.

Für jedes $\alpha \in Y$ ist das additive Untermonoid $\langle\alpha\rangle^+$ endlich. Seien m_α^+, c_α^+ und d_α^+ wie oben beschrieben. Wir definieren nun für alle $n \in [0, d_\alpha^+]$ die Baumsprachen $L^{en}_{\alpha,n} := \{t \in T_\Sigma \mid (S_\alpha^{en}, t) = n\alpha\}$. Offenbar ist $L^{en}_{0,0} = T_\Sigma$. Damit sind alle Baumsprachen $L^{en}_{\alpha,n}$ nach Lemma 3.2.16 erkennbar, denn

$$L^{en}_{\alpha,n} = \begin{cases} \left\{ t \in T_\Sigma \;\middle|\; (S^{en}_{M_\alpha^{en}}, t) = n \right\} & \text{falls } 0 \le n < m_\alpha^+ \quad \text{und} \\ \left\{ t \in T_\Sigma \;\middle|\; (S^{en}_{M_\alpha^{en}}, t) \in (n + c_\alpha^+ \mathbb{N}_0) \right\} & \text{falls } m_\alpha^+ \le n \le d_\alpha^+ \;. \end{cases}$$

Es gibt nun zu jedem Baum $t \in T_\Sigma$ genau eine Zahl $n \in [0, d_\alpha^+]$ mit $(S_\alpha^{en}, t) = n\alpha$. Folglich ist $t \in L^{en}_{\alpha,n}$ und für jedes $\alpha \in Y$ bilden die Baumsprachen $L^{en}_{\alpha,n}$ mit $n \in [0, d_\alpha^+]$ eine Partition von T_Σ. Wir erhalten, dass

$$S_\alpha^{en} = \sum_{0 \le n \le d_\alpha^+} n\alpha \cdot \mathbb{1}_{L^{en}_{\alpha,n}}$$

eine erkennbare Stufenfunktion ist. Da die erkennbaren Baumsprachen unter den booleschen Operationen abgeschlossen sind, ist

$$S_M^{en} = \sum_{\alpha \in Y} S_\alpha^{en} = \sum_{\substack{\alpha \in Y \\ 0 \le n \le d_\alpha^+}} n\alpha \cdot \mathbb{1}_{L^{en}_{\alpha,n}}$$

ebenfalls eine erkennbare Stufenfunktion. □

Als unmittelbare Konsequenz dieses Satzes erhalten wir, dass in bi-lokal endlichen Bimonoiden die TS- und $<$-erkennbaren Reihen genau die erkennbaren Stufenfunktionen sind. Außerdem fallen damit die Begriffe *TS-erkennbar* und *$<$-erkennbar* zusammen und ebenso *$<$-erkennbar* und *schwach $<$-erkennbar*. Weiterhin folgt, dass in einem bi-lokal endlichen Bimonoid $<$-Erkennbarkeit von der Wahl der linearen Ordnung unabhängig ist. Dies fassen wir im folgenden Korollar zusammen.

Korollar 3.2.18. *Sei K bi-lokal endlich und durch $<$ linear geordnet und $S \in K\langle\langle T_\Sigma \rangle\rangle$. Es sind äquivalent:*

1. *S ist eine erkennbare Stufenfunktion.*
2. *S ist TS-erkennbar.*
3. *S ist $<$-erkennbar.*
4. *S ist schwach $<$-erkennbar.*

Sei $\prec$ eine weitere lineare Ordnung auf K, dann ist zu obigen ebenfalls äquivalent:

5. *S ist $\prec$-erkennbar*

Beweis. Die Implikationen 1. $\Rightarrow$ 2. und 1. $\Rightarrow$ 3. und 1. $\Rightarrow$ 5. folgen aus Lemma 3.2.9, denn dieses besagt, dass $S \in K\langle\langle T_\Sigma \rangle\rangle$ genau dann eine erkennbare Stufenfunktion ist, wenn $S = S_M$ eines crisp-deterministischen Baumautomaten M. Der im Beweis konstruierte wta M ist sogar für alle drei Fälle derselbe. Lemma 3.2.10 besagt, dass jede Stufenfunktion schwach erkennbar ist; es gilt also 1. $\Rightarrow$ 4.

Die Umkehrungen dieser Implikationen folgen hingegen alle direkt aus Satz 3.2.17. □

3.3 Abschlusseigenschaften erkennbarer Baumreihen

In diesem Abschnitt führen wir verschiedene Operationen auf formalen Potenzreihen ein, welche wir im nachfolgenden Kapitel benötigen werden. Wir untersuchen anschließend den Erhalt der Erkennbarkeit formaler Potenzreihen unter diesen Operationen. Wir betrachten dabei sowohl erkennbare Stufenfunktionen wie auch schwach erkennbare und erkennbare Reihen, wobei jeweils eine Positionsaufzählung für Gewichtsbäume vorausgesetzt ist.

Im Folgenden seien stets, sofern nicht anders gefordert, Σ ein Rangalphabet und K ein starkes Bimonoid.

Proposition 3.3.1. *Sei en eine Positionsaufzählung für Gewichtsbäume. Ist $L \subseteq T_\Sigma$ eine erkennbare Baumsprache, so ist $\mathbb{1}_L \in K\langle\langle T_\Sigma \rangle\rangle$ en-erkennbar.*

Beweis. Die Baumsprache L ist erkennbar, also ist $\mathbb{1}_L$ eine erkennbare Stufenfunktion. Nach Lemma 3.2.9 gibt es damit einen crisp-deterministischen wta M mit $S_M = \mathbb{1}_L$. □

Proposition 3.3.2. *Sei $k \in K$ und en eine Positionsaufzählung für Gewichtsbäume. Die konstante Reihe $\tilde{k} \in K\langle\langle T_\Sigma \rangle\rangle$, mit $(\tilde{k}, t) := k$ für alle $t \in T_\Sigma$, ist en-erkennbar.*

Beweis. $\tilde{k} = k \cdot \mathbb{1}_{T_\Sigma}$ ist eine erkennbare Stufenfunktion und damit nach Lemma 3.2.9 erkennbar. □

Als Nächstes betrachten wir die Operationen *Summe* (+) und *Hadamardprodukt* ($\odot$), welche die Operationen eines Bimonoides auf formale Potenzreihen über einem Bimonoid erweitern. Sie sind für alle $S, T \in K\langle\langle T_\Sigma \rangle\rangle$ und alle Bäume $t \in T_\Sigma$ wie folgt definiert:

$$(S + T, t) := (S, t) + (T, t) \qquad \text{und} \qquad (S \odot T, t) := (S, t) \cdot (T, t)\ .$$

Mithilfe des Hadamard-Produktes lässt sich auch das Skalarprodukt darstellen. Sei $k \in K$ und $S \in K\langle\langle T_\Sigma \rangle\rangle$. Dann ist $k \cdot S = \tilde{k} \odot S$ und $S \cdot k = S \odot \tilde{k}$.

Proposition 3.3.3. *Seien $S, T \in K\langle\langle T_\Sigma \rangle\rangle$ und en eine Positionsaufzählung für Gewichtsbäume. Sind S und T (schwach) en-erkennbar, dann ist auch $S + T$ (schwach) en-erkennbar.*

Beweis. Sei M_S der wta mit $S = S_{M_S}$ und M_T der wta mit $T = S_{M_T}$. Wir bilden nun den Vereinigungsautomaten M der wta M_S und M_T. Für alle $t \in T_\Sigma$ gilt damit

$$(S_M, t) = \sum_{r \in R_M(t)} \mathrm{wt}_M^{en}(t, r) = \sum_{r \in R_{M_S}(t)} \mathrm{wt}_{M_S}^{en}(t, r) + \sum_{r \in R_{M_T}(t)} \mathrm{wt}_{M_T}^{en}(t, r) = (S + T, t)\ .$$

Sind S und T schwach *en*-erkennbar, so sind M_S und M_T schwach und somit auch M. Dann ist $S + T$ ebenfalls schwach erkennbar. □

Das Hadamard-Produkt zweier erkennbarer formaler Potenzreihen ist im Allgemeinen nur erkennbar, wenn das zugrunde liegende Bimonoid kommutativ ist. Wir benötigen jedoch nur die Fälle, in denen eine der beiden Potenzreihen eine charakteristische Funktion ist oder in denen beide Potenzreihen erkennbare Stufenfunktionen sind. In diesen Fällen erhalten wir ein positives Resultat.

Proposition 3.3.4. *Sei en eine Positionsaufzählung für Gewichtsbäume und $L \subseteq T_\Sigma$ erkennbar. Ist $S \in K\langle\langle T_\Sigma \rangle\rangle$ (schwach) en-erkennbar, dann ist auch $S \odot \mathbb{1}_L$ (schwach) en-erkennbar.*

Beweis. Wir bilden das direkte Produkt eines wta $M_1 = (Q_1, \Sigma, \mu_1, \gamma_1)$, welcher S *en*-erkennt und des crisp-deterministischen wwta $M_2 = (Q_2, \Sigma, \mu_2, \gamma_2)$ aus dem Beweis von Proposition 3.3.1, der $\mathbb{1}_L$ erkennt.

Es sei dazu $M := (Q, \Sigma, \mu, \gamma)$ mit

$$\begin{aligned} Q &:= Q_1 \times Q_2 \;, \\ \mu(\sigma)_{(q_1,p_1)\dots(q_m,p_m),(q,p)} &:= \mu_1(\sigma)_{q_1\dots q_m,q} \cdot \mu_2(\sigma)_{p_1\dots p_m,p} \quad \text{für } \sigma \in \Sigma^{(m)} \text{ und} \\ \gamma((q,p)) &:= \gamma_1(q) \cdot \gamma_2(p) \;. \end{aligned}$$

Weil M_2 crisp-deterministisch, gilt $\operatorname{img}(\mu_2) \subseteq \{0,1\} \supseteq \operatorname{img}(\gamma_2)$. Da 0 und 1 in jedem Bimonoid kommutieren, gilt für jeden Baum $t \in T_\Sigma$ und jeden Lauf $r \in R_M(t)$ damit $\mathrm{wt}^{en}_M(t,r) \cdot \gamma(r(\varepsilon)) = \mathrm{wt}^{en}_{M_1}(t, \pi_1 \circ r) \cdot \gamma_1(\pi_1 r(\varepsilon)) \cdot \mathrm{wt}^{en}_{M_2}(t, \pi_2 \circ r) \cdot \gamma_2(\pi_2 r(\varepsilon))$. Mit dem Determinismus von M_2 folgt schließlich

$$\begin{aligned} (S_M, t) &= \sum_{r \in R_M(t)} \mathrm{wt}^{en}_M(t,r) \cdot \gamma(r(\varepsilon)) \\ &= \sum_{r_1 \in R_{M_1}(t)} \mathrm{wt}^{en}_{M_1}(t,r_1) \cdot \gamma_1(r_1(\varepsilon)) \quad \cdot \sum_{r_2 \in R_{M_2}(t)} \mathrm{wt}^{en}_{M_2}(t,r_2) \cdot \gamma_2(r_2(\varepsilon)) \\ &= (S \odot \mathbb{1}_L, t) \;. \end{aligned}$$

Sei S schwach *en*-erkennbar. Dann ist M ein schwacher Automat und damit $S \odot \mathbb{1}_L$ ebenfalls schwach *en*-erkennbar. □

Weil 0 und 1 mit jedem Element eines Bimonoides kommutieren, ist $\mathbb{1}_L \odot S = S \odot \mathbb{1}_L$. Nach dem obigen Lemma ist $\mathbb{1}_L \odot S$ somit *en*-erkennbar, wenn S *en*-erkennbar ist.

Proposition 3.3.5. *Sind $S, T \in K\langle\langle T_\Sigma \rangle\rangle$ erkennbare Stufenfunktionen, so sind auch $S + T$ und $S \odot T$ erkennbare Stufenfunktionen.*

Beweis. Wir verfahren wie in [18, proof of Lemma 3.1].

Sei $S = \sum_{i=1}^n l_i \cdot \mathbb{1}_{L_i}$ und $T = \sum_{j=1}^m k_j \cdot \mathbb{1}_{K_j}$. Dabei bilden die Familien $(L_i \mid i \in [1,n])$ und $(K_j \mid j \in [1,m])$ jeweils Partitionen von T_Σ und $l_i, k_j \in K$ für alle $i \in [1,n]$ und $j \in [1,m]$. Dann ist

$$\begin{aligned} S + T &= \sum_{\substack{1 \le i \le n \\ 1 \le j \le m}} (l_i + k_j) \cdot \mathbb{1}_{L_i \cap K_j} \quad \text{und} \\ S \odot T &= \sum_{\substack{1 \le i \le n \\ 1 \le j \le m}} (l_i \cdot k_j) \cdot \mathbb{1}_{L_i \cap K_j} \;. \end{aligned}$$

Falls die Elemente $(k_i + l_j)$ beziehungsweise $(k_i \cdot l_j)$ für $i \in [1,n]$ und $j \in [1,m]$ nicht paarweise verschieden sind, so bilden wir die Vereinigung der entsprechenden Mengen $(K_i \cap L_j)$. Somit sind $S + T$ und $S \odot T$ erkennbare Stufenfunktionen. □

Weil $\tilde{k}$ für jedes $k \in K$ eine erkennbare Stufenfunktion ist, folgt nun sofort für jede erkennbare Stufenfunktion S, dass die Skalarprodukte $k \cdot S = \tilde{k} \odot S$ und $S \cdot k = S \odot \tilde{k}$ ebenfalls erkennbare Stufenfunktionen sind.

Im nächsten Schritt betrachten wir Umbenennungen von Knoten eines Baumes. Diese spielen im nachfolgenden Abschnitt eine zentrale Rolle. Seien dazu Σ und Δ zwei Rangalphabete und $\tau : \Sigma \to \wp(\Delta)$ eine Abbildung derart, dass für alle $\sigma \in \Sigma^{(m)}$ stets $\tau(\sigma) \subseteq \Delta^{(m)}$ gilt. Wir betrachten jede Funktion $\tau : \Sigma \to \Delta$ ebenfalls als Umbenennung, indem wir τ mit $\theta : \Sigma \to \wp(\Delta)$ identifizieren, wobei $\theta(\sigma) := \{\tau(\sigma)\}$ gelte.

Wir erweitern τ nun zu einer Abbildung $\tau' : T_\Sigma \to \wp(T_\Delta)$. Dazu definieren wir τ' für alle $\sigma \in \Sigma^{(m)}$ und $s_1, \ldots, s_m \in T_\Sigma$ induktiv über die Termschreibweise von Bäumen:

$$\tau'(\sigma(s_1, \ldots, s_m)) := \left\{\delta(t_1, \ldots, t_m) \;\middle|\; \delta \in \tau(\sigma) \;\wedge\; \forall i \in [1, m] : t_i \in \tau'(s_i)\right\} .$$

Offenbar ist die Menge aller Urbilder $\tau'^{-1}(t) = \{s \mid t \in \tau'(s)\}$ eines Baumes $t \in T_\Delta$ unter τ endlich.

Schließlich bilden wir aus τ' eine dritte Funktion $\tau'' : K\langle\langle T_\Sigma\rangle\rangle \to K\langle\langle T_\Delta\rangle\rangle$. Für alle $S \in K\langle\langle T_\Sigma\rangle\rangle$ und $t \in T_\Delta$ definieren wir hierzu

$$(\tau''(S), t) := \sum_{s \in \tau'^{-1}(t)} (S, s) .$$

Wir werden τ' und τ'' ebenfalls mit τ bezeichnen, da aus dem Kontext ersichtlich ist, welche Funktion von uns verwendet wird.

Der Abschluss erkennbarer Baumreihen unter Umbenennung wurde für kommutative stetige Semiringe in [25, Corollar 14] gezeigt und in [17, Lemma 3.4] auf beliebige Semiringe verallgemeinert. Wir zeigen nun den Abschluss der Erkennbarkeit mit einer Positionsaufzählung für Gewichtsbäume in starken Bimonoiden.

Lemma 3.3.6. *Sei $S \in K\langle\langle T_\Sigma\rangle\rangle$ (schwach) en-erkennbar und $\tau : K\langle\langle T_\Sigma\rangle\rangle \to K\langle\langle T_\Delta\rangle\rangle$ eine Umbenennung. Dann ist $\tau(S)$ ebenfalls (schwach) en-erkennbar.*

Beweis. Sei $M = (Q, \Sigma, \mu, \gamma)$ ein gewichteter Baumautomat, der S mit *en* erkennt. Konstruiere den wta $M' = (Q', \Delta, \mu', \gamma')$, wobei für alle $q_1, \ldots, q_m, q \in Q$ und alle $\sigma \in \Sigma, \delta \in \Delta$ gelte:

$$\begin{aligned} Q' &:= Q \times \Sigma , \\ \mu'(\delta)_{(q_1,\sigma_1)\ldots(q_m,\sigma_m),(q,\sigma)} &:= \begin{cases} \mu(\sigma)_{q_1\ldots q_m,q} & \text{falls } \delta \in \tau(\sigma) , \\ 0 & \text{sonst und} \end{cases} \\ \gamma'((q, \sigma)) &:= \gamma(q) . \end{aligned}$$

Für jedes Paar von Bäumen $s \in T_\Sigma$ und $t \in \tau(s)$ gilt $\mathrm{pos}(s) = \mathrm{pos}(\tau(s))$. Außerdem gibt es für jeden Lauf r von M auf s genau einen Lauf r' von M' auf t, der r codiert:

$$\forall s \in T_\Sigma, t \in \tau(s), r \in R_M(s)\, \exists! r' \in R_{M'}(t) \;:\; r = \pi_1 \circ r' \;\wedge\; s = \pi_2 \circ r' .$$

Umgekehrt gilt für jeden Baum $t \in T_\Delta$ und alle $s \in \tau^{-1}(t) : \mathrm{pos}(t) = \mathrm{pos}(s)$. Die Transitionsfunktion μ' berücksichtigt die Umbenennung τ derart, dass jeder Lauf in M'

auf t das Gewicht 0 erhält oder einen Baum $s \in \tau^{-1}(t) \in T_\Sigma$ und einen Lauf von M auf s codiert:

$$\forall t \in T_\Delta, r' \in R_{M'}(t) \,:\, \mathrm{wt}^{en}_{M'}(t,r') = 0 \quad \vee \quad (\exists! s \in \tau^{-1}(t), r \in R_M(s) \,:\, r = \pi_1 \circ r' \wedge s = \pi_2 \circ r') \,.$$

Seien $s \in T_\Sigma$, $t \in \tau(s)$ und $r' \in R_{M'}(t)$. Für $r := \pi_1 \circ r'$ gilt dann $\mathrm{wt}^{en}_{M}(s,r) = \mathrm{wt}^{en}_{M'}(t,r')$. Für alle $t \in T_\Delta$, erhalten wir nun:

$$\begin{aligned}
(S_{M'},t) &= \sum_{r' \in R_{M'}(t)} \mathrm{wt}^{en}_{M'}(t,r') \cdot \gamma'(r'(\varepsilon)) \\
&= \sum_{\substack{r' \in R_{M'}(t) \\ \pi_2 \circ r' \in T_\Sigma}} \mathrm{wt}^{en}_{M'}(t,r') \cdot \gamma'(r'(\varepsilon)) \\
&= \sum_{s \in \tau^{-1}(t)} \sum_{\substack{r' \in R_{M'}(t) \\ s = \pi_2 \circ r'}} \mathrm{wt}^{en}_{M'}(t,r') \cdot \gamma'(r'(\varepsilon)) \\
&= \sum_{s \in \tau^{-1}(t)} \sum_{r \in R_M(s)} \mathrm{wt}^{en}_{M}(s,r) \cdot \gamma(r(\varepsilon)) \\
&= \sum_{s \in \tau^{-1}(t)} (S_M, s) \\
&= (\tau(S),t) \,.
\end{aligned}$$

Folglich ist $\tau(S) = S_{M'}$ *en* erkennbar.

Sei S schwach *en*-erkennbar. Dann ist M' schwach, denn $\mathrm{img}(\gamma') \subseteq \{0,1\}$. Also ist $\tau(S)$ ebenfalls schwach *en*-erkennbar. □

Man beachte, dass erkennbare Stufenfunktionen nicht durch Umbenennungen erhalten werden, was das folgende Beispiel über $\mathbb{N}_0$ zeigt. Sei $\Sigma^{(1)} := \{\alpha, \beta\}$, $\Sigma^{(0)} := \{\gamma\}$ und $\Delta^{(1)} := \{\delta\}$, $\Delta^{(0)} := \{\zeta\}$. Wir betrachten die konstante Baumreihe $S := \tilde{1}$ sowie die Umbenennung $\tau : \Sigma \to \wp(\Delta)$, wobei $\tau(\alpha) := \tau(\beta) := \{\delta\}$ und $\tau(\gamma) := \{\zeta\}$. Nun hat die erkennbare Reihe

$$(\tau(S),t) \;=\; \sum_{s \in \tau^{-1}(t)} (S,s) \;=\; \left|\tau^{-1}(t)\right| \;=\; 2^{|\mathrm{pos}(t)-1|}$$

offensichtlich kein endliches Bild und ist damit keine erkennbare Stufenfunktion.

Proposition 3.3.7. *Seien Σ und Δ Rangalphabete, K ein multiplikativ lokal endliches kommutatives Bimonoid und $\tau : T_\Sigma \to T_\Delta$ eine Umbenennung. Weiterhin sei die Abbildung $\Pi_\tau : K\langle\langle T_\Sigma \rangle\rangle \to K\langle\langle T_\Delta \rangle\rangle$ für alle $S \in K\langle\langle T_\Sigma \rangle\rangle$ und alle $t \in T_\Delta$ definiert mit $(\Pi_\tau(S),t) := \prod_{s \in \tau^{-1}(t)} (S,s)$. Ist S eine erkennbare Stufenfunktion, so ist auch auch $\Pi_\tau(S)$ eine erkennbare Stufenfunktion.*

Beweis. Sei die Baumreihe $S \in K\langle\langle T_\Sigma \rangle\rangle$ eine erkennbare Stufenfunktion. Dann gibt es Baumsprachen $L_1, \ldots, L_n$, die T_Σ partitionieren, und paarweise verschiedene Elemente $k_1, \ldots, k_n \in K$ für ein $n \in \mathbb{N}$, sodass $S = \sum_{i=1}^n k_i \cdot \mathbb{1}_{L_i}$. Wir betrachten zunächst die

Erweiterung von τ auf die Menge $\mathbb{N}_0\langle\langle T_\Sigma\rangle\rangle$. Für $i \in [1,n]$ und $\mathbb{1}_{L_i} \in \mathbb{N}_0\langle\langle T_\Sigma\rangle\rangle$ seien $m_i \in K\langle\langle T_\Delta\rangle\rangle$ Baumreihen mit

$$(m_i, t) := (\tau(\mathbb{1}_{L_i}), t) = \sum_{s\in\tau^{-1}(t)} (\mathbb{1}_{L_i}, s) = \left|\tau^{-1}(t) \cap L_i\right|$$

für alle $t \in T_\Delta$. Nach Lemma 3.3.6 sind diese TS-erkennbar. Sie geben an, wie viele Urbilder von $\tau(t)$ in der jeweiligen Sprache L_i liegen. Nach Lemma 3.2.16 sind dann die Baumsprachen$L_i^m := \tau(\mathbb{1}_{L_i})^{-1}(m)$ für alle $m \in \mathbb{N}_0$ erkennbar.

Wir betrachten nun die Potenzreihe $\Pi_\tau(S)$. Sei dazu $t \in T_\Delta$. Da K kommutativ ist, gilt

$$(\Pi_\tau(S), t) = \prod_{s\in\tau^{-1}(t)} (S, s) \;=\; k_1^{(m_1,t)} \cdot \ldots \cdot k_n^{(m_n,t)} \,.$$

Außerdem ist K multiplikativ lokal endlich. Also gibt es in jedem multiplikativen Untermonoid $\langle k_i\rangle^*$ mit $i \in [1,n]$ für alle $m \in \mathbb{N}_0$ einen minimalen Repräsentanten $rep_{k_i}(m)$ mit $k_i^{rep_{k_i}(m)} = k_i^m$. Folglich ist die Menge

$$M := \{(m_1,\ldots,m_n) \mid m_1 \in \mathrm{img}(rep_{k_1}) \;\wedge\; \ldots \;\wedge\; m_n \in \mathrm{img}(rep_{k_n})\}$$

endlich. Für $(m_1,\ldots,m_n) \in M$ sei $L_{(m_1,\ldots,m_n)} := L_1^{m_1} \cap \ldots \cap L_n^{m_n}$. Da die erkennbaren Baumsprachen unter Schnitt abgeschlossen sind, folgt aus $(\Pi_\tau(S), t) = k_1^{(m_1,t)} \cdot \ldots \cdot k_n^{(m_n,t)}$ für alle $t \in T_\Delta$, dass $\Pi_\tau(S)$ eine erkennbare Stufenfunktion ist:

$$\Pi_\tau(S) = \sum_{(m_1,\ldots,m_n)\in M} (k_1^{m_1} \cdot \ldots \cdot k_n^{m_n}) \cdot \mathbb{1}_{L_{(m_1,\ldots,m_n)}} \,.$$

□

Wir fassen nun die Abschlusseigenschaften für die Klasse $K^{en-wrec}\langle\langle T_\Sigma\rangle\rangle$ der *en*-erkennbaren Baumreihen und die Klasse $K^{en-rec}\langle\langle T_\Sigma\rangle\rangle$ der schwach *en*-erkennbaren Baumreihen im nachfolgenden Satz zusammen.

Satz 3.3.8. *Sei K ein starkes Bimonoid und en eine Positionsaufzählung für Gewichtsbäume. Die Klassen $K^{en-rec}\langle\langle T_\Sigma\rangle\rangle$ und $K^{en-wrec}\langle\langle T_\Sigma\rangle\rangle$ sind abgeschlossen unter Summe, Hadamard-Produkt mit erkennbaren charakteristischen Funktionen und Umbenennung.*

Beweis. Die Aussage folgt direkt aus Proposition 3.3.3, Proposition 3.3.4 und Lemma 3.3.6. □

4 Erkennbarkeit und Definierbarkeit in wMSO

4.1 MSO-Logik über Bäumen

Wir rekapitulieren in diesem Abschnitt kurz die monadische Logik zweiter Stufe über Bäumen und das für uns zentrale Ergebnis im Bezug auf erkennbare Baumsprachen: Äquivalent zu Büchis Ergebnis für Wortsprachen sind die erkennbaren Baumsprachen genau die MSO-definierbaren Baumsprachen [11, 28].

Sei Σ ein Rangalphabet. Die Menge $\mathrm{MSO}(\Sigma)$ aller Formeln der monadischen Logik zweiter Stufe über Σ definieren wir mit der folgenden Grammatik:

$$\phi := 0 \mid 1 \mid label_\sigma(x) \mid edge_i(x,y) \mid x \in X \mid \neg\phi \mid \phi \wedge \phi \mid \forall x.\phi \mid \forall X.\phi \;.$$

Dabei seien x, y Variablen erster Stufe, X eine Variable zweiter Stufe, $\sigma \in \Sigma$ und $i \in [1, max_\Sigma]$. Quantifiziert ein Quantor $\forall$ beziehungsweise $\exists$ eine Variable erste Stufe (Individuenvariable), so nennen wir ihn Quantor erster Stufe. Analog dazu sprechen wir von einem Quantor zweiter Stufe, falls dieser sich auf eine Variable zweiter Stufe (Mengenvariable) bezieht. Man verwendet zur praktikablen Beschreibung von Baumsprachen die folgenden üblichen Abkürzungen, wobei $\phi, \psi \in \mathrm{MSO}(\Sigma)$:

$$\begin{aligned}
\phi \vee \psi &:= \neg(\neg\phi \wedge \neg\psi) \;, \\
\phi \to \psi &:= \neg\phi \vee \psi \;, \\
\exists x.\phi &:= \neg\forall x.\neg\phi \quad \text{und} \\
\exists X.\phi &:= \neg\forall X.\neg\phi \;.
\end{aligned}$$

Wir könnten auch auf die Konstanten 0 und 1 verzichten und sie als Abkürzungen $0 := \forall x.edge_1(x,x)$ und $1 := \neg 0$ einführen. Durch die gewählte Definition wird jedoch die Verbindung zur nachfolgenden gewichteten MSO-Logik deutlicher. Die Menge der freien Variablen einer Formel ϕ umfasst alle Variablen, die in ϕ vorkommen und sich nicht im Wirkungsbereich eines Quantors befinden (sprich: ungebunden sind). Wir bezeichnen sie mit $\mathrm{Free}(\phi)$.

Sei V eine endliche Variablenmenge von Variablen erster und zweiter Stufe. Das Rangalphabet $\Sigma_V := (\Sigma \times \{0,1\}^V, rk)$ ist definiert durch $rk((\sigma, f)) := rk_\Sigma(\sigma)$ für alle $\sigma \in \Sigma$ und alle $f \in \{0,1\}^V$. Wir kürzen die Projektion $\pi_1(\sigma, f) = \sigma$ auf die erste Komponente eines Symbols $(\sigma, f) \in \Sigma_V$ mit $(\sigma, f)_1$ ab. Ebenso steht $(\sigma, f)_2$ für $\pi_2(\sigma, f) = f$. Ein Baum $t \in T_{\Sigma_V}$ heißt *korrekt*, falls es für alle Variablen $x \in V$ erster Stufe genau eine Position $w \in \mathrm{pos}(t)$ mit $(t(w)_2)(x) = 1$ gibt. Die Menge aller korrekten Bäume aus T_Σ bezeichnen wir mit $T^v_{\Sigma_V}$ (valid trees) und Σ_ϕ kürzt $\Sigma_{\mathrm{Free}(\phi)}$ ab.

Sei t ein Baum aus T_Σ. Eine Abbildung $\rho : V \to \mathrm{pos}(t) \cup \wp(\mathrm{pos}(t))$ nennen wir (V,t)-*Variablenbelegung*, wenn $\rho(x) \in \mathrm{pos}(t)$ für alle Variablen x erster Stufe gilt und

$\rho(X) \subseteq \text{pos}(t)$ für alle Variablen X zweiter Stufe. Wir definieren nun die Funktion va (variable assignment), welche jeden Baum $s \in T^v_{\Sigma_V}$ auf die durch s induzierte (V,t)-Variablenbelegung abbildet. Es sei $va : T^v_{\Sigma_V} \to \bigcup_{t \in T_\Sigma} (\wp(\text{pos}(t)) \cup \text{pos}(t))^V$ für alle $s \in T^v_{\Sigma_V}$ definiert mit

$$\begin{aligned} va(s)(x) = w &\iff s(w)_2(x) = 1 \text{ falls } x \text{ Variable erster Stufe} \quad \text{und} \\ va(s)(X) \ni w &\iff s(w)_2(X) = 1 \text{ falls } X \text{ Variable zweiter Stufe.} \end{aligned}$$

Für jedes $t \in T_\Sigma$ ist die auf $M_t := \left\{ s \in T^v_{\Sigma_V} \,\middle|\, \pi_1 \circ s = t \right\}$ eingeschränkte Abbildung $va|_{M_t}$ eine Bijektion zwischen M_t und den zugehörigen (V,t)-Variablenbelegungen. Deshalb können wir jeden Baum $s \in T^v_{\Sigma_V}$ mit einem Paar (t,ρ) identifizieren, wobei $t := \pi_1 \circ s$ und $\rho := va(s)$. Umgekehrt entspricht jedem Paar (t,ρ) aus einem $t \in T_\Sigma$ und einer (V,t)-Variablenbelegung ρ derjenige Baum $s \in T^v_{\Sigma_V}$, für welchen $t = \pi_1 \circ s$ und $\rho = va(s)$. Wir schreiben daher $s = (t,\rho)$.

Sei s ein Baum aus T_{Σ_V}, $x \in V$ eine Variable erster Stufe und $X \in V$ eine Variable zweiter Stufe. Für eine Position $u \in \text{pos}(s)$ entsteht der Baum $s[x \to u] \in T_{\Sigma_{V \cup \{x\}}}$ aus s, indem wir $(s[x \to u](v)_2)(x) = 1 \iff v = u$ für alle $v \in \text{pos}(s)$ setzen. Analog dazu entsteht aus s der Baum $s[X \to I]$ für eine Menge $I \subseteq \text{pos}(s)$ durch Setzen von $(s[X \to I](v)_2) = 1 \iff v \in I$. Mit $s = (t,\rho)$ schreiben wir diese auch als $(t, \rho[x \to u])$ beziehungsweise $(t, \rho[X \to I])$. Ist s korrekt, so sind $s[x \to u]$ und $s[X \to I]$ ebenfalls korrekte Bäume.

Sei $\phi \in \text{MSO}$ und $s \in T_{\Sigma_V}$. Wir führen nun die *Erfüllbarkeitsrelation* $s \models \phi$ ein. Falls ein Baum s nicht korrekt ist, so erfüllt s keine Formel; es gilt stets $s \not\models \phi$. Ist s hingegen korrekt, definieren wir die Erfüllbarkeitsrelation induktiv über den Formelaufbau. Seien dazu $\phi\psi \in \text{MSO}(\Sigma)$, $\sigma \in \Sigma$, $i \in [1, rk_\Sigma(\sigma)]$ und $x, y \in V$ Variablen erster Stufe sowie $X \in V$ eine Variable zweiter Stufe. Wir setzen:

$$\begin{aligned} (t,\rho) &\not\models 0 \\ ,(t,\rho) &\models 1 \\ ,(t,\rho) \models label_\sigma(x) &\iff t(\rho(x)) = \sigma\ , \\ (t,\rho) \models edge_i(x,y) &\iff \rho(y) = \rho(x).i\ , \\ (t,\rho) \models \neg\phi &\iff (t,\rho) \not\models \phi\ , \\ (t,\rho) \models \phi \wedge \psi &\iff (t,\rho) \models \phi \text{ und } (t,\rho) \models \psi\ , \\ (t,\rho) \models \forall x.\phi &\iff (t,\rho[x \to u]) \models \phi \text{ für alle } u \in \text{pos}\, t \quad \text{und} \\ (t,\rho) \models \forall X.\phi &\iff (t,\rho[X \to I]) \models \phi \text{ für alle } I \subseteq \text{pos}\, t\ . \end{aligned}$$

Sei nun $\phi \in \text{MSO}(\Sigma)$ eine Formel und V eine Variablenmenge, die $\text{Free}(\phi)$ umfasst. Dann *definiert* ϕ die Baumsprache

$$L_V(\phi) := \{ s \in T_{\Sigma_V} \mid s \models \phi \}\,.$$

Wir beobachten, dass $L_V(\phi)$ offenbar nur korrekte Bäume enthält. Im Folgenden benutzen wir die Abkürzung $L(\phi)$ für die Sprache $L_{\text{Free}(\phi)}(\phi)$.

Betrachten wir die Menge Σ_V für $V = \emptyset$. Die einzige Funktion aus $\{0,1\}^V$ ist dann die leere Funktion $\emptyset$. Also haben alle Elemente aus Σ_V die Form $(\sigma, \emptyset)$ mit $\sigma \in \Sigma$. Somit existiert eine natürliche Bijektionen η zwischen Σ und Σ_V mit $\eta((\sigma, \emptyset)) := \sigma$ für alle $\sigma \in \Sigma$. Die entsprechende Erweiterung von η auf eine Abbildung von T_Σ auf T_{Σ_V} ist ebenfalls eine Bijektionen. In diesem Sinne identifizieren wir die Mengen T_Σ und T_{Σ_V} im Folgenden jeweils miteinander. Von zentraler Bedeutung der folgende Satz.

Satz 4.1.1 (Büchi)**.** *Eine Baumsprache $L \subseteq T_\Sigma$ ist genau dann MSO-definierbar, wenn L erkennbar ist [28, Theorems 14, 17], [11, Theorems 3.7, 3.9].*

Die Menge $T^v_{\Sigma_V}$ ist eine erkennbare Baumsprache. Betrachte für jede Variable $x \in V$ erster Stufe den Baumautomaten $A_x := (Q, \Sigma_V, \delta_x, F)$ mit $Q := \{0, 1, 2\}$ und $F := \{1\}$. Für $(\sigma, f) \in \Sigma_V^{(m)}$ und $q_1, \ldots, q_m \in Q$ sei die Transitionsfunktion wie folgt definiert:

$$\delta_{x,(\sigma,f)}(q_1, \ldots, q_m) := \begin{cases} 1 & \text{falls } q_1 + \ldots + q_m = 0 \text{ und } f(x) = 1 \ , \\ 2 & \text{falls } q_1 + \ldots + q_m + f(x) > 1 \ , \\ 0 & \text{sonst.} \end{cases}$$

Der Baumautomat A_x erkennt die Sprache

$$L(A_x) = \{t \in T_{\Sigma_V} \mid \exists! u \in \mathrm{pos}(t) : \pi_2(t(u))(x) = 1\} \ .$$

Damit folgt die Erkennbarkeit von $T^v_{\Sigma_V}$ aus

$$T^v_{\Sigma_V} = \bigcap_{\substack{x \in V \\ x \text{ ist erste Stufe}}} L(A_x) \ .$$

4.2 Gewichtete MSO-Logik über Bäumen

In diesem Abschnitt führen wir die gewichtete monadische Logik zweiter Stufe über Bäumen ein und betrachten die Verbindung zum ungewichteten Fall. Es stellt sich heraus, dass die Erkennbarkeit der Semantik einer Formel durch einen gewichteten Baumautomaten nicht von der Wahl der Variablenmenge abhängt, sofern in ihr alle Variablen der Formel enthalten sind.

Im Vergleich zu der zugrunde liegenden Arbeit von Droste und Vogler über gewichtete Baumautomaten [17] auf Semiringen, definieren wir die gewichtete monadische Logik zweiter Stufe geringfügig anders, entsprechend dem Vorgehen von Bollig and Gastin [4]. Während beide Ansätze äquivalent sind, benötigen wir mit dieser Definition keine syntaktische Transformation um Mehrdeutigkeiten aufzulösen.

Definition 4.2.1. Sei K ein starkes Bimonoid mit Negation, Σ ein Rangalphabet und V eine endliche Menge von Variablen. Weiterhin seien $x, y \in V$ Variablen erster Stufe, $X \in V$ eine Variable zweiter Stufe, $k \in K$, $\sigma \in \Sigma$, und $i \in [1, max_\Sigma]$. Die Menge $\mathrm{wMSO}(K,\Sigma)$ der Formeln der *gewichteten monadischen Logik zweiter Stufe* über K und Σ definieren wir mit der folgenden Grammatik:

$$\begin{aligned}\phi := {} & k \mid label_\sigma(x) \mid edge_i(x,y) \mid x \in X \mid \neg\phi \mid \phi \wedge \phi \mid \phi \vee \phi \mid \\ & \forall x.\phi \mid \forall X.\phi \mid \exists x.\phi \mid \exists X.\phi\ .\end{aligned}$$

Die Formeln k, $label_\sigma(x)$ und $edge_i(x,y)$ wollen wir *Atomformeln* nennen, alle Formeln k auch *konstant*. Eine Formel $\phi \in \mathrm{wMSO}(K,\Sigma)$ heißt *boolesch*, falls in ϕ weder Disjunktionen, Existenzquantoren erster oder zweiter Stufe noch konstante Teilformeln außer 0 und 1 vorkommen:

$$\begin{aligned}\phi_{bool} := {} & 0 \mid 1 \mid label_\sigma(x) \mid edge_i(x,y) \mid x \in X \mid \neg\phi_{bool} \mid \phi_{bool} \wedge \phi_{bool} \mid \\ & \forall x.\phi_{bool} \mid \forall X.\phi_{bool}\ .\end{aligned}$$

Somit sind die Formeln des booleschen Fragmentes (BOOL) genau die Formeln der ungewichteten monadischen Logik zweiter Stufe.

Im Folgenden sei $K := (K, +, \cdot, \bar{}, 0, 1)$ stets ein starkes Bimonoid mit Negation und Σ ein Rangalphabet, sofern nicht anders gefordert.

Die Semantik der Formeln aus wMSO soll in unserem Kontext jeweils eine formale Potenzreihe sein. Zur Interpretation der Allquantifikation erster Stufe als Produkt benötigen wir wieder eine Positionsaufzählung für Gewichtsbäume en. Die durch ϕ definierte Potenzreihe bezeichnen wir mit $[\![\phi]\!]_V^{en}$, wobei $\phi \in \mathrm{wMSO}(K,\Sigma)$ und V eine Variablenmenge mit $\mathrm{Free}(\phi) \subseteq V$. Sei $s \in T_{\Sigma_V}$ ein Baum. Für eine Individuenvariable x definiert die Formel ϕ eine partielle Funktion $\phi_{V,x}^{en} : T_{\Sigma_V} \dashrightarrow [\mathbb{N}^* \dashrightarrow K]$ mit $\phi_{V,x}^{en}(s)(w) := ([\![\phi]\!]_{V\cup\{x\}}^{en}, s[x \to w])$ für alle $w \in \mathrm{pos}(s)$. Das Bild eines Baumes s unter $\phi_{V,x}^{en}$ ist also ein Gewichtsbaum.

Definition 4.2.2. Seien $\phi, \psi \in \mathrm{wMSO}(K,\Sigma)$ Formeln, $k \in K$ sowie x, y Variablen erster Stufe und X eine Variable zweiter Stufe. Weiterhin sei V eine Variablenmenge, mit $\mathrm{Free}(\phi) \cup \mathrm{Free}(\psi) \subseteq V$, $s \in T_{\Sigma_V}$ und en eine Positionsaufzählung für Gewichtsbäume.

Ist $s = (t, \rho) \in T_{\Sigma_V}$ nicht korrekt, so sei $(\llbracket \phi \rrbracket_V^{en}, s) := 0$. Anderenfalls definieren wir die Semantik einer Formel induktiv über den Formelaufbau.

$$\begin{aligned}
(\llbracket k \rrbracket_V^{en}, s) &:= k \ , \\
(\llbracket label_\sigma(x) \rrbracket_V^{en}, (t, \rho)) &:= \begin{cases} 1 & \text{falls } \rho(x) = \sigma \ , \\ 0 & \text{sonst} \ , \end{cases} \\
(\llbracket edge_i(x, y) \rrbracket_V^{en}, (t, \rho)) &:= \begin{cases} 1 & \text{falls } \rho(y) = \rho(x).i \ , \\ 0 & \text{sonst} \ , \end{cases} \\
(\llbracket \neg\phi \rrbracket_V^{en}, s) &:= \overline{(\llbracket \phi \rrbracket_V^{en}, s)} \ , \\
(\llbracket \phi \wedge \psi \rrbracket_V^{en}, s) &:= (\llbracket \phi \rrbracket_V^{en}, s) \cdot (\llbracket \psi \rrbracket_V^{en}, s) \ , \\
(\llbracket \phi \vee \psi \rrbracket_V^{en}, s) &:= (\llbracket \phi \rrbracket_V^{en}, s) + (\llbracket \psi \rrbracket_V^{en}, s) \ , \\
(\llbracket \forall x.\phi \rrbracket_V^{en}, s) &:= \prod_{i=1}^{|\mathrm{pos}(s)|} (\llbracket \phi \rrbracket_{V \cup \{x\}}^{en}, s[x \to en(\phi_{V,x}^{en}(s))(i)]) \ , \\
(\llbracket \forall X.\phi \rrbracket_V^{en}, s) &:= \prod_{I \subseteq \mathrm{pos}(s)} (\llbracket \phi \rrbracket_{V \cup \{X\}}^{en}, s[X \to I]) \ , \\
(\llbracket \exists x.\phi \rrbracket_V^{en}, s) &:= \sum_{w \in \mathrm{pos}(s)} (\llbracket \phi \rrbracket_{V \cup \{x\}}^{en}, s[x \to w]) \text{ und} \\
(\llbracket \exists X.\phi \rrbracket_V^{en}, s) &:= \sum_{I \subseteq \mathrm{pos}(s)} (\llbracket \phi \rrbracket_{V \cup \{x\}}^{en}, s[X \to I]) \ .
\end{aligned}$$

Zur Bestimmung von $(\llbracket \forall X.\phi \rrbracket_V^{en}, s)$ fixieren wir eine lineare Ordnung auf $\wp(\mathrm{pos}(s))$.

Sei $s \in T_{\Sigma_V}$ ein nicht notwendigerweise korrekter Baum. Wir beobachten, dass die folgenden Gleichungen gelten:

$$\begin{aligned}
\llbracket k \rrbracket_V^{en} &= \tilde{k} \odot \mathbb{1}_{T_{\Sigma_V}^v} \ , \\
\llbracket \phi \wedge \psi \rrbracket_V^{en} &= \llbracket \phi \rrbracket_V^{en} \odot \llbracket \psi \rrbracket_V^{en} \text{ und} \\
\llbracket \phi \vee \psi \rrbracket_V^{en} &= \llbracket \phi \rrbracket_V^{en} + \llbracket \psi \rrbracket_V^{en} \ .
\end{aligned}$$

Sei $X \in V$ eine Mengenvariable. Ist s nicht korrekt, so ist $s[X \to I]$ für beliebige $I \subseteq \mathrm{pos}(s)$ ebenfalls nicht korrekt und umgekehrt. Damit gilt auch für nicht korrekte Bäume s:

$$\begin{aligned}
(\llbracket \forall X.\phi \rrbracket_V^{en}, s) &= \prod_{I \subseteq \mathrm{pos}(s)} (\llbracket \phi \rrbracket_{V \cup \{X\}}^{en}, s[X \to I]) \text{ und} \\
(\llbracket \exists X.\phi \rrbracket_V^{en}, s) &= \sum_{I \subseteq \mathrm{pos}(s)} (\llbracket \phi \rrbracket_{V \cup \{x\}}^{en}, s[X \to I]) \ .
\end{aligned}$$

Sei $x \in V$ eine Individuenvariable. Ist s nicht korrekt, weil $f(x) = 0$ in allen Beschriftungen $(\sigma, f) \in \mathrm{img}(s)$, so sind alle Bäume $s[x \to w]$ mit $w \in \mathrm{pos}(s)$ korrekt. Deshalb können wir die obige Beobachtung von $\forall X.\phi$ und $\exists X.\phi$ nicht auf $\forall x.\phi$ und $\exists x.\phi$ übertragen.

Betrachten wir noch einmal die Sprache wMSO. Es handelt sich hierbei um eine nichtklassische logische Sprache mit den Prädikatensymbolen $label_\sigma(x)$ und $edge_i(x, y)$

für $\sigma \in \Sigma$ und $i \in [1, max_\Sigma]$, einer (im Allgemeinen nicht endlichen) Menge von Konstanten K sowie den Junktoren $\neg, \wedge$ und $\vee$. Außerdem nutzt wMSO die Quantoren $\forall$ und $\exists$ sowohl für Individuen- als auch Mengenvariablen. Auf ein Bimonoid K bezogen, ist wMSO somit eine mehrwertige Logik [22]. Ein Baum $t \in T_\Sigma$ entspricht dann einer Interpretation von wMSO mit dem Individuenbereich $\mathrm{pos}(t)$ und den Relationen $label_\sigma$ und $edge_i$. Ein korrekter Baum $(t, \rho) \in T^v_{\Sigma_V}$ ist damit eine konkrete Situation, das heißt eine Interpretation mit einer Variablenbelegung. Sei $\phi \in$ wMSO. Die Semantik $[\![\phi]\!]^{en}_V$ ist demnach die Auswertung von ϕ in allen Interpretationen, welche Bäume aus T_Σ sind.

Somit ergeben sich aus der Nichtklassischen Logik viele natürliche Beispiele für Bimonoide und deren Anwendungen in Logik, Linguistik, Philosophie und Technik.

Beispiel 4.2.3. Betrachten wir die Mengen $W_m := \{k/(m-1) \mid k \in [0, m-1]\}$ für $m \in \mathbb{N}$, $m > 1$ und $W_\infty := [0,1] \subset \mathbb{R}$. Die *Gödel-Logik* [21] ist definiert in der Struktur $G_m := (W_m, \max, \min, \bar{}, \supset, 0, 1)$ beziehungsweise $G_\infty := (W_\infty, \max, \min, \bar{}, \supset, 0, 1)$, wobei

$$\bar{k} := \begin{cases} 1 & \text{falls } k = 0 \\ 0 & \text{sonst} \end{cases} \qquad \text{und} \qquad k_1 \supset k_2 := \begin{cases} 1 & \text{falls } k_1 \leq k_2 \\ k_2 & \text{sonst} \end{cases} .$$

Die Logiken $\mathrm{wMSO}(G_m, \Sigma)$ beziehungsweise $\mathrm{wMSO}(G_\infty, \Sigma)$ sind somit Fragmente der Gödel-Logik ohne Implikation $\supset$.

Beispiel 4.2.4. Sei W_∞ das oben definierte reelle Intervall $[0,1]$. Eine *T-Norm* ist eine zweistellige Funktion $T : W_\infty \times W_\infty \to W_\infty$, die assoziativ, kommutativ, nicht vermindernd ist und für welche 1 das neutrale Element darstellt, das heißt, für alle $k_1, k_2, k_3 \in W_\infty$ gilt:

- $T(k_1, T(k_2, k_3) = T(T(k_1, k_2), k_3)$,
- $T(k_1, k_2) = T(k_2, k_1)$,
- $k_1 \leq k_2 \implies T(k_1, k_3) \leq T(k_2, k_3)$ und
- $T(k_1, 1) = k_1$.

Eine *T-Norm Logik* [24] ist eine auf einer gewählten T-Norm aufbauende *Fuzzy-Logik*. Die starke Konjunktion bezeichnet man mit $\&_T$, wobei $k_1 \&_T k_2 := T(k_1, k_2)$. Die Standardimplikation ist durch $k_1 \supset_T k_2 := \sup\{k \in W_\infty \mid T(k_1, k) \leq k_2\}$ definiert. Diese bestimmt nun die Negation $\overline{k_1}^T := \phi \supset 0$ und die Disjunktion sei $\max(k_1, k_2)$. Somit ist die Struktur $T := (W_\infty, \max, \&_T, \bar{}^T, \supset_T, 0, 1)$ ein starkes Bimonoid mit Negation. Also kann man $\mathrm{wMSO}(T, \Sigma)$ als eine prädikatenlogische Erweiterung einer T-Norm-Aussagenlogik betrachten.

Beispiel 4.2.5. Die vierwertige *Belnap-Logik* [1] hat eine wichtige Bedeutung im Hardwaredesign, bei Datenbanksystemen und als Relevanzlogik, weil sie parakonsistentes Schließen ermöglicht. Ihre Struktur $(W^*, \sup, \inf, \bar{}, \{\top\}, \{\bot\})$ ist definiert durch den Verband in Abbildung 4.1 über der Menge $W^* := \{\emptyset, \{\top\}, \{\bot\}, \{\top, \bot\}\}$.

Die Negation ist für alle $k \in W^*$ bestimmt mit

$$\bar{k} := \begin{cases} \{\top\} & \text{falls } k = \{\bot\} \ , \\ \{\bot\} & \text{falls } k = \{\top\} \text{ und} \\ k & \text{sonst} \ . \end{cases}$$

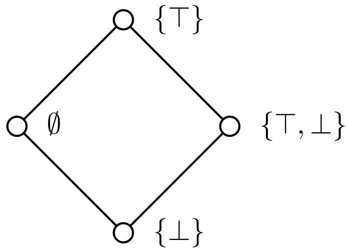


Abbildung 4.1: Belnap-Verband

Die Logik wMSO(W^*, Σ) ist somit eine auf der Belnap-Aussagenlogik aufbauende Prädikatenlogik.

Als Nächstes wollen wir die Verbindung zwischen ungewichteter monadischer Logik zweiter Stufe und der gewichteten monadischen Logik zweiter Stufe untersuchen.

Lemma 4.2.6. *Sei $\phi \in$ wMSO(K, Σ) eine boolesche Formel, V eine endliche Variablenmenge mit* Free$(\phi) \subseteq V$ *und en eine Positionsaufzählung für Gewichtsbäume. Dann gilt:*

$$[\![\phi]\!]_V^{en} = \mathbb{1}_{L_V(\phi)} \,.$$

Beweis. Induktion über den Formelaufbau:

$$[\![0]\!]_V^{en} = \tilde{0} = \mathbb{1}_{L_V(0)}$$

$$[\![1]\!]_V^{en} = \tilde{1} \odot \mathbb{1}_{T^v_{\Sigma_V}} = \mathbb{1}_{L_V(1)}$$

Sei $s \in T_{\Sigma_V}$. Angenommen s ist nicht korrekt, dann ist $([\![\phi]\!]_V^{en}, s) = 0$ und $(t, \rho) \not\models \phi$. Sei s nun also korrekt und entspreche dem Paar (t, ρ).

$$\begin{aligned}&\phi \equiv label_\sigma(x) :\\ &([\![\phi]\!]_V^{en}, s) = 1 \iff t(\rho(x)) = \sigma \iff (t, \rho) \models \phi\end{aligned}$$

$$\begin{aligned}&\phi \equiv edge_i(x, y) :\\ &([\![\phi]\!]_V^{en}, s) = 1 \iff \rho(y) = \rho(x).i \iff (t, \rho) \models \phi\end{aligned}$$

$$\begin{aligned}&\phi \equiv x \in X :\\ &([\![\phi]\!]_V^{en}, s) = 1 \iff \rho(x) \in \rho(X) \iff (t, \rho) \models \phi\end{aligned}$$

Gelte nun die Aussage für die booleschen Formeln $\psi, \chi \in$ wMSO(K, Σ).

$$\begin{aligned}&\phi \equiv \neg\psi :\\ &([\![\phi]\!]_V^{en}, s) = 1 \iff ([\![\psi]\!]_V^{en}, s) = 0\\ &\qquad\qquad\qquad\iff (t, \rho) \not\models \psi \qquad \text{(Voraussetzung)}\\ &\qquad\qquad\qquad\iff (t, \rho) \models \phi\end{aligned}$$

$\phi \equiv \psi \wedge \chi:$

$$\begin{aligned}
([\![\phi]\!]^{en}_V, s) = 1 &\iff ([\![\psi]\!]^{en}_V, s) \cdot ([\![\chi]\!]^{en}_V, s) = 1 \\
&\iff ([\![\psi]\!]^{en}_V, s) = 1 \quad \text{und} \quad ([\![\chi]\!]^{en}_V, s) = 1 \\
&\iff (t, \rho) \models \psi \quad \text{und} \quad (t, \rho) \models \chi \qquad \text{(Voraussetzung)} \\
&\iff (t, \rho) \models \phi
\end{aligned}$$

$\phi \equiv \forall x.\psi: \quad [\![\psi]\!]^{en}_{V\cup\{x\}} = \mathbb{1}_{L_{V\cup\{x\}}(\psi)} \implies$ Reihenfolge der Faktoren ist beliebig

$$\begin{aligned}
([\![\phi]\!]^{en}_V, s) = 1 &\iff \prod_{i=1}^{|\mathrm{pos}(s)|} ([\![\psi]\!]^{en}_{V\cup\{x\}}, s[x \to en(\phi^{en}_{V,x}(s))(i)]) = 1 \\
&\iff \forall w \in \mathrm{pos}(s) : ([\![\psi]\!]^{en}_{V\cup\{x\}}, s[x \to w]) = 1 \\
&\iff \forall w \in \mathrm{pos}(t) : (t, \rho[x \to w]) \models \psi \qquad \text{(Voraussetzung)} \\
&\iff (t, \rho) \models \phi
\end{aligned}$$

$\phi \equiv \forall X.\psi: \quad [\![\psi]\!]^{en}_{V\cup\{X\}} = \mathbb{1}_{L_{V\cup\{X\}}(\psi)} \implies$ Reihenfolge der Faktoren ist beliebig

$$\begin{aligned}
([\![\phi]\!]^{en}_V, s) = 1 &\iff \prod_{I \subseteq \mathrm{pos}(s)} ([\![\psi]\!]^{en}_{V\cup\{X\}}, s[X \to I]) = 1 \\
&\iff \forall I \subseteq \mathrm{pos}(s) : ([\![\psi]\!]^{en}_{V\cup\{X\}}, s[X \to I]) = 1 \\
&\iff \forall I \subseteq \mathrm{pos}(t) : (t, \rho[X \to I]) \models \psi \qquad \text{(Voraussetzung)} \\
&\iff (t, \rho) \models \phi
\end{aligned}$$

□

Da für boolesche Formeln $\phi \in$ wMSO und jede Positionsaufzählung für Gewichtsbäume en die Potenzreihe $[\![\phi]\!]^{en}_V$ stets die charakteristische Funktion $\mathbb{1}_{L_V(\phi)}$ ist, schreiben wir von nun an verkürzend $[\![\phi]\!]_V$ statt $[\![\phi]\!]^{en}_V$ und $\phi_{V,x}$ anstelle von $\phi^{en}_{V,x}$.

Aus Satz 4.1.1 (der Satz von Büchi für Baumsprachen) und Proposition 3.3.1 folgt außerdem, dass die Semantik $[\![\phi]\!]_V$ jeder booleschen Formel $\phi \in \mathrm{wMSO}(K, \Sigma)$ eine erkennbare Stufenfunktion ist.

Satz 4.2.7. *Eine Baumsprache $L \subseteq T_\Sigma$ ist genau dann MSO-definierbar, wenn es einen booleschen Satz $\phi \in \mathrm{wMSO}(K, \Sigma)$ gibt mit $\mathbb{1}_L = [\![\phi]\!]_{\mathrm{Free}(\phi)}$.*

Beweis. ($\Rightarrow$) Sei L MSO-definierbar, dann gibt es einen Satz $\phi \in \mathrm{MSO}(\Sigma)$ (ohne Abkürzungen) mit $L(\phi) = L$. Dieser Satz ist ein boolescher Satz in $\mathrm{wMSO}(\Sigma, K)$. Also gilt nach Lemma 4.2.6 $[\![\phi]\!]_{\mathrm{Free}(\phi)} = \mathbb{1}_{L(\phi)} = \mathbb{1}_L$.

($\Leftarrow$) Sei L eine Baumsprache und $\phi \in \mathrm{wMSO}(K, \Sigma)$ ein boolescher Satz (ohne Abkürzungen) mit $[\![\phi]\!]_{\mathrm{Free}(\phi)} = \mathbb{1}_L$. Dann ist ϕ ebenfalls ein Satz aus $\mathrm{MSO}(\Sigma)$. Aus der Definition der Semantik und Lemma 4.2.6 folgt nun $\mathbb{1}_{L(\phi)} = [\![\phi]\!]_{\mathrm{Free}(\phi)} = \mathbb{1}_L$. Also ist L MSO-definierbar. □

Lemma 4.2.8. *Sei* $\phi \in \mathrm{wMSO}(K, \Sigma)$*,* V *eine endliche Variablenmenge mit* $\mathrm{Free}(\phi) \subseteq V$ *und en eine Positionsaufzählung für Gewichtsbäume. Dann gilt für jede Variablenmenge* W *mit* $\mathrm{Free}(\phi) \subseteq W \subset V$ *und alle Bäume* $(t, \rho) \in T^v_{\Sigma_V}$*:*

$$(\llbracket\phi\rrbracket^{en}_V, (t, \rho)) = (\llbracket\phi\rrbracket^{en}_W, (t, \rho|_W))$$

und insbesondere also

$$(\llbracket\phi\rrbracket^{en}_V, (t, \rho)) = \left(\llbracket\phi\rrbracket^{en}_W, (t, \rho|_{\mathrm{Free}(\phi)})\right) .$$

Beweis. Der Beweis erfolgt induktiv über den Formelaufbau. Sei $k \in K$, $(t, \rho) \in T^v_{\Sigma_V}$ und $\phi \in \mathrm{wMSO}(K, \Sigma)$ atomar.

$$\begin{aligned}&\phi \equiv k :\\ &(\llbracket k\rrbracket^{en}_V, (t, \rho)) = k = (\llbracket k\rrbracket^{en}_W, (t, \rho|_W))\end{aligned}$$

Ist ϕ boolesch, so gilt nach Definition der Semantik $\mathrm{img}(\llbracket\phi\rrbracket_V) \subseteq \{0, 1\}$.

$$\begin{aligned}&\phi \equiv label_\sigma(x) :\\ &(\llbracket\phi\rrbracket_V, (t, \rho)) = 1 \iff t(\rho(x)) = \sigma = t(\rho|_W(x))\\ &\qquad\qquad\qquad\quad\;\; \iff (\llbracket\phi\rrbracket_W, (t, \rho|_W)) = 1\end{aligned}$$

$$\begin{aligned}&\phi \equiv edge_i(x, y) :\\ &(\llbracket\phi\rrbracket_V, (t, \rho)) = 1 \iff \rho(y) = \rho(x).i\\ &\qquad\qquad\qquad\quad\;\; \iff \rho|_W(y) = \rho|_W(x).i\\ &\qquad\qquad\qquad\quad\;\; \iff (\llbracket\phi\rrbracket_W, (t, \rho|_W)) = 1\end{aligned}$$

$$\begin{aligned}&\phi \equiv x \in X :\\ &(\llbracket\phi\rrbracket_V, (t, \rho)) = 1 \iff \rho(x) \in \rho(X)\\ &\qquad\qquad\qquad\quad\;\; \iff \rho|_W(x) \in \rho|_W(X)\\ &\qquad\qquad\qquad\quad\;\; \iff (\llbracket\phi\rrbracket_W, (t, \rho|_W)) = 1\end{aligned}$$

Gelte nun die Aussage für beliebige Formeln ψ und χ. Sei $x \in V$ eine Individuenvariable und $X \in V$ eine Mengenvariable.

$$\begin{aligned}&\phi \equiv \neg\psi :\\ &(\llbracket\phi\rrbracket^{en}_V, (t, \rho)) = \overline{(\llbracket\psi\rrbracket^{en}_V, (t, \rho))}\\ &\qquad\qquad\qquad\;\; = \overline{(\llbracket\psi\rrbracket^{en}_W, (t, \rho|_W))}\\ &\qquad\qquad\qquad\;\; = (\llbracket\phi\rrbracket^{en}_W, (t, \rho|_W))\end{aligned}$$

$$\begin{aligned}&\phi \equiv \psi \wedge \chi :\\ &(\llbracket\phi\rrbracket^{en}_V, (t, \rho)) = (\llbracket\psi\rrbracket^{en}_V, (t, \rho)) \cdot (\llbracket\chi\rrbracket^{en}_V, (t, \rho))\\ &\qquad\qquad\qquad\;\; = (\llbracket\psi\rrbracket^{en}_W, (t, \rho|_W)) \cdot (\llbracket\chi\rrbracket^{en}_W, (t, \rho|_W))\\ &\qquad\qquad\qquad\;\; = (\llbracket\phi\rrbracket^{en}_W, (t, \rho|_W))\end{aligned}$$

$\phi \equiv \psi \vee \chi:$

$$\begin{aligned}
([\![\phi]\!]^{en}_V, (t,\rho)) &= ([\![\psi]\!]^{en}_V, (t,\rho)) + ([\![\chi]\!]^{en}_V, (t,\rho))\\
&= ([\![\psi]\!]^{en}_W, (t,\rho|_W)) + ([\![\chi]\!]^{en}_W, (t,\rho|_W))\\
&= ([\![\phi]\!]^{en}_W, (t,\rho|_W))
\end{aligned}$$

$\phi \equiv \forall x.\psi:$

$$\begin{aligned}
([\![\phi]\!]^{en}_V, (t,\rho)) &= \prod_{i=1}^{|\mathrm{pos}(t)|} ([\![\psi]\!]^{en}_{V\cup\{x\}}, (t,\rho[x \to en(\phi^{en}_{V,x}(s))(i)]))\\
&\qquad \text{mit der Induktionsvoraussetzung folgt}\\
&= \prod_{i=1}^{|\mathrm{pos}(t)|} ([\![\psi]\!]^{en}_{W\cup\{x\}}, (t,\rho[x \to en(\phi^{en}_{V,x}(s))(i)]|_{W\cup\{x\}}))\\
&= \prod_{i=1}^{|\mathrm{pos}(t)|} ([\![\psi]\!]^{en}_{W\cup\{x\}}, (t,\rho|_W[x \to en(\phi^{en}_{V,x}(s))(i)]))\\
&= ([\![\phi]\!]^{en}_W, (t,\rho|_W))
\end{aligned}$$

$\phi \equiv \forall X.\psi:$

$$\begin{aligned}
([\![\phi]\!]^{en}_V, (t,\rho)) &= \prod_{I\subseteq \mathrm{pos}(s)} ([\![\psi]\!]^{en}_{V\cup\{X\}}, (t,\rho[x \to I]))\\
&\qquad \text{mit der Induktionsvoraussetzung folgt}\\
&= \prod_{I\subseteq \mathrm{pos}(s)} ([\![\psi]\!]^{en}_{W\cup\{X\}}, (t,\rho[x \to I]|_{W\cup\{X\}}))\\
&= \prod_{I\subseteq \mathrm{pos}(s)} ([\![\psi]\!]^{en}_{W\cup\{X\}}, (t,\rho|_W[x \to I]))\\
&= ([\![\phi]\!]^{en}_W, (t,\rho|_W))
\end{aligned}$$

$\phi \equiv \exists x.\psi:$

$$\begin{aligned}
([\![\phi]\!]^{en}_V, (t,\rho)) &= \sum_{w\in \mathrm{pos}(t)} ([\![\psi]\!]^{en}_{V\cup\{x\}}, (t,\rho[x \to w]))\\
&\qquad \text{mit der Induktionsvoraussetzung folgt}\\
&= \sum_{w\in \mathrm{pos}(t)} ([\![\psi]\!]^{en}_{W\cup\{x\}}, (t,\rho[x \to w]|_{W\cup\{x\}}))\\
&= \sum_{w\in \mathrm{pos}(t)} ([\![\psi]\!]^{en}_{W\cup\{x\}}, (t,\rho|_W[x \to w]))\\
&= ([\![\phi]\!]^{en}_W, (t,\rho|_W))
\end{aligned}$$

$\phi \equiv \exists X.\psi$:

$$
\begin{aligned}
([\![\phi]\!]_V^{en}, (t,\rho)) &= \sum_{I \subset \mathrm{pos}(t)} ([\![\psi]\!]_{V\cup\{X\}}^{en}, (t, \rho[X \to I])) \\
&\qquad \text{mit der Induktionsvoraussetzung folgt} \\
&= \sum_{I \subset \mathrm{pos}(t)} ([\![\psi]\!]_{W\cup\{X\}}^{en}, (t, \rho[X \to I]|_{W\cup\{X\}})) \\
&= \sum_{I \subset \mathrm{pos}(t)} ([\![\psi]\!]_{W\cup\{X\}}^{en}, (t, \rho|_W[X \to I])) \\
&= ([\![\phi]\!]_W^{en}, (t, \rho|_W))
\end{aligned}
$$

□

Mithilfe dieser letzten Aussage folgern wir nun, dass es nicht von der gewählten Variablenmenge abhängt, ob die Semantik einer Formel eine erkennbare Stufenfunktion ist oder nicht. Wir nennen eine erkennbare Stufenfunktion $\sum_{i=0}^{n} k_i \cdot \mathbb{1}_{L_i}$ *normalisiert*, falls $k_0 = 0$ gilt.. Da dies nur eine Umbenennung darstellt, handelt es ich um keine Einschränkung. Eine normalisierte Stufenfunktion sammelt also insbesondere alle nicht korrekten Bäume in L_0; es gilt $T_{\Sigma_V} \setminus T_{\Sigma_V}^v \subseteq L_0$.

Lemma 4.2.9. *Sei $\phi \in \mathrm{wMSO}(K,\Sigma)$ und V, W Mengen von Variablen derart, dass $\mathrm{Free}(\phi) \subseteq W \subset V$, sowie en eine Positionsaufzählung für Gewichtsbäume. Dann gilt: $[\![\phi]\!]_V^{en}$ ist genau dann eine erkennbare Stufenfunktion, wenn $[\![\phi]\!]_W^{en}$ eine erkennbare Stufenfunktion ist.*

Beweis. Sei $\tau : \Sigma_V \to \Sigma_W$ die surjektive Umbenennung mit $\tau((\sigma, f)) := (\sigma, f|_W)$ für alle $(\sigma, f) \in \Sigma_V$.

($\Rightarrow$) Sei $[\![\phi]\!]_V^{en} = \sum_{i=0}^{n} k_i \cdot \mathbb{1}_{L_i}$ eine normalisierte erkennbare Stufenfunktion, wobei $k_i \in K$ und $L_i \subseteq T_{\Sigma_V}$ für alle $i \in [0,n]$. Wir zeigen $[\![\phi]\!]_W^{en} = S := \sum_{i=0}^{n} k_i \cdot \mathbb{1}_{\tau(L_i)}$.

Aus Lemma 4.2.8 folgt $([\![\phi]\!]_V^{en}, s') = ([\![\phi]\!]_V^{en}, s'')$ für alle $s', s'' \in \tau^{-1}(s) \cap T_{\Sigma_V}^v$ mit $s \in T_{\Sigma_W}$. Da $k_i \neq k_j$ für $i \neq j \in [0,n]$, gilt für alle $i \in [0,n]$ und $s \in \tau(L_i)$ somit $\tau^{-1}(s) \cap T_{\Sigma_V}^v \subseteq L_i$ und $\tau^{-1}(s) \cap (T_{\Sigma_V} \setminus T_{\Sigma_V}^v) \subseteq L_0$. Also sind $\tau(L_1), \ldots, \tau(L_n)$ paarweise disjunkt.
Sei $s \in T_{\Sigma_W}$ korrekt. Dann existiert ein korrekter Baum $s' \in \tau^{-1}(s)$ und ein $i \in [0,n]$, sodass $s' \in L_i$. Also ist nach Lemma 4.2.8 $([\![\phi]\!]_W^{en}, s) = ([\![\phi]\!]_V^{en}, s') = k_i$. Da $s \in \tau(L_i)$ und $\tau(L_1), \ldots, \tau(L_n)$ paarweise disjunkt sind, ist auch $(S, s) = k_i$.

Sei s nicht korrekt. Dann ist $([\![\phi]\!]_W^{en}, s) = 0$. Kein Baum aus $\tau^{-1}(s)$ ist korrekt, also ist $\tau^{-1}(s) \subseteq L_0$ und $([\![\phi]\!]_V^{en}, s') = k_0 = 0$ für alle $s' \in \tau^{-1}(s)$. Folglich gilt $s \in \tau(L_0)$ und $s \notin \tau(L_i)$ für alle $i \in [1,n]$ und damit $(S, s) = k_0 = 0$. Somit gilt $[\![\phi]\!]_W^{en} = \sum_{i=0}^{n} k_i \cdot \mathbb{1}_{\tau(L_i)}$.
Es ist möglich, dass ein Baum $\tau(s)$ korrekt ist, obwohl $s \in T_{\Sigma_V}$ nicht korrekt ist. Dann kann es eine Baumsprache L_j mit $j \in [1,n]$ geben, sodass $\tau(s) \in \tau(L_0)$ und $\tau(s) \in \tau(L_j)$. Da aber alle Baumsprachen $\tau(L_i)$ mit $i \in [0,n]$ nach Lemma 3.3.6 erkennbar sind und die erkennbaren Baumsprachen unter den booleschen Operationen abgeschlossen sind, ist $[\![\phi]\!]_W^{en}$ eine erkennbare Stufenfunktion.

($\Leftarrow$) Sei $[\![\phi]\!]_W^{en} = \sum_{i=0}^{n} k_i \cdot \mathbb{1}_{L_i}$ eine normalisierte erkennbare Stufenfunktion mit $k_i \in K$ und $L_i \subseteq T_{\Sigma_W}$ für $i \in [0,n]$. Nach Proposition 3.3.5 ist $S := \sum_{i=0}^{n} k_i \cdot \mathbb{1}_{\tau^{-1}(L_i)} \odot \mathbb{1}_{T_{\Sigma_V}^v}$ eine erkennbare Stufenfunktion. Wir zeigen nun $[\![\phi]\!]_V^{en} = S$.

Angenommen $s \in T_{\Sigma_V}$ ist korrekt. Dann gibt es nach Lemma 4.2.8 genau ein $s' \in \tau(s)$ und genau ein $i \in [0,n]$ mit $(\llbracket\phi\rrbracket_V^{en}, s) = (\llbracket\phi\rrbracket_W^{en}, s') = k_i$. Folglich ist $s \in \phi^{-1}(L_i)$ und damit $(S,s) = k_i$. Angenommen s ist nicht korrekt. Dann ist $(\llbracket\phi\rrbracket_V^{en}, s) = 0 = (S,s)$. □

Ab sofort werden wir für $\llbracket\phi\rrbracket_{\mathrm{Free}(\phi)}^{en}$ auch die Abkürzung $\llbracket\phi\rrbracket^{en}$ benutzen. Analog dazu schreiben wir ϕ_x für die partielle Funktion $\phi_{\mathrm{Free}(\phi)\cup\{x\}}$.

Lemma 4.2.10. *Sei $\phi \in \mathrm{wMSO}(K,\Sigma)$ und V eine Variablenmenge mit $\mathrm{Free}(\phi) \subseteq V$. Seien weiterhin en_1 und en_2 Positionsaufzählungen für Gewichtsbäume. Dann gilt: $\llbracket\phi\rrbracket_V^{en_1}$ ist genau dann (schwach) en_2-erkennbar, wenn $\llbracket\phi\rrbracket^{en_1}$ (schwach) en_2-erkennbar ist.*

Beweis. Wir verwenden hier die Idee aus [17, proof of Lemma 4.7].

Sei $\tau : \Sigma_V \to \Sigma_{\mathrm{Free}(\phi)}$ diejenige Umbenennung, die für alle $(\sigma, f) \in \Sigma_V$ durch $\tau((s,f)) := (s, f|_{\mathrm{Free}(\phi)})$ definiert ist.

($\Rightarrow$) Wir fixieren die erkennbare Baumsprache $F := L_V(\phi_F) \subseteq T_{\Sigma_V}^v$ mit:

$$\phi_F := \bigwedge_{\substack{x \in V\setminus\mathrm{Free}(\phi)\\ \text{erste Stufe}}} root(x) \ \wedge \bigwedge_{\substack{X \in V\setminus\mathrm{Free}(\phi)\\ \text{zweite Stufe}}} (\exists x.(x \in X) \ \wedge \ \forall x.(x \in X \to root(x))) \ ,$$

$$root(x) := \forall y.(\neg edge_1(y,x) \ \wedge \ \ldots \ \wedge \ \neg edge_{max_\Sigma}(y,x)) \ .$$

F enthält genau diejenigen Bäume, die alle Individuenvariablen x und alle Mengenvariablen X, welche nicht frei in ϕ vorkommen, mit der Wurzel belegen.

Es gibt nun zu jedem Baum $(t,\rho') \in T_{\Sigma_{\mathrm{Free}(\phi)}}^v$ genau einen Baum $(t,\rho) \in F$ mit $(t,\rho') \in \tau((t,\rho))$. Außerdem besagt Lemma 4.2.8, dass für alle $(t,\rho) \in T_{\Sigma_V}^v$ die Gleichung $(\llbracket\phi\rrbracket_V^{en_1}, (t,\rho)) = (\llbracket\phi\rrbracket^{en_1}, (t, \rho|_{\mathrm{Free}(\phi)}))$ gilt. Damit ist $\llbracket\phi\rrbracket^{en_1} = \tau(\llbracket\phi\rrbracket_V^{en_1} \odot \mathbb{1}_F)$, wobei F die Urbildmenge einschränkt. Ist $\llbracket\phi\rrbracket_V^{en_1}$ (schwach) en_2-erkennbar, so ist $\llbracket\phi\rrbracket^{en_1}$ nach Lemma 3.3.6 ebenfalls (schwach) en_2-erkennbar.

($\Leftarrow$) Sei nun $\llbracket\phi\rrbracket^{en_1}$ (schwach) en_2-erkennbar und $\tau^{-1} : \Sigma_{\mathrm{Free}(\phi)} \to \wp(\Sigma_V)$ die Umbenennung mit $\tau^{-1}(s) := \{s' \in T_{\Sigma_V} \mid s \in \tau(s')\}$ für alle $s \in T_{\Sigma_{\mathrm{Free}(\phi)}}$. Wiederum mit Lemma 4.2.8 gilt nun $\llbracket\phi\rrbracket_V^{en_1} = \tau^{-1}(\llbracket\phi\rrbracket^{en_1}) \odot \mathbb{1}_{T_{\Sigma_V}^v}$. Die charakteristische Funktion $\mathbb{1}_{T_{\Sigma_V}^v}$ ist (schwach) en_2-erkennbar und $\tau^{-1}(\llbracket\phi\rrbracket^{en_1})$ nach Lemma 3.3.6 ebenso. Deshalb ist auch $\llbracket\phi\rrbracket_V^{en_1}$ (schwach) en_2-erkennbar. □

Die sofortige Konsequenz dieses Lemmas ist, dass für die Erkennbarkeit der Semantik einer Formel $\phi \in \mathrm{wMSO}(K,\Sigma)$ die zugrunde liegende Variablenmenge unerheblich ist, solange sie $\mathrm{Free}(\phi)$ umfasst. Das heißt, es gilt für beliebige Variablenmengen V und W mit $V \supseteq \mathrm{Free}(\phi) \subseteq W$ und Positionsaufzählungen für Gewichtsbäume en_1, dass $\llbracket\phi\rrbracket_V^{en_1}$ genau dann mit einer Positionsaufzählung en_2 erkennbar ist, wenn $\llbracket\phi\rrbracket_W^{en_1}$ en_2-erkennbar ist.

Wir haben ebenfalls gesehen, dass im obigen Lemma die Positionsaufzählung, welche die Semantik bestimmt, nicht unmittelbar mit der Positionsaufzählung der Erkennbarkeit zusammenhängt. Im nächsten Abschnitt werden uns weitere Beispiele dieser Art begegnen.

Sind im Folgenden sowohl das Bimonoid K wie auch das Rangalphabet Σ fixiert, so schreiben wir wMSO anstelle von $\mathrm{wMSO}(K,\Sigma)$. Das Fragment der *existenziellen Formeln von* $\mathrm{wMSO}(K,\Sigma)$ enthält alle Formeln der Form ψ und $\exists X_1 \ldots \exists X_n.\psi$ mit $n \in \mathbb{N}$, wobei ψ keine weiteren Quantifizierungen zweiter Stufe enthält. Wir bezeichnen es mit $\mathrm{wEMSO}(K,\Sigma)$ und schreiben verkürzend wEMSO.

Seien $\phi, \psi \in$ wMSO, wobei $[\![\phi]\!]^{en}$ und $[\![\psi]\!]^{en}$ *en*-erkennbar sind. Da das Hadamard-Produkt *en*-erkennbarer Baumreihen im Allgemeinen nicht *en*-erkennbar ist, ist auch die Semantik der Konjunktion $[\![\phi \wedge \psi]\!]^{en}$ im Allgemeinen nicht *en*-erkennbar. Wir suchen deshalb Fragmente von wMSO, deren Formeln ϕ stets eine *en*-erkennbare Semantik $[\![\phi]\!]^{en}$ haben und die somit $K^{en-rec}\langle\langle T_\Sigma \rangle\rangle$ charakterisieren.

Definition 4.2.11. Eine Formel $\phi \in$ wMSO(K, Σ) heißt *gestuft*, falls

- ϕ boolesch oder konstant ist oder
- ϕ Negation, Konjunktion oder Disjunktion gestufter Formeln ist.

Die folgende Grammatik erzeugt genau die gestuften Formeln, wobei $k \in K$:

$$\phi_{step} := k \mid \phi_{bool} \mid \neg\phi_{step} \mid \phi_{step} \wedge \phi_{step} \mid \phi_{step} \vee \phi_{step} \ .$$

Definition 4.2.12. Eine Formel $\phi \in$ wMSO(K, Σ) heißt *eingeschränkt*, falls

- ϕ gestuft ist

oder

- in allen Teilformeln $\forall x.\psi$ (x Individuenvariable) ψ gestuft ist und
- in allen Teilformeln $\psi \wedge \chi$, welche nicht im Wirkungsbereich eines Allquantors erster oder zweiter Stufe stehen, ψ oder χ boolesch ist.

Das eingeschränkte Fragment wird damit durch die nachfolgende Grammatik definiert.

$$\begin{aligned}\phi_{res} := \ & \phi_{step} \mid \phi_{res} \wedge \phi_{bool} \mid \phi_{bool} \wedge \phi_{res} \mid \phi_{res} \vee \phi_{res} \mid \\ & \forall x.\phi_{step} \mid \exists x.\phi_{res} \mid \exists X.\phi_{res} \mid\end{aligned}$$

Die Menge der *eingeschränkten Formeln* von wMSO(K, Σ) nennen wir wRMSO(K, Σ) (restricted wMSO, kurz wRMSO). Weiterhin sei wREMSO(K, Σ) (kurz wREMSO) die Menge aller *eingeschränkten existenziellen Formeln* von wMSO(K, Σ).

Sei $Z \subseteq$ wMSO ein Fragment von wMSO und *en* eine Positionsaufzählung für Gewichtsbäume. Eine formale Potenzreihe $S : T_\Sigma \to K$ heißt *en-definierbar in* Z, falls es einen Satz $\phi \in Z$ gibt, sodass $[\![\phi]\!]^{en} = S$. Die Menge aller Baumreihen $S \in K\langle\langle T_\Sigma \rangle\rangle$, die in einem Fragment Z *en*-definierbar sind, bezeichnen wir mit $K^{en-z}\langle\langle T_\Sigma \rangle\rangle$, wobei wir, in Anlehnung an andere Arbeiten, die Bezeichnung des Fragmentes Z klein schreiben und das vorangestellte w weglassen. So ist $K^{TS-mso}\langle\langle T_\Sigma \rangle\rangle$ beispielsweise die Menge der in wMSO TS-definierbaren Baumreihen. Ist *en* eine ordnungserhaltende Positionsaufzählung, basierend auf einer linearen Ordnung $<$ von K, so schreiben wir verkürzend $<$-definierbar (linear definierbar) in Z und $K^{<-z}$ anstelle von $en^<$-definierbar und $K^{en^< -z}$.

4.3 Erkennbarkeit in wMSO definierbarer Baumreihen

In diesem Abschnitt untersuchen wir, unter welchen Bedingungen Semantiken von wMSO-Formeln mit einer gegebenen Positionsaufzählung für Gewichtsbäume erkennbar sind. Für das Fragment wRMSO zeigen wir, dass die Semantik jeder dieser Formeln mit der entsprechenden Positionsaufzählung der Semantik schwach erkennbar ist. Für das Teillogik wEMSO und wMSO selbst müssen wir uns auf bi-lokal endliche starke Bimonoide und die Positionsaufzählungen TS und $en^{<}$ beschränken.

Nachfolgend sei, sofern nicht anders erwähnt, $K = (K, +, \cdot, \bar{\ }, 0, 1)$ stets ein starkes Bimonoid mit Negation und Σ immer ein Rangalphabet.

4.3.1 Charakterisierung erkennbarer Stufenfunktionen

Zunächst wollen wir gestufte Formeln untersuchen und den Zusammenhang zu erkennbaren Stufenfunktionen herstellen.

Proposition 4.3.1. *Sei* $\phi \in$ wMSO *eine Atomformel und en eine Positionsaufzählung für Gewichtsbäume. Dann ist* $[\![\phi]\!]^{en}$ *eine erkennbare Stufenfunktion.*

Beweis. Sei zunächst $\phi \equiv k$ für ein $k \in K$. Dann ist $[\![\phi]\!]^{en} = \tilde{k}$ eine erkennbare Stufenfunktion, denn $T^{v}_{\Sigma_{\mathrm{Free}(\phi)}} = T_{\Sigma_{\mathrm{Free}(\phi)}}$.

Alle verbleibenden atomaren Formeln ϕ sind boolesch. Nach Lemma 4.2.6 ist deshalb $[\![\phi]\!]^{en} = \mathbb{1}_{L(\phi)}$ ebenfalls eine erkennbare Stufenfunktion. □

Proposition 4.3.2. *Sei* $\phi \in$ wMSO *und en eine Positionsaufzählung für Gewichtsbäume. Ist* $[\![\phi]\!]^{en}$ *eine erkennbare Stufenfunktion, so auch* $[\![\neg\phi]\!]^{en}$.

Beweis. Nach der Semantikdefinition ist $[\![\neg\phi]\!]^{en} = \overline{[\![\phi]\!]^{en}}$ und es gibt eine Partition erkennbarer Baumsprachen $L_1, \ldots, L_n$ von $T_{\Sigma_{\mathrm{Free}\,\phi}}$ und paarweise verschiedene Elemente $k_1, \ldots, k_n \in K$ mit $[\![\phi]\!]^{en} = \sum_{i=1}^{n} k_i \cdot \mathbb{1}_{L_i}$. Dann ist aber $\overline{[\![\phi]\!]^{en}} = \sum_{i=1}^{n} \overline{k_i} \cdot \mathbb{1}_{L_i}$ und deshalb ebenfalls eine erkennbare Stufenfunktion. □

Proposition 4.3.3. *Sei* $\phi, \psi \in$ wMSO *und en eine Positionsaufzählung für Gewichtsbäume. Sind* $[\![\phi]\!]^{en}$ *und* $[\![\psi]\!]^{en}$ *erkennbare Stufenfunktionen, so sind auch* $[\![\phi \wedge \psi]\!]^{en}$ *und* $[\![\phi \vee \psi]\!]^{en}$ *erkennbare Stufenfunktionen.*

Beweis. Seien $[\![\phi]\!]^{en}$, $[\![\psi]\!]^{en}$ erkennbare Stufenfunktionen und $V := \mathrm{Free}(\phi) \cup \mathrm{Free}(\psi)$. Dann sind $[\![\phi]\!]^{en}_V$ und $[\![\psi]\!]^{en}_V$ nach Lemma 4.2.9 ebenfalls erkennbare Stufenfunktionen. Nach der Definition der Semantik ist

$$[\![\phi \vee \psi]\!]^{en} = [\![\phi]\!]^{en}_V + [\![\psi]\!]^{en}_V \quad \text{und} \quad [\![\phi \wedge \psi]\!]^{en} = [\![\phi]\!]^{en}_V \odot [\![\psi]\!]^{en}_V \,.$$

Proposition 3.3.5 impliziert somit die Behauptung. □

Lemma 4.3.4. *Sei* $\phi \in$ wMSO *eine gestufte Formel und en eine Positionsaufzählung für Gewichtsbäume. Dann ist die Semantik* $[\![\phi]\!]^{en}$ *eine erkennbare Stufenfunktion.*

Beweis. (Induktion über Formelaufbau) Sei ϕ konstant oder boolesch. Dann ist $[\![\phi]\!]^{en}$ nach Proposition 4.3.1 beziehungsweise Lemma 4.2.6 eine erkennbare Stufenfunktion. Seien nun $\phi, \psi \in$ wMSO gestufte Formeln, für welche die Behauptung gilt. Dann sind $[\![\neg\phi]\!]^{en}$, $[\![\phi \wedge \psi]\!]^{en}$ und $[\![\phi \vee \psi]\!]^{en}$ nach Proposition 4.3.2 und Proposition 4.3.3 erkennbare Stufenfunktionen. □

Da in einer gestuften Formel ϕ die Generalisierung erster Stufe nur in booleschen Teilformeln vorkommen kann, ist die Semantik von ϕ unabhängig von der gewählten Positionsaufzählung für Gewichtsbäume.

Korollar 4.3.5. *Sei $\phi \in$ wMSO eine gestufte Formel. Dann gilt für alle Positionsaufzählungen für Gewichtsbäume en_1 und en_2 die Gleichung $[\![\phi]\!]^{en_1} = [\![\phi]\!]^{en_2}$.*

Beweis. (Induktion über den Formelaufbau) Sei en eine Positionsaufzählung für Gewichtsbäume. Ist $\phi \equiv k$ für ein $k \in K$, dann ist $[\![\phi]\!]^{en} = \tilde{k}$. Ist ϕ boolesch, so ist $[\![\phi]\!]^{en} = \mathbb{1}_{L(\phi)}$.

Treffe nun die Behauptung auf zwei gestufte Formeln ϕ und ψ zu, wobei $S_\phi := [\![\phi]\!]^{en}_V$ und $S_\psi := [\![\psi]\!]^{en}_V$ mit $V := \mathrm{Free}(\phi) \cup \mathrm{Free}(\psi)$. Dann gelten

$$[\![\phi \vee \psi]\!]^{en}_V = S_\phi + S_\psi \quad , \quad [\![\phi \wedge \psi]\!]^{en}_V = S_\phi \odot S_\psi \quad \text{und}$$

$$[\![\neg\phi]\!]^{en} = \sum_{i=1}^{n} \overline{k_i} \cdot \mathbb{1}_{L_i} \quad \text{für} \quad S_\phi = \sum_{i=1}^{n} k_i \cdot \mathbb{1}_{L_i} \; .$$

Somit wird die Semantik $[\![\phi]\!]^{en}$ jeder gestuften Formel $\phi \in$ wMSO durch einen, von der Positionsaufzählung en unabhängigen, Term bestimmt. □

Analog zu den booleschen Formeln schreiben wir deshalb ab sofort verkürzend $[\![\phi]\!]_V$ für $[\![\phi]\!]^{en}_V$ beziehungsweise $[\![\phi]\!]$ für $[\![\phi]\!]^{en}$, falls ϕ gestuft ist.

Für $V = \emptyset$ identifizieren wir T_Σ mit T_{Σ_V} und somit auch $K\langle\langle T_\Sigma \rangle\rangle$ mit $K\langle\langle T_{\Sigma_V} \rangle\rangle$. Offenbar charakterisieren die gestuften Formeln unabhängig von der gewählten Positionsaufzählung die erkennbaren Stufenfunktionen.

Korollar 4.3.6. *Sei $S \in K\langle\langle T_\Sigma \rangle\rangle$. Es sind äquivalent:*

1. *S ist eine erkennbare Stufenfunktion.*
2. *Es gibt einen gestuften Satz $\phi \in$ wMSO mit $S = [\![\phi]\!]$.*

Beweis. Die Implikation (2. ⇒ 1.) folgt direkt aus Lemma 4.3.4.

(1. ⇒ 2.) Sei S eine erkennbare Stufenfunktion. Dann gibt es eine Partition erkennbarer Baumsprachen $L_1, \ldots, L_n$ von T_Σ und paarweise verschiedene $k_1, \ldots, k_n \in K$ mit $S = \sum_{i=1}^n k_i \cdot \mathbb{1}_{L_i}$. Nach Satz 4.2.7 gibt es boolesche Sätze $\phi_1, \ldots, \phi_n$ mit $[\![\phi_i]\!] = \mathbb{1}_{L_i}$ für $i \in [1, n]$. Wir bilden nun den gestuften Satz $\phi := \bigvee_{i=1}^n (k_i \wedge \phi_i)$. Sei en eine Positionsaufzählung für Gewichtsbäume. Nach Definition der Semantik ist dann

$$[\![\phi]\!]^{en} = \sum_{i=1}^{n} k_i \cdot \mathbb{1}_{L(\phi)} = \sum_{i=1}^{n} k_i \cdot \mathbb{1}_{L_i} = S \; .$$

□

4.3.2 Erkennbarkeit der Semantik von wMSO-Formeln

Wir betrachten nun das Fragment wRMSO sowie beliebige wMSO-Formeln und untersuchen die Erkennbarkeit der Semantiken in starken Bimonoiden.

Proposition 4.3.7. *Seien $\phi, \psi \in$ wMSO und en_1 und en_2 Positionsaufzählungen für Gewichtsbäume. Seien weiterhin $[\![\phi]\!]^{en_1}$ und $[\![\psi]\!]^{en_1}$ (schwach) en_2-erkennbar. Dann gelten:*

1. *$[\![\phi \vee \psi]\!]^{en_1}$ ist (schwach) en_2-erkennbar.*
2. *Ist ϕ oder ψ boolesch, dann ist $[\![\phi \wedge \psi]\!]^{en_1}$ (schwach) en_2-erkennbar.*

Beweis. Sei $V := \mathrm{Free}(\phi) \cup \mathrm{Free}(\psi)$, und $[\![\phi]\!]^{en_1}$, $[\![\psi]\!]^{en_1}$ en_2-erkennbar. Dann sind $[\![\phi]\!]_V^{en_1}$ und $[\![\psi]\!]_V^{en_1}$ nach Lemma 4.2.10 ebenfalls en_2-erkennbar.

(1.) Nach der Definition der Semantik ist $[\![\phi \vee \psi]\!]^{en_1} = [\![\phi]\!]_V^{en_1} + [\![\psi]\!]_V^{en_1}$. Aus Proposition 3.3.3 folgt damit, dass $[\![\phi \vee \psi]\!]^{en_1}$ en_2-erkennbar ist.

(2.) Ohne Beschränkung der Allgemeinheit sei ϕ boolesch. Dann ist $[\![\phi]\!]_V^{en_1} = \mathbb{1}_{L_V(\phi)}$ und $[\![\phi \wedge \psi]\!]^{en_1} = \mathbb{1}_{L_V(\phi)} \odot [\![\psi]\!]_V^{en_1}$. Also ist nach Proposition 3.3.4 $[\![\phi \wedge \psi]\!]^{en_1}$ ebenfalls en_2-erkennbar.

Seien $[\![\phi]\!]^{en_1}$, $[\![\psi]\!]^{en_1}$ schwach en_2-erkennbar. Dann ist nach Proposition 3.3.3 auch $[\![\phi \vee \psi]\!]^{en_1}$ schwach en_2-erkennbar. Ist ohne Beschränkung der Allgemeinheit ϕ boolesch, so ist mit Proposition 3.3.4 ebenfalls $[\![\phi \wedge \psi]\!]^{en_1}$ schwach en_2-erkennbar. □

Weil das Hadamard-Produkt erkennbarer Potenzreihen im Allgemeinen nicht erkennbar ist, ist für Formeln $\phi, \psi \in$ wMSO, deren Semantiken $[\![\phi]\!]^{en_1}$ und $[\![\psi]\!]^{en_1}$ en_2-erkennbar sind, $[\![\phi \wedge \psi]\!]^{en_1}$ im Allgemeinen nicht en_2-erkennbar.

Proposition 4.3.8. *Sei $\phi \in$ wMSO und seien en_1 und en_2 Positionsaufzählungen für Gewichtsbäume. Wenn $[\![\phi]\!]^{en_1}$ eine (schwach) en_2-erkennbare Potenzreihe ist, dann ist auch $[\![\exists x.\phi]\!]^{en_1}$ (schwach) en_2-erkennbar.*

Beweis. Sei $V := \mathrm{Free}(\exists x.\phi)$ und $\tau : \Sigma_{V \cup \{x\}} \to \Sigma_V$ eine Umbenennung derart, dass $\tau((\sigma, f)) = (\sigma, f|_V)$ für alle $(\sigma, f) \in \Sigma_{V \cup \{x\}}$ gelte. Weiterhin sei s ein Baum aus T_{Σ_V}. Es gilt für alle Positionen $w \in \mathrm{pos}(s)$, dass (1) s genau dann korrekt ist, wenn $s[x \to w]$ korrekt ist, denn $x \notin V$ und (2) kein $t \in \tau^{-1}(s)$ ist korrekt, wenn s nicht korrekt ist. Damit erhalten wir

$$\begin{aligned}
([\![\exists x.\phi]\!]^{en_1}, s) &= \sum_{w \in \mathrm{pos}(s)} ([\![\phi]\!]^{en_1}_{V \cup \{x\}}, s[x \to w]) \quad (1) \\
&= \sum_{t \in \tau^{-1}(s)} ([\![\phi]\!]^{en_1}_{V \cup \{x\}}, t) \qquad (2), ([\![\phi]\!]^{en_1}_{V \cup \{x\}}, t) = 0, \text{ falls } t \notin T^v_{\Sigma_V} \\
&= (\tau([\![\phi]\!]^{en_1}_{V \cup \{x\}}), s)\ .
\end{aligned}$$

Nach Lemma 4.2.10 ist $[\![\phi]\!]^{en_1}_{V \cup \{x\}}$ en_2-erkennbar, falls $[\![\phi]\!]^{en_1}$ en_2-erkennbar ist. Also ist auch $[\![\exists x.\phi]\!]^{en_1}$ nach Lemma 3.3.6 en_2-erkennbar.

Sei $[\![\phi]\!]^{en_1}$ schwach en_2-erkennbar. Dann ist $[\![\exists x.\phi]\!]^{en_1}$ mit obigem Beweis nach Lemma 3.3.6 ebenfalls schwach en_2-erkennbar. □

Vollkommen analog zum vorherigen Lemma erhält man das Ergebnis für den Existenzquantor zweiter Stufe.

Proposition 4.3.9. *Sei $\phi \in$ wMSO und seien en_1 und en_2 Positionsaufzählungen für Gewichtsbäume. Wenn $[\![\phi]\!]^{en_1}$ eine (schwach) en_2-erkennbare Potenzreihe ist, dann ist auch $[\![\exists X.\phi]\!]^{en_1}$ (schwach) en_2-erkennbar.*

Beweis. Sei $V := \mathrm{Free}(\exists X.\phi)$ und $\tau : \Sigma_{V\cup\{X\}} \to \Sigma_V$ eine Umbenennung, definiert durch $\tau((s,f)) := (s, f|_V)$ für alle $(\sigma, f) \in \Sigma_{V\cup\{x\}}$. Weiterhin sei s ein Baum aus T_{Σ_V}

$$\begin{aligned}([\![\exists X.\phi]\!]^{en_1}, s) &= \sum_{I\subseteq \mathrm{pos}(s)} ([\![\phi]\!]^{en_1}_{V\cup\{X\}}, s[X \to I]) \\ &= \sum_{t\in\tau^{-1}(s)} ([\![\phi]\!]^{en_1}_{V\cup\{X\}}, t)) \\ &= (\tau([\![\phi]\!]^{en_1}_{V\cup\{X\}}), s)\end{aligned}$$

Ist $[\![\phi]\!]^{en_1}$ en_2-erkennbar, ist somit nach Lemma 4.2.10 und Lemma 3.3.6 auch $[\![\exists X.\phi]\!]^{en_1}$ en_2-erkennbar.

Sei $[\![\phi]\!]^{en_1}$ schwach en_2-erkennbar. Dann ist $[\![\exists X.\phi]\!]^{en_1}$ mit obigem Beweis nach Lemma 3.3.6 ebenfalls schwach en_2-erkennbar. □

Im Allgemeinen sind weder $[\![\exists x.\phi]\!]^{en}$ noch $[\![\exists X.\phi]\!]^{en}$ erkennbaren Stufenfunktionen, auch wenn $[\![\phi]\!]^{en}$ eine erkennbare Stufenfunktion ist. Sei beispielsweise Σ ein Rangalphabet mit $max_\Sigma > 0$, K der Semiring der natürlichen Zahlen und en eine Positionsaufzählung. Dann ist $[\![1]\!]^{en} = \tilde{1}$, denn $T_{\Sigma_\emptyset} = T^v_{\Sigma_\emptyset}$. Doch für alle $s \in T_{\Sigma_\emptyset}$ ist $([\![\exists x.1]\!]^{en}, s) = |\mathrm{pos}(s)|$ und $([\![\exists X.1]\!]^{en}, s) = |\wp(\mathrm{pos}(s))|$.

Wir beobachten, dass die Semantik derjenigen eingeschränkter Formeln, in welchen alle Generalisierungen erster Stufe boolesche Formeln quantifizieren, sogar mit jeder beliebigen Positionsaufzählung für Gewichtsbäume en-erkennbar sind. Ist in einer Formel $\forall x.\phi$ die Teilformel ϕ hingegen gestuft, so ist $[\![\forall x.\phi]\!]^{en}$ im Allgemeinen nur erkennbar mit der Positionsaufzählung en.

Lemma 4.3.10. *Sei $\phi \in \mathrm{MSO}(K, \Sigma)$ und en eine Positionsaufzählung für Gewichtsbäume. Ist $[\![\phi]\!]^{en}$ eine erkennbare Stufenfunktion, so ist $[\![\forall x.\phi]\!]^{en}$ schwach en-erkennbar.*

Beweis. Wir folgen in diesem Beweis dem entsprechenden Beweis von Droste und Vogler [17, Lemma 5.5] für gewichtete MSO-Logik über Bäumen in Semiringen.

Seien $W := \mathrm{Free}(\phi)$ und $V := \mathrm{Free}(\forall x.\phi) = W \setminus \{x\}$. Angenommen $x \notin W$, dann betrachten wir zunächst die Formel $\phi' := \phi \vee edge_1(x, x)$. Aus der Semantikdefinition folgt $[\![\phi']\!]^{en}_{V\cup\{x\}} = [\![\phi]\!]^{en}_{V\cup\{x\}}$, also auch $[\![\forall x.\phi']\!]^{en}_{V\cup\{x\}} = [\![\forall x.\phi]\!]^{en}_{V\cup\{x\}}$. Außerdem ist ϕ' nach Proposition 4.3.3 eine erkennbare Stufenfunktion, weshalb wir nur den Fall $x \in W$ betrachten müssen.

Sei also $x \in W$. Da ϕ eine erkennbare Stufenfunktion ist, gibt es Baumsprachen $L_1, \dots, L_n$, die T_{Σ_W} partitionieren, und paarweise verschiedene Elemente $k_1, \dots, k_n \in K$ mit $[\![\phi]\!]^{en} = \sum_{i=1}^n k_i \cdot \mathbb{1}_{L_i}$. Wir bilden das Rangalphabet $\widetilde{\Sigma}_V := \Sigma_V \times [1, n]$, dessen Elemente wir als (σ, f, i) schreiben mit $\sigma \in \Sigma$, $f \in \{0,1\}^V$ und $i \in [1, n]$. Der Rang eines Symbols $(\sigma, f) \in \Sigma_V$ bleibe dabei erhalten; $rk((\sigma, f, i)) := rk_\Sigma(\sigma, f)$. Jeder Baum $s \in T_{\widetilde{\Sigma}_V}$ entspricht nun genau einem Paar (s', v). Dabei sei der Baum $s' \in T_{\Sigma_V}$ definiert durch Projektion auf die erste und zweite Komponente; $s' := \pi_{1,2} \circ s$. In s' steht also an jeder Stelle $w \in \mathrm{pos}(s)$ genau dann das Symbol (σ, f), wenn $s(w) = (\sigma, f, i)$. Die

Funktion $v : \mathrm{pos}(s) \to [1,n]$ ist bestimmt durch $v := \pi_3 \circ s$. Sie ordnet damit jeder Position $w \in \mathrm{pos}(s)$ mit $s(w) = (\sigma, f, i)$ die Zahl i zu. Umgekehrt entspricht jedes Paar (s', v) mit $s' \in T_{\Sigma_V}$ und $v : \mathrm{pos}(s') \to [1,n]$ einem Baum $s \in T_{\widetilde{\Sigma}_V}$ mit $s(w) = (s'(w), v(w))$. In s werden also alle Positionen w mit $v(w)$ markiert. Im Folgenden identifizieren wir deshalb Bäume $s \in T_{\widetilde{\Sigma}_V}$ mit dem zugehörigen Paar $s = (s', v)$.

Wir betrachten nun die folgende Baumsprache:

$$\widetilde{L} := \left\{ (s', v) \in T_{\widetilde{\Sigma}_V} \;\middle|\; \forall w \in \mathrm{pos}(s'), i \in [1,n] : v(w) = i \implies s'[x \to w] \in L_i \right\} .$$

Da $L_1, \dots, L_n$ eine Partition bilden, ist für jeden Baum $(s', v) \in \widetilde{L}$ die Funktion v eindeutig bestimmt. Also ist $\widetilde{L} = \bigcap_{i=1}^{n} \widetilde{L}_i$ mit

$$\widetilde{L}_i := \left\{ (s', v) \in T_{\widetilde{\Sigma}_V} \;\middle|\; \forall w \in \mathrm{pos}(s') : v(w) = i \implies s'[x \to w] \in L_i \right\} .$$

Offenbar enthält $\widetilde{L}_i$ ebenfalls alle die Bäume (s', v), in denen $v(w) \neq i$ an allen Positionen $w \in \mathrm{pos}(s')$ gilt. Wir zeigen nun, dass die Baumsprachen $\widetilde{L}_i$ erkennbar sind und damit auch die Baumsprache $\widetilde{L}$.

Die Baumsprachen $L_1, \dots, L_n$ sind erkennbar, also gibt es deterministische Baumautomaten $M_1, \dots, M_n$ mit $L(M_i) = L_i$ und $M_i = (Q_i, \Sigma, \delta^i, F_i)$ für $i \in [1,n]$. Zu jedem dieser Automaten M_i konstruieren wir einen Baumautomaten $\widetilde{M_i}$, welcher auf einem Baum $s = (s', v)$ das Verhalten von M_i simuliert. An jeder Stelle w mit $v(w) = i$ wird eine Kopie von M_i gestartet, für die x auf w gesetzt ist. Das realisieren wir analog zur Potenzmengenkonstruktion. Dabei müssen wir jedoch berücksichtigen, dass x gemäß der Definition von $s[x \to w]$ nur an einer Position gesetzt sein darf. Das werden wir mithilfe eines Flags gewährleisten.

Sei also $\widetilde{M_i} := (\widetilde{Q}_i, \widetilde{\Sigma}_V, \tilde{\delta}^i, \widetilde{F}_i)$ ein Baumautomat den Zuständen $\widetilde{Q}_i := \wp(Q_i \times \{0,1\})$ und den Finalzuständen $\widetilde{F}_i = \{U \subseteq Q_i \times \{0,1\} \mid U \cap (Q_i \times \{1\}) \subseteq F_i \times \{1\}\}$. Die finalen Zustandsmengen enthalten also Paare (q, b) mit $q \in Q_i$ und $b \in \{0,1\}$ derart, dass q final ist, falls $b = 1$ das Setzen von x an genau einer Position signalisiert. Insbesondere enthalten sie auch Zustände mit dem Flag-Wert 0, welches angibt, dass der betrachtete Baum für die Variable x nicht korrekt ist. Wir definieren die Familie der Transitionsfunktionen $\tilde{\delta}^i := \{\tilde{\delta}^i_{(\sigma,f,l)}\}_{(\sigma,f,l) \in \widetilde{\Sigma}_V}$ für alle $(\sigma, f, l) \in \widetilde{\Sigma}_V^{(m)}$ und $P_1, \dots, P_m \in \widetilde{Q}_i$ wie folgt:

$$\tilde{\delta}^i_{(\sigma,f,l)}(P_1, \dots, P_m) := \begin{cases} P' & \text{falls } l \neq i \text{ und} \\ P' \cup P'' & \text{falls } l = i \text{ mit} \end{cases}$$

$$P' := \left\{ \left(\delta^i_{(\sigma, f[x \to 0])}(p_1, \dots, p_m), \sum_{j=1}^{m} b_j \right) \;\middle|\; \forall j \in [1,m] : (p_j, b_j) \in P_j \;\wedge\; \sum_{j=1}^{m} b_j \leq 1 \right\},$$

$$P'' := \left\{ \left(\delta^i_{(\sigma, f[x \to 1])}(p_1, \dots, p_m), 1 \right) \;\middle|\; \forall j \in [1,m] : (p_j, 0) \in P_j \right\}.$$

P' beschreibt hierbei die Übernahme des Flag-Wertes: 0, falls alle Vorgängerzustände das Flag 0 tragen, das heißt x bisher nicht gesetzt wurde; 1, wenn genau ein Vorgängerzustand den Flag-Wert 1 hat, das heißt x genau einmal gesetzt wurde. P'' setzt das Flag auf 1, wenn $l = i$ und alle Vorgängerzustände den Flag-Wert 0 tragen.

Wir zeigen nun, dass $\widetilde{M}_i$ die Baumsprache $\widetilde{L}_i$ erkennt. M_i und $\widetilde{M}_i$ sind deterministisch, also sind für jeden Baum $(s', v) \in T_{\widetilde{\Sigma}_V}$ und jede Position $w \in \mathrm{pos}((s', v))$ die Mengen der Läufe $R_{M_i}(s'[x \to w])$ und $R_{\widetilde{M}_i}((s', v))$ einelementig. Wir bezeichnen diese Läufe mit $r_{s'[x \to w]}$ beziehungsweise $r_{(s',v)}$.

Behauptung: Es gilt die folgende Gleichung:

$$r_{(s',v)}(\varepsilon) = \left\{(r_{s'[x\to\emptyset]}(\varepsilon), 0)\right\} \cup \left\{(r_{s'[x\to w]}(\varepsilon), 1) \;\middle|\; w \in \mathrm{pos}(s') \;\wedge\; v(w) = i\right\} \tag{4.1}$$

Dabei stehe $s'[x \to \emptyset]$ für denjenigen Baum, der aus $s' \in T_{\Sigma_V}$ entsteht, indem jedes Symbol (σ, f) in s' durch $(\sigma, f \cup (x, 0))$ ersetzt wird. Dies ist wohldefiniert, weil Funktionen Mengen sind und V die Variable x nicht enthält.

Wir beobachten, dass der Baum $s'[x \to \emptyset]$ nicht korrekt ist, denn es gibt keine Position $w \in \mathrm{pos}(s'[x \to \emptyset])$ mit $\pi_2(s'[x \to \emptyset](w)) = 1$.

Beweis. Den Beweis der Behauptung führen wir induktiv über den Aufbau eines Baumes $s := (s', v)$. Dazu schreiben wir den Baum s als Funktionsterm $s := (\sigma, f, l)(s_1, \ldots, s_m)$. Für $j \in [1, m]$ ist ein Teilbaum $s_j : \mathbb{N}^* \dashrightarrow \widetilde{\Sigma}_V$ jeweils mit $\forall j.w \in \mathrm{pos}(s) : s_j(w) = s(w)$ definiert. Entsprechend erhalten wir für jeden Lauf $r \in R_{\widetilde{M}_i}(s)$ die Teilläufe $r_j \in R_{\widetilde{M}_i}(s_j)$ mit $\forall j.w \in \mathrm{pos}(s) : r_j(w) = r(j.w)$. Vollkommen analog dazu betrachten wir s' ebenfalls als Term $s' := (\sigma, f)(s'_1, \ldots, s'_m)$.

Sei $s := (s', v) \in T_{\widetilde{\Sigma}_V}$ ein Baum, welcher aus genau einem Blatt besteht, das heißt $\mathrm{pos}(s) = \{\varepsilon\}$. Sei $s(\varepsilon)$ mit (σ, f, l) beschriftet, wobei $\sigma \in \Sigma^{(0)}$, $f \in \{0,1\}^V$ und $l \in [1, n]$. Wir nehmen zuerst $l \neq i$ an und erhalten:

$$\begin{aligned}
r_{(s',v)}(\varepsilon) &= \tilde{\delta}^i_{(\sigma,f,l)}() \\
&\qquad \text{mit Definition von } \tilde{\delta}^i \text{ folgt} \\
&= P' \\
&= \left\{(\delta^i_{(\sigma, f[x\to 0])}, 0)\right\} \\
&\qquad \text{da } v(w) = l \neq i \text{ folgt} \\
&= \left\{(r_{s'[x\to\emptyset]}(\varepsilon), 0)\right\} \cup \left\{(r_{s'[x\to w]}(\varepsilon), 1) \;\middle|\; w \in \mathrm{pos}(s') \;\wedge\; v(w) = i\right\} .
\end{aligned}$$

Angenommen $l = i$, dann gilt

$$\begin{aligned}
r_{(s',v)}(\varepsilon) &= \tilde{\delta}^i_{(\sigma,f,l)}() \\
&\qquad \text{mit Definition von } \tilde{\delta}^i \text{ folgt} \\
&= P' \cup P'' \\
&= \left\{(\delta^i_{(\sigma, f[x\to 0])}, 0)\right\} \cup \left\{(\delta^i_{(\sigma, f[x\to 1])}, 1)\right\} \\
&\qquad \text{aus } v(w) = l = i \text{ folgt} \\
&= \left\{(r_{s'[x\to\emptyset]}(\varepsilon), 0)\right\} \cup \left\{(r_{s'[x\to w]}(\varepsilon), 1) \;\middle|\; w \in \mathrm{pos}(s') \;\wedge\; v(w) = i\right\} .
\end{aligned}$$

Nun gelte die Aussage für die Bäume $s_1, \ldots, s_m \in T_{\tilde{\Sigma}_V}$. Wir werden die nach folgende Gleichung benötigen.

(4.2)
$$\left\{ \left(\delta^i_{(\sigma, f[x \to 0])}(p_1, \ldots, p_m), \sum_{j=1}^{m} b_j \right) \middle| \forall j \in [1,m] : (p_j, b_j) \in r_{s_j}(\varepsilon) \ \wedge \ \sum_{j=1}^{m} b_j \leq 1 \right\}$$
$$= \left\{ (r_{s'[x \to \emptyset]}(\varepsilon), 0) \right\} \cup \left\{ (r_{s'[x \to uw]}(\varepsilon), 1) \mid u \in [1,m] \ \wedge \ w \in \mathrm{pos}(s'_u) \ \wedge \ v_u(w) = i \right\}$$

Sie lässt sich folgendermaßen herleiten.

$$\left\{ \left(\delta^i_{(\sigma, f[x \to 0])}(p_1, \ldots, p_m), \sum_{j=1}^{m} b_j \right) \middle| \forall j \in [1,m] : (p_j, b_j) \in r_{s_j}(\varepsilon) \ \wedge \ \sum_{j=1}^{m} b_j \leq 1 \right\}$$

mit der Induktionsvoraussetzung folgt

$$= \left\{ \left(\delta^i_{(\sigma, f[x \to 0])}(p_1, \ldots, p_m), \sum_{j=1}^{m} b_j \right) \middle| \forall j \in [1,m] : \Big[(p_j, b_j) = (r_{s'_j[x \to \emptyset]}(\varepsilon), 0) \ \vee \right.$$
$$\left. \Big((p_j, b_j) = (r_{s'_j[x \to w]}(\varepsilon), 1) \ \wedge \ w \in \mathrm{pos}(s'_j) \ \wedge \ v'_j(w) = i \Big) \Big] \ \wedge \ \sum_{j=1}^{m} b_j \leq 1 \right\}$$
$$= \left\{ \left(\delta^i_{(\sigma, f[x \to 0])}(r_{s'_1[x \to \emptyset]}(\varepsilon), \ldots, r_{s'_m[x \to \emptyset]}(\varepsilon)), 0 \right) \right\} \cup$$
$$\Big\{ (\delta^i_{(\sigma, f[x \to 0])}(r_{s'_1[x \to \emptyset]}(\varepsilon), \ldots, r_{s'_u[x \to w]}(\varepsilon), \ldots, r_{s'_m[x \to \emptyset]}(\varepsilon)), 1) \ \Big|$$
$$u \in [1,m] \ \wedge \ w \in \mathrm{pos}(s'_u) \ \wedge \ v_u(w) = i \Big\}$$
$$= \left\{ (r_{s'[x \to \emptyset]}(\varepsilon), 0) \right\} \cup \left\{ (r_{s'[x \to uw]}(\varepsilon), 1) \mid u \in [1,m] \ \wedge \ w \in \mathrm{pos}(s'_u) \ \wedge \ v_u(w) = i \right\}$$

Diese Gleichung kommt nun in einer Fallunterscheidung zum Einsatz. Sei $s(\varepsilon) = (\sigma, f, l)$ mit $\sigma \in \Sigma^{(0)}$, $f \in \{0,1\}^V$ und $l \in [1,n]$. Wir nehmen zunächst $l \neq i$ an.

$$r_{(s',v)}(\varepsilon)$$
$$= \tilde{\delta}^i_{(\sigma,f,l)}(r_{s_1}(\varepsilon), \ldots, r_{s_m}(\varepsilon))$$

mit Definition von $\tilde{\delta}^i$ folgt

$$= \left\{ \left(\delta^i_{(\sigma, f[x \to 0])}(p_1, \ldots, p_m), \sum_{j=1}^{m} b_j \right) \middle| \forall i \in [1,m] : (p_j, b_j) \in r_{s_j}(\varepsilon) \ \wedge \ \sum_{j=1}^{m} b_j \leq 1 \right\}$$

Anwendung von Gleichung 4.2 ergibt

$$= \left\{ (r_{s'[x \to \emptyset]}(\varepsilon), 0) \right\} \cup \left\{ (r_{s'[x \to uw]}(\varepsilon), 1) \mid u \in [1,m] \ \wedge \ w \in \mathrm{pos}(s'_u) \ \wedge \ v_u(w) = i \right\}$$
$$= \left\{ (r_{s'[x \to \emptyset]}(\varepsilon), 0) \right\} \cup \left\{ (r_{s'[x \to w]}(\varepsilon), 1) \mid w \in \mathrm{pos}(s') \ \wedge \ v(w) = i \right\}$$

Sei anderenfalls $l = i$.

$$
\begin{aligned}
&r_{(s',v)}(\varepsilon)\\
&= \tilde{\delta}^i_{(\sigma,f,l)}(r_{s_1}(\varepsilon),\dots,r_{s_m}(\varepsilon))\\
&\qquad\text{mit Definition von } \tilde{\delta}^i \text{ folgt}\\
&= \left\{ \left(\delta^i_{(\sigma,f[x\to 0])}(p_1,\dots,p_m), \sum_{j=1}^m b_j \right) \,\middle|\, \forall i \in [1,m] : (p_j,b_j) \in r_{s_j}(\varepsilon) \wedge \sum_{j=1}^m b_j \le 1 \right\}\\
&\quad \cup \left\{ (\delta^i_{(\sigma,f[x\to 1])}(p_1,\dots,p_m),1) \,\middle|\, \forall i \in [j,m] : (p_j,0) \in r_{s_j}(\varepsilon) \right\}\\
&\qquad\text{Anwendung von Gleichung 4.2 und der Induktionsvoraussetzung ergibt}\\
&= \big\{(r_{s'[x\to\emptyset]}(\varepsilon),0)\big\} \cup \big\{(r_{s'[x\to uw]}(\varepsilon),1) \,\big|\, u \in [1,m] \ \wedge\ w \in \mathrm{pos}(s'_u) \ \wedge\ v_u(w) = i\big\}\\
&\quad \cup \left\{ (\delta^i_{(\sigma,f[x\to 1])}(r_{s'_1[x\to\emptyset]}(\varepsilon),\dots,r_{s'_m[x\to\emptyset]}(\varepsilon)),1) \right\}\\
&= \big\{(r_{s'[x\to\emptyset]}(\varepsilon),0)\big\} \cup \big\{(r_{s'[x\to uw]}(\varepsilon),1) \,\big|\, u \in [1,m] \ \wedge\ w \in \mathrm{pos}(s'_u) \ \wedge\ v_u(w) = i\big\}\\
&\quad \cup \big\{(r_{s'[x\to\varepsilon]}(\varepsilon),1)\big\}\\
&= \big\{(r_{s'[x\to\emptyset]}(\varepsilon),0)\big\} \cup \big\{(r_{s'[x\to w]}(\varepsilon),1) \,\big|\, w \in \mathrm{pos}(s') \ \wedge\ v(w) = i\big\}
\end{aligned}
$$

Damit haben wir Gleichung 4.1 bewiesen. □

Nach Voraussetzung ist $L(M_i) = L_i$, sodass für jede Position $w \in \mathrm{pos}(s')$ gilt:

$$s'[x \to w] \in L_i \iff r_{s'[x\to w]}(\varepsilon) \in F\,.$$

Daraus folgt wiederum, dass

$$
\begin{aligned}
(s',v) \in \widetilde{L}_i &\iff \forall w \in \mathrm{pos}(s') : v(w) = i \implies r_{s'[x\to w]}(\varepsilon) \in F\\
&\iff r_{(s',v)}(\varepsilon) \in \widetilde{F} \qquad\qquad \text{(Gleichung 4.1)}\\
&\iff (s',v) \in L(\widetilde{M}_i)\,.
\end{aligned}
$$

Also wird $\widetilde{L}_i$ von $\widetilde{M}_i$ erkannt und damit existiert auch ein deterministischer Baumautomat $\widetilde{M} := (\widetilde{Q}, \widetilde{\Sigma}_V, \tilde{\delta}, \widetilde{F})$ mit $L(\widetilde{M}) = \widetilde{L} = \bigcap_{i=1}^n \widetilde{L}_i$. Aus diesem konstruieren wir einen schwachen gewichteten Baumautomaten $M := (\widetilde{Q}, \widetilde{\Sigma}_V, \mu, \gamma)$. Dazu definieren wir für alle $m \ge 0$, $(\sigma, f, l) \in \widetilde{\Sigma}_V$ und $q_1, \dots, q_m, q \in \widetilde{Q}$

$$
\begin{aligned}
\mu_m((\sigma,f,l))_{q_1,\dots,q_m,q} &:= \begin{cases} k_l & \text{falls } \tilde{\delta}^i_{(\sigma,f,l)}(q_1,\dots,q_m) = q\,,\\ 0 & \text{sonst,} \end{cases}\\
\gamma(q) &:= \begin{cases} 1 & \text{falls } q \in \widetilde{F} \text{ und}\\ 0 & \text{sonst}\,. \end{cases}
\end{aligned}
$$

Sei $(s', v) \in T_{\widetilde{\Sigma}_V}$. Da $\widetilde{M}$ deterministisch ist, gibt es genau einen Lauf $r \in R_{\widetilde{M}}((s', v))$. Der Lauf r ist ebenfalls ein Lauf von M auf (s', v). Also hängt das Gewicht $(S_M, (s', v))$ nun allein davon ab, ob r in $\widetilde{M}$ erfolgreich ist oder nicht. Es gilt somit

$$
(S_M,(s',v)) = \begin{cases} \prod_{i=1}^{|\mathrm{pos}(s')|} M((s',v),r)(en(M((s',v),r))(i)) & \text{falls } (s',v) \in \widetilde{L}\,,\\ 0 & \text{sonst.} \end{cases}
$$

Wir beobachten, dass an jeder Position $w \in \mathrm{pos}((s', v))$ das Gewicht $M((s', v), r)(w)$ der Wert $k_{v(w)} \in \{k_1, \ldots, k_n\}$ ist. Dieses ist aber genau $([\![\phi]\!]^{en}, s'[x \to w])$. Also gilt für die Gewichtsbäume von M und ϕ:

$$M((s', v), r)(w) = \phi_x^{en}(s')(w) .$$

Abschließend definieren wir eine Umbenennung $\tau : \widetilde{\Sigma}_V \to \Sigma_V$ mit $\tau((\sigma, f, l)) := (\sigma, f)$ für alle $(\sigma, f, l) \in \widetilde{\Sigma}_V$. Da $L_1, \ldots, L_n$ eine Partition bilden, hat jeder Baum $s' \in T_{\Sigma_V}$ genau ein Urbild $(s', v) \in \tau^{-1}(s')$ mit $(s', v) \in \widetilde{L}$. Das Gewicht $(S_M, (s', \theta))$ aller anderen Urbilder $(s', \theta) \in \tau^{-1}(s')$ beträgt 0. Damit erhalten wir für alle $s' \in T_{\Sigma_V}$:

$$\begin{aligned}
(\tau(S_M), s') &= \sum_{(s',\theta) \in \tau^{-1}(s')} (S_M, (s', \theta)) \\
&\qquad (s', v) \text{ sei das eindeutige Urbild mit } (s', v) \in \widetilde{L} \\
&= (S_M, (s', v)) \\
&\qquad \text{sei } r \in R_{\widetilde{M}}((s', v)) \text{ (Das heißt, } r \text{ ist eindeutig.)} \\
&= \prod_{i=1}^{|\mathrm{pos}(s')|} M((s', v), r)(en(M((s', v), r))(i)) \\
&\qquad \text{aus } M((s', v), r)(w) = \phi_x^{en}(s')(w) \text{ folgt} \\
&= \prod_{i=1}^{|\mathrm{pos}(s')|} \phi_x^{en}(s')(en(\phi_x^{en}(s'))(i)) \\
&= \prod_{i=1}^{|\mathrm{pos}(s')|} ([\![\phi]\!]^{en}_{\mathrm{Free}(\phi) \cup \{x\}}, s'[x \to en(\phi_x^{en}(s'))(i)]) \\
&= ([\![\forall x.\phi]\!]^{en}, s') \qquad \text{Semantikdefinition von } \forall .
\end{aligned}$$

Also ist $\tau(S_M) = [\![\forall x.\phi]\!]^{en}$ und damit nach Lemma 3.3.6 schwach en-erkennbar. □

Wir haben in diesem Beweis einen Baumautomaten, der die Baumsprache $\tilde{L}$ erkennt, direkt konstruiert. Mit einer Idee von Droste und Gastin in [17, Lemma 5.4] können wir die Erkennbarkeit von $\tilde{L}$ auch mithilfe des Satzes von Büchi für Baumsprachen zeigen. Es folgt eine kurze Skizze dieses Beweises.

Beweis. (Skizze)
Zunächst definieren wir $\widetilde{\Sigma} := \Sigma \times [1, n]$ und damit $\widetilde{\Sigma}_V := \widetilde{\Sigma} \times \{0, 1\}^V$. Der obige Beweis lässt sich vollkommen analog mit dieser Definition führen. Ein Baum $s \in T_{\widetilde{\Sigma}_V}$ entspricht nun einem Paar (s', v) mit $s' := \pi_{1,3} \circ s$ und $v := \pi_2 \circ s$. Wir konstruieren die Formel $\widetilde{\phi} \in \mathrm{MSO}(\widetilde{\Sigma})$ aus einer Formel $\phi \in \mathrm{MSO}(\Sigma)$, indem wir alle Teilformeln der Form $label_\sigma(x)$ durch $\bigvee_{i \in [1,n]} label_{(\sigma,i)}(x)$ ersetzen. Nun folgern wir $(s', v) \models \widetilde{\psi} \iff s' \models \psi$ für alle $(s', v) \in T_{\widetilde{\Sigma}_V}$ und Formeln $\phi \in \mathrm{MSO}(\Sigma)$, denn für alle $x \in V$ gilt folgende Äquivalenz:

$$(s', v) \models \bigvee_{i \in [1,n]} label_{(\sigma,i)}(x) \iff s' \models label_\sigma(x) .$$

Nach dem Satz von Büchi für Baumsprachen gibt es für jede Baumsprache L_j mit $i \in [i,n]$ eine Formel ψ_i, die L_j definiert. Wir betrachten die Formel

$$\zeta := \forall x \left(\bigwedge_{i \in [1,n]} \bigvee_{\sigma \in \Sigma} label_{(\sigma,i)}(x) \to \widetilde{\psi_i} \right) .$$

Mit dieser erhalten wir für alle $(s', v) \in T^v_{\widetilde{\Sigma}_V}$:

$$\begin{aligned}
(s',v) \models \zeta \quad &\iff \quad \forall w \in \mathrm{pos}(s'), i \in [1,n] : v(w) = i \implies (s'[x \to w], v) \models \widetilde{\psi_i} \\
&\iff \quad \forall w \in \mathrm{pos}(s'), i \in [1,n] : v(w) = i \implies (s'[x \to w]) \models \psi_i \\
&\iff \quad \forall w \in \mathrm{pos}(s'), i \in [1,n] : v(w) = i \implies (s'[x \to w]) \in L_i \ .
\end{aligned}$$

Also definiert ζ die Baumsprache $\widetilde{L}$. □

Auch wenn $[\![\phi]\!]^{en}$ eine erkennbare Stufenfunktion ist, so ist $[\![\forall x.\phi]\!]^{en}$ im Allgemeinen keine erkennbare Stufenfunktion. Betrachten wir dazu wieder den Semiring der natürlichen Zahlen und ein Rangalphabet Σ mit $max_\Sigma > 0$. Dann ist $[\![2]\!]^{en} = \tilde{2}$, denn $T_{\Sigma_\emptyset} = T^v_{\Sigma_\emptyset}$. Für alle $s \in T_{\Sigma_\emptyset}$ gilt jedoch $([\![\forall x.2]\!]^{en}, s) = 2^{|\mathrm{pos}(s)|}$.

Proposition 4.3.11. *Sei $\phi \in$ wMSO und en eine Positionsaufzählung für Gewichtsbäume. Sei K zudem ein multiplikativ lokal endliches und kommutatives Bimonoid. Ist $[\![\phi]\!]^{en}$ eine erkennbare Stufenfunktion, so ist $[\![\forall X.\phi]\!]^{en}$ ebenfalls eine erkennbare Stufenfunktion.*

Beweis. Sei $V := \mathrm{Free}(\forall X.\phi)$ und $\tau : \Sigma_{V \cup \{X\}} \to \Sigma_V$ eine Umbenennung definiert durch $\tau((\sigma, f)) := (\sigma, f|_V)$ für alle $(\sigma, f) \in \Sigma_{V \cup \{X\}}$. Da K kommutativ ist, gilt nach Definition der Semantik für alle $s \in T^v_{\Sigma_V}$ unabhängig von der auf $\wp(V)$ fixierten Ordnung:

$$\begin{aligned}
([\![\forall X.\phi]\!]^{en}, s) &= \prod_{I \subseteq \mathrm{pos}(s)} ([\![\phi]\!]_{V \cup \{X\}}, s[X \to I]) \\
&= \prod_{s' \in \tau^{-1}(s)} ([\![\phi]\!]_{V \cup \{X\}}, s') \\
&= (\Pi_\tau([\![\phi]\!]_{V \cup \{X\}}), s) \ .
\end{aligned}$$

Nach Proposition 3.3.7 ist $[\![\forall X.\phi]\!]^{en}$ also eine erkennbare Stufenfunktion. □

Nun fassen wir die Erkenntnisse dieses Abschnittes zur Erkennbarkeit zusammen. Zunächst betrachten wir starke Bimonoide ohne weitere Einschränkungen und Formeln aus wRMSO.

Satz 4.3.12. *Sei K ein starkes Bimonoid mit Negation und en eine Positionsaufzählung für Gewichtsbäume. Dann gilt:*

$$K^{en-rmso}\langle\!\langle T_\Sigma \rangle\!\rangle \subseteq K^{en-wrec}\langle\!\langle T_\Sigma \rangle\!\rangle \ .$$

Beweis. Wir beweisen die Aussage induktiv über den Formelaufbau, wobei wir die gestuften Formeln als kleinste Teilformeln betrachten. Sei also $\phi \in$ wRMSO.

- Gestufte Formeln ϕ:
  Nach Lemma 4.3.4 ist die Semantik $[\![\phi]\!]^{en}$ eine erkennbare Stufenfunktion und somit nach Lemma 3.2.10 schwach en-erkennbar.

- Generalisierung zweiter Stufe $\forall X.\phi$
  Nach Definition der eingeschränkten Formeln muss ϕ boolesch sein. Also ist auch $\forall X.\phi$ boolesch und damit gestuft.

Gelte die Aussage nun für die Formeln $[\![\phi]\!]^{en}$ und $[\![\psi]\!]^{en}$.

- Konjunktion $\phi \wedge \psi$:
  Sei ohne Beschränkung der Allgemeinheit ϕ boolesch. Dann ist nach Proposition 4.3.7 $[\![\phi \wedge \psi]\!]^{en}$ schwach en-erkennbar.

- Disjunktion $\phi \vee \psi$:
  Ebenfalls nach Proposition 4.3.7 ist $[\![\phi \vee \psi]\!]^{en}$ schwach en-erkennbar.

- Partikularisierung erster $\exists x.\phi$ und zweiter Stufe $\exists X.\phi$:
  Nach Proposition 4.3.8 und Proposition 4.3.9 sind $[\![\exists x.\phi]\!]^{en}$ und $[\![\exists X.\phi]\!]^{en}$ schwach en-erkennbar.

□

Den Abschluss dieses Abschnittes bildet der folgende Satz, welcher für die Positionsaufzählungen für Gewichtsbäume TS und $en^{<}$ die Erkennbarkeit von wEMSO-Formeln zusammenfasst. Hierbei beschränken wir uns, wie in [18], auf bi-lokal endliche starke Bimonoide. Das ist notwendig, da die Erkennbarkeit des Hadamard-Produktes, welches wir für die Konjunktion benötigen, im Allgemeinen sonst nicht gegeben wäre.

Satz 4.3.13. *Sei K ein bi-lokal endliches starkes Bimonoid mit Negation und $<$ eine lineare Ordnung auf K. Dann gelten:*

- $K^{TS-emso}\langle\langle T_\Sigma\rangle\rangle \subseteq K^{TS-wrec}\langle\langle T_\Sigma\rangle\rangle$ *und*

- $K^{<-emso}\langle\langle T_\Sigma\rangle\rangle \subseteq K^{<-wrec}\langle\langle T_\Sigma\rangle\rangle$.

Ist K zudem kommutativ, gelten darüber hinaus:

- $K^{TS-mso}\langle\langle T_\Sigma\rangle\rangle \subseteq K^{TS-wrec}\langle\langle T_\Sigma\rangle\rangle$ *und*

- $K^{<-mso}\langle\langle T_\Sigma\rangle\rangle \subseteq K^{<-wrec}\langle\langle T_\Sigma\rangle\rangle$.

Beweis. Nach Lemma 3.2.10 sind alle erkennbaren Stufenfunktionen mit jeder Positionsaufzählung en schwach erkennbar, insbesondere also für $en = TS$ und $en = en^{<}$. Sei en im Folgenden entweder stets die Positionsaufzählung TS oder stets die Positionsaufzählung $en^{<}$. Diese Einschränkung benötigen wir zur Anwendung von Satz 3.2.17. Induktiv über den Formelaufbau folgt nun die Aussage.

- Atomformeln ϕ:
  Nach Proposition 4.3.1 ist $[\![\phi]\!]^{en}$ eine erkennbare Stufenfunktion.

Für $\phi, \psi \in$ wMSO seien nun $[\![\phi]\!]^{en}$ und $[\![\psi]\!]^{en}$ schwach en-erkennbar. Aus Satz 3.2.17 folgt, dass dann $[\![\phi]\!]^{en}$ und $[\![\psi]\!]^{en}$ erkennbare Stufenfunktionen sind, denn K ist bi-lokal endlich und en entweder die Positionsaufzählung TS oder $en^{<}$.

- Negation $\neg\phi$:
  Nach Proposition 4.3.2 ist $[\![\neg\phi]\!]^{en}$ eine erkennbare Stufenfunktion.

- Konjunktion $\phi \wedge \psi$ und Disjunktion $\phi \vee \psi$:
  Nach Proposition 4.3.3 sind $[\![\phi \wedge \psi]\!]^{en}$ und $[\![\phi \vee \psi]\!]^{en}$ erkennbare Stufenfunktionen.

- Partikularisierung erster $\exists x.\phi$ und zweiter Stufe $\exists X.\phi$:
  Nach Proposition 4.3.8 und Proposition 4.3.9 sind $[\![\exists x.\phi]\!]^{en}$ und $[\![\exists X.\phi]\!]^{en}$ schwach en-erkennbar.

- Generalisierung erster Stufe $\forall x.\phi$:
  Nach Lemma 4.3.10 ist $[\![\forall x.\phi]\!]^{en}$ schwach en-erkennbar.

Sei K kommutativ und $[\![\phi]\!]^{en}$ schwach en-erkennbar und damit nach Satz 3.2.17 eine erkennbare Stufenfunktion.

- Generalisierung zweiter Stufe $\forall X.\phi$:
  Nach Proposition 4.3.11 ist $[\![\forall X.\phi]\!]^{en}$ eine erkennbare Stufenfunktion.

$\square$

4.4 Definierbarkeit erkennbarer Baumreihen in wMSO

Der finale Abschnitt führt nun zu den Hauptergebnissen dieser Arbeit. Wir konstruieren dazu eine Formel, welche das Verhalten eines gewichteten Baumautomaten reflektiert. Einerseits erhalten wir die Charakterisierung der mit einer Positionsaufzählung für Gewichtsbäume schwach erkennbaren Potenzreihen durch das Fragment wRMSO. Andererseits zeigen wir, dass sich in bi-lokal endlichen Bimonoiden die, mit den Positionsaufzählungen TS und $<$-erkennbaren, Baumreihen durch wEMSO charakterisieren lassen.

Zur Beschreibung formaler Potenzreihen werden wir die Kovalenz (exklusives oder) und die Implikation benutzen. Für Formeln $\phi, \psi \in$ wMSO führen wir sie deshalb als Abkürzungen ein:

$$\phi \dot\vee \psi := \neg(\neg(\phi \wedge \neg\psi) \wedge \neg(\neg\phi \wedge \psi)) \text{ und}$$
$$\phi \to \psi := \neg(\phi \wedge \neg\psi)\ .$$

Sind ϕ und ψ boolesch, dann sind offenbar auch $\phi \dot\vee \psi$ und $\phi \to \psi$ boolesch. Nach Lemma 4.2.6 gelten in diesem Fall $[\![\phi \dot\vee \psi]\!] = \mathbb{1}_{L(\phi \dot\vee \psi)}$ und $[\![\phi \to \psi]\!] = \mathbb{1}_{L(\phi \to \psi)}$ für beliebige Positionsaufzählungen für Gewichtsbäume. Sind ϕ und ψ hingegen gestufte Formeln, so ist die Semantik unserer Abkürzungen nach Lemma 4.3.4 stets eine erkennbare Stufenfunktion.

Außerdem wollen wir noch eine an die ungewichtete MSO-Logik angelehnte Partikularisierung als Abkürzung einführen:

$$\mathsf{E}\, x.\phi := \neg\forall x.\neg\phi\ .$$

Ist ϕ boolesch, dann auch $\mathsf{E}\, x.\phi$.

Im Folgenden wollen wir das Verhalten eines gewichteten Baumautomaten mithilfe einer wRMSO-Formel beschreiben. Hierfür werden wir zunächst eine boolesche Formel konstruieren, die einen Lauf eines gewichteten Baumautomatens beschreibt. Sei dazu $M = (Q, \Sigma, \mu, \gamma)$ ein gewichteter Baumautomat.

- Die *Menge aller Transitionen* in M sei

$$B_M := \left\{ (\vec{q}, \sigma, q) \;\middle|\; m \geq 0 \wedge \vec{q} \in Q^m \wedge \sigma \in \Sigma^{(m)} \wedge q \in Q \right\}\ .$$

- Für die Mengenvariablen $X_{\vec{q},\sigma,q}$ mit $(\vec{q}, \sigma, q) \in B_M$ ist die boolesche *Partitionsformel* $part \in$ wMSO wie folgt definiert:

$$part := \forall x. \left(\dot{\bigvee_{(\vec{q},\sigma,q) \in B_M}} x \in X_{\vec{q},\sigma,q} \right)\ .$$

- Die *Laufformel von M und v* ist die folgende Formel $\psi_M \in$ wMSO über den freien Variablen $X_{\vec{q},\sigma,q}$ mit $(\vec{q}, \sigma, q) \in B_M$:

$$\psi_M := part\ \wedge\ \psi_M^1\ \wedge\ \psi_M^2\ ,$$

wobei

$$\psi_M^1 := \bigwedge_{(\vec{q},\sigma,q)\in B_M} \forall x.\, \big((x \in X_{\vec{q},\sigma,q}) \to label_\sigma(x)\big) \text{ und}$$

$$\psi_M^2 := \bigwedge_{\substack{(\vec{q},\sigma,q)\in B_M\\ \vec{q}=(q_1,\dots,q_m)\\ |\vec{q}|>0}} \forall x.\, \Big((x \in X_{\vec{q},\sigma,q}) \to \mathsf{E}\, y_1 \dots \mathsf{E}\, y_{|\vec{q}|}.\psi_M^{\vec{q}}\Big) \quad \text{mit}$$

$$\psi_M^{\vec{q}} := \bigwedge_{1\le i\le|\vec{q}|} \left(edge_i(x,y_i) \;\wedge \dot{\bigvee_{(\vec{p},\sigma,q_i)\in B_M}} y_i \in X_{\vec{p},\sigma,q_i} \right) .$$

Unter Berücksichtigung der Abkürzungen $\dot{\vee}$ und $\to$ ist ψ_M boolesch, und deshalb ist $[\![\psi_M]\!] = \mathbb{1}_{L(\psi_M)}$. Vereinfacht ausgedrückt, fixiert die Lauf-Formel für jede Belegung der Variablen $X_{\vec{q},\sigma,q}$ mit $(\vec{q},\sigma,q) \in B_M$ einen Lauf von M auf einem Baum. Die Formel *part* fasst die Positionen nach ihren Transitionen zusammen, ψ_M^1 besagt, dass jede Position mit dem Symbol der jeweiligen Transition beschriftet ist und ψ_M^2 stellt schließlich sicher, dass alle Positionen und deren Kinder mit Zuständen der zugehörigen Transition belegt sind.

Satz 4.4.1. *Sei K ein starkes Bimonoid mit Negation und en eine Positionsaufzählung für Gewichtsbäume. Dann gilt:*

$$K^{en-wrec}\langle\!\langle T_\Sigma \rangle\!\rangle \subseteq K^{en-remso}\langle\!\langle T_\Sigma \rangle\!\rangle \;.$$

Beweis. Sei S eine Baumreihe und $M = (Q,\Sigma,\mu,\gamma)$ ein wwta, der S schwach *en*-erkennt. Wir setzen $V = \big\{X_{\vec{q},\sigma,q} \;\big|\; (\vec{q},\sigma,q) \in B_M\big\}$, wobei B_M die Menge der Transitionen von M ist. Wir zeigen zunächst: Es gibt für jeden Baum $t \in T_\Sigma$ eine Bijektion der Läufe $R_M(t)$ auf die Menge

$$P_{\psi_M}(t) := \{\rho \mid \rho \text{ ist } (V,t)\text{-Variablenbelegung } \wedge\ (t,\rho) \models \psi_M\} \;.$$

Sei dazu $r : \mathrm{pos}(t) \to Q$ ein Lauf von M auf t. Wir definieren die (V,t)-Belegung $\rho_r : V \to \wp(\mathrm{pos}(t))$ folgendermaßen: Für alle $(q_1,\dots,q_m,\sigma,q) = (\vec{q},\sigma,q) \in B_M$ sei

$$\rho_r(X_{\vec{q},\sigma,q}) := \{w \in \mathrm{pos}(t) \mid r(w.1) = q_1 \,,\, \dots \,,\, r(w.m) = q_m \;\wedge\; t(w) = \sigma \;\wedge\; r(w) = q\} \;.$$

Das Paar (t,ρ_r) erfüllt somit die Laufformel ψ_M.

Sei umgekehrt ρ eine (V,t)-Belegung mit $(t,\rho) \models \psi_M$. Wir konstruieren nun einen Lauf $r_\rho : \mathrm{pos}(t) \to Q$. Da (v,ρ) die Teilformel *part* erfüllt, gibt es für jede Position $w \in \mathrm{pos}(t)$ genau eine Menge $X_{\vec{q},\sigma,q}$ mit $w \in X_{\vec{q},\sigma,q}$. Also setzen wir $r_\rho(w) := q$. Da (v,ρ) die Teilformel ψ_M^1 erfüllt, ist $t(w) = \sigma$ an jeder Position $w \in X_{\vec{q},\sigma,q}$. Außerdem folgt aus ψ_M^2, dass an jeder Position $w \in X_{\vec{q},\sigma,q}$ die Invariante $r_\rho(w.i) = q_i$ für $i \in [1,m]$ und $\vec{q} = (q_1,\dots,q_m)$ gilt. Deshalb ist r_ρ ein Lauf von M auf t.

Wir führen eine nicht-boolesche Implikation als Abkürzung ein: $\phi \supset \psi := \neg\phi \vee (\phi \wedge \psi)$. Sei *en* eine Positionsaufzählung für Gewichtsbäume und $s \in T_{\Sigma_V}$ ein Baum. Der Wert von $([\![\phi \supset \psi]\!]_V^{en}, s)$ ist $([\![\psi]\!]_V^{en}, s)$, falls $([\![\phi]\!]_V^{en}, s) = 1$ und 1 falls $([\![\phi]\!]_V^{en}, s) = 0$. Wir werden die Implikation $\supset$ deshalb als Filter für die Transitionsgewichte verwenden. Die boolesche

Implikation $\rightarrow$ vermag das nicht zu leisten, da die Gleichung $[\![\neg\neg\phi]\!]_V^{en} = [\![\phi]\!]_V^{en}$ in starken Bimonoiden mit Negation im Allgemeinen nicht gilt.

Der gewichtete Baumautomat M ist schwach, also ist $\mathrm{img}(\gamma) = \{0, 1\}$. Somit gelingt es uns, das Verhalten von γ mithilfe von E und $\dot{\vee}$ in einer booleschen Formel auszudrücken. Wir setzen nun:

$$
\begin{aligned}
\phi &:= \psi_M \;\wedge\; \psi_{wt} \;\wedge\; \psi_\gamma \quad \text{, wobei} \\
\psi_{wt} &:= \forall x. \left(\bigwedge_{(\vec{q},\sigma,q)\in B_M} \big((x \in X_{\vec{q},\sigma,q}) \supset \mu_m(\sigma)_{\vec{q},q}\big) \right) \quad \text{und} \\
\psi_\gamma &:= \mathsf{E}\, z. \left(root(z) \;\wedge \dot{\bigvee_{(\vec{q},\sigma,q)\in B_M}} ((z \in X_{\vec{q},\sigma,q}) \wedge \gamma(q)) \right) \quad \text{mit} \\
root(z) &:= \forall x. \left(\neg edge_1(x, z) \;\wedge\; \ldots \;\wedge\; \neg edge_{max_\Sigma}(x, z)\right) \;.
\end{aligned}
$$

Die Laufformel ψ_M und die Finalgewichtsformel ψ_γ sind boolesch; die Laufgewichtsformel ψ_{wt} hingegen ist eine eingeschränkte Formel. Somit ist ϕ eine Formel aus wRMSO.

Sei nun $t \in T_\Sigma$ beliebig und $r : \mathrm{pos}(t) \rightarrow Q$ ein Lauf von M auf t. Wir haben gezeigt, dass eine Bijektion der Läufe $R_M(t)$ auf die Variablenbelegungen $P_{\psi_M}(t)$ existiert. Also gibt es zu r genau eine (V, t)-Variablenbelegung ρ_r, sodass $(t, \rho_r) \models \psi_M$ und damit $([\![\psi_M]\!], (t, \rho_r)) = 1$ gilt. Da in allen Bäumen ausschließlich die Wurzel ε die boolesche Formel $root(z)$ erfüllt, erhalten wir für ρ_r außerdem:

$$([\![\psi_\gamma]\!], (t, \rho_r)) = 1 \iff \gamma(r(\varepsilon)) = 1 \;.$$

Wir setzen nun

$$\psi'_{wt} := \bigwedge_{(\vec{q},\sigma,q)\in B_M} \big((x \in X_{\vec{q},\sigma,q}) \supset \mu_m(\sigma)_{\vec{q},q}\big) \;.$$

Das Gewicht des Laufes r von M auf t inklusive seines Finalgewichtes berechnet sich durch ϕ wie folgt, wobei $V := \mathrm{Free}(\phi)$ und $s := (t, \rho_r)$.

$$
\begin{aligned}
([\![\phi]\!]^{en}, (t, \rho_r)) &= \prod_{i=1}^{|\mathrm{pos}(s)|} \Big([\![\psi'_{wt}]\!]^{en}_{V\cup\{x\}}, s[x \rightarrow en((\psi'_{wt})^{en}_x(s))(i)]\Big) \cdot \gamma(r(\varepsilon)) \\
&\qquad X_{\vec{q},\sigma,q} \text{ mit } (\vec{q}, \sigma, q) \in B_M \text{ bilden eine Partition, also folgt} \\
&= \prod_{i=1}^{|\mathrm{pos}(t)|} \mu_m(t(p(i)))_{r(p(i).1)\ldots r(p(i).m), r(p(i))} \cdot \gamma(r(\varepsilon)) \\
&\qquad \text{wobei } p(i) := en((\psi'_{wt})^{en}_x(s))(i) \\
&= \mathrm{wt}^{en}_M(t, r) \cdot \gamma(r(\varepsilon))
\end{aligned}
$$

Die Reihenfolge der Faktoren in $([\![\psi_{wt}]\!]^{en}, s)$ entspricht genau der Reihenfolge der Faktoren in $\mathrm{wt}^{en}_M(t, r)$. Wird S durch M beispielsweise schwach TS-erkannt, so wäre $en = TS$.

Im letzten Schritt bilden wir die wREMSO-Formel $\xi := \exists X_1 \ldots \exists X_{|V|}.\phi$, welche das Verhalten des wta M beschreibt. Dabei sei $X_1, \ldots, X_{|V|}$ eine beliebige Aufzählung der

Variablenmenge V. Es bleibt noch zu zeigen, dass $S_M^{en} = [\![\xi]\!]^{en}$. Sei also $t \in T_\Sigma$, dann gilt

$$\begin{aligned}
([\![\xi]\!]^{en}, t) &= \sum_{\rho \text{ ist } (V,t)\text{-VB.}} ([\![\phi]\!]^{en}, (t, \rho)) \\
&= \sum_{\substack{\rho \text{ ist } (V,t)\text{-VB.} \\ (t,\rho) \models \psi_{M,v}}} ([\![\phi]\!]^{en}, (t, \rho)) \\
&= \sum_{r \in R_M(t)} ([\![\phi]\!]^{en}, (t, \rho_r)) \\
&= \sum_{r \in R_M(t)} \mathrm{wt}_M^{en}(t, r) \cdot \gamma(r(\varepsilon)) \;=\; (S_M^{en}, t) \;.
\end{aligned}$$

□

Mit diesem Ergebnis und gemeinsam mit Satz 4.3.12 erhalten wir schließlich, dass die Menge der mit einer Positionsaufzählung für Gewichtsbäume *en* schwach *en*-erkennbaren Potenzreihen durch wRMSO charakterisiert wird. Damit haben wir [18, Theorem 5.5] von Potenzreihen über Worten auf Potenzreihen über Bäumen übertragen und erhalten unser erstes Hauptergebnis.

Satz 4.4.2. *Sei K ein starkes Bimonoid und en eine Positionsaufzählung für Gewichtsbäume. Dann gilt:*

$$K^{en-wrec}\langle\langle T_\Sigma \rangle\rangle = K^{en-rmso}\langle\langle T_\Sigma \rangle\rangle \;.$$

Beweis. Wir verwenden nun einen Ringschluss.

$$\begin{aligned}
K^{en-wrec}\langle\langle T_\Sigma \rangle\rangle &\subseteq K^{en-remso}\langle\langle T_\Sigma \rangle\rangle \qquad \text{(Satz 4.4.1)} \\
&\subseteq K^{en-rmso}\langle\langle T_\Sigma \rangle\rangle \\
&\subseteq K^{en-wrec}\langle\langle T_\Sigma \rangle\rangle \qquad \text{(Satz 4.3.12)}
\end{aligned}$$

□

Nach Lemma 3.2.12 fallen schwach erkennbar und erkennbar zusammen für Positionsaufzählungen für Gewichtsbäume, in welchen die Wurzel stets das letzte Element jedes Gewichtsbaumes ist. Wir erhalten sofort die folgende Aussage.

Korollar 4.4.3. *Sei K ein starkes Bimonoid mit Negation und en eine Positionsaufzählung für Gewichtsbäume mit $en(|\mathrm{pos}(t)|) = \varepsilon$ für alle $t \in T_K$. Dann gilt:*

$$K^{en-rec}\langle\langle T_\Sigma \rangle\rangle = K^{en-rmso}\langle\langle T_\Sigma \rangle\rangle \;.$$

Insbesondere also:

$$K^{TS-rec}\langle\langle T_\Sigma \rangle\rangle = K^{TS-rmso}\langle\langle T_\Sigma \rangle\rangle \;.$$

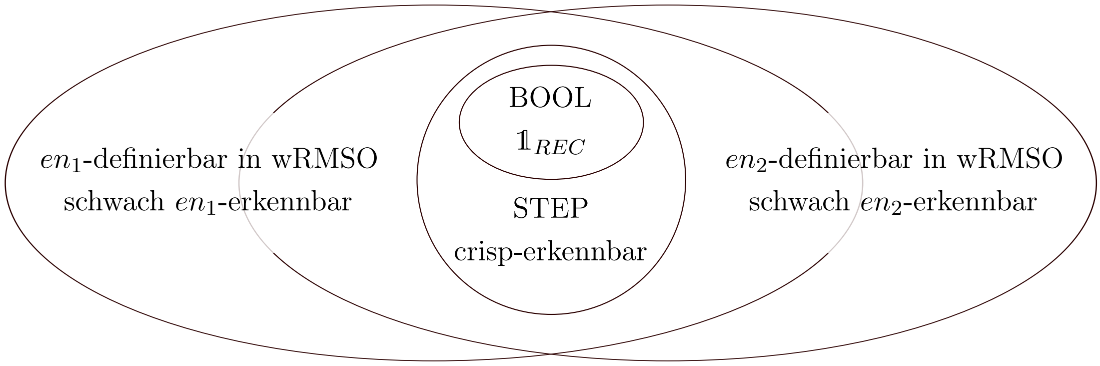


Abbildung 4.2: Klassen schwach erkennbarer Baumreihen

Unser erstes Hauptergebnis lässt sich mit der folgenden Abbildung veranschaulichen.

Betrachten wir nun bi-lokal endliche Bimonoide. Wir haben gezeigt, dass in diesen für die Positionsaufzählungen für Gewichtsbäume TS und $en^<$, wobei $<$ eine lineare Ordnung auf K ist, alle erkennbaren Reihen erkennbare Stufenfunktionen sind. Das begründet nun die Verallgemeinerung von [18, Theorem 5.3] auf Baumreihen, unser zweites Hauptergebnis.

Satz 4.4.4. *Sei K ein bi-lokal endliches starkes Bimonoid mit Negation und $<$ eine lineare Ordnung auf K. Dann gelten:*

- $K^{TS-rec}\langle\langle T_\Sigma\rangle\rangle = K^{TS-emso}\langle\langle T_\Sigma\rangle\rangle$ *und*
- $K^{<-rec}\langle\langle T_\Sigma\rangle\rangle = K^{<-emso}\langle\langle T_\Sigma\rangle\rangle$.

Ist K zudem kommutativ, gelten darüber hinaus:

- $K^{TS-rec}\langle\langle T_\Sigma\rangle\rangle = K^{TS-mso}\langle\langle T_\Sigma\rangle\rangle$ *und*
- $K^{<-rec}\langle\langle T_\Sigma\rangle\rangle = K^{<-mso}\langle\langle T_\Sigma\rangle\rangle$.

Beweis. Sei nun en entweder die Positionsaufzählung TS oder die Positionsaufzählung $en^<$. Wir verwenden wieder einen Ringschluss.

$$\begin{aligned} K^{en-wrec}\langle\langle T_\Sigma\rangle\rangle &\subseteq K^{en-remso}\langle\langle T_\Sigma\rangle\rangle && \text{(Satz 4.4.1)} \\ &\subseteq K^{en-emso}\langle\langle T_\Sigma\rangle\rangle \\ &\subseteq K^{en-wrec}\langle\langle T_\Sigma\rangle\rangle && \text{(Satz 4.3.13, Teil 1)} \end{aligned}$$

Da K bi-lokal endlich ist, erhalten wir mit Satz 3.2.17 die Aussagen des ersten Teils.

Sei K außerdem kommutativ. Dann folgen beide Aussagen des zweiten Teils vollkommen analog aus dem zweiten Teil von Satz 4.3.13. □

Dieser Beweis lässt sich auch führen, indem man im Beweis von Satz 4.4.1 von einem crisp-deterministischen Baumautomaten ausgeht. Diese Automaten erkennen nach Lemma 3.2.9 genau die erkennbaren Stufenfunktionen und laut Satz 3.2.17 ist in bi-lokal endlichen Bimonoiden jede TS- beziehungsweise $<$-erkennbare Baumreihe eine erkennbare Stufenfunktion. Ersetzt man nun in ϕ_{wt} die nicht-boolesche Implikation $\supset$ durch die boolesche Implikation $\rightarrow$, so ist ϕ_{wt} ebenfalls boolesch und ξ somit TS- beziehungsweise $<$-erkennbar.

Als Konsequenz aus Korollar 3.2.18 erhalten wir abschließend, dass in bi-lokal endlichen Bimonoiden die in wEMSO TS-definierbaren mit den in wEMSO $<$-definierbaren Potenzreihen zusammenfallen. Ist das Bimonoid außerdem kommutativ, ergibt sich auch die Gleichheit der in wMSO TS-definierbaren Reihen mit den in wMSO $<$-definierbaren Reihen.

Korollar 4.4.5. *Sei K ein bi-lokales endliches starkes Bimonoid mit Negation und $<$ eine lineare Ordnung auf K. Dann gilt:*

$$K^{TS-emso}\langle\langle T_\Sigma\rangle\rangle = K^{<-emso}\langle\langle T_\Sigma\rangle\rangle \ .$$

Ist K außerdem kommutativ, so gilt darüber hinaus:

$$K^{TS-mso}\langle\langle T_\Sigma\rangle\rangle = K^{<-mso}\langle\langle T_\Sigma\rangle\rangle \ .$$

5 Diskussion

Das letzte Kapitel erweitert die Charakterisierung von erkennbaren formalen Potenzreihen über Wörtern in starken Bimonoiden durch die gewichtete monadische Logik zweiter Stufe [18] auf Bäume. Wir haben gezeigt, dass die mit einer Positionsaufzählung der Gewichtsbäume *en* erkennbaren Baumreihen genau die im Fragment wRMSO *en*-definierbaren Baumreihen sind. Damit verallgemeinern wir die entsprechenden Ergebnisse für erkennbare Baumreihen über Semiringen [17]. Beschränkt man sich auf bi-lokal endliche starke Bimonoide, werden die mit einer linearen Ordnung $<$ und die mit TS erkennbaren Baumreihen von wEMSO charakterisiert. Dies beruht auf der Aussage von Satz 3.2.17, dass alle $<$- beziehungsweise TS-erkennbaren Baumreihen in bi-lokal endlichen starken Bimonoiden erkennbare Stufenfunktionen sind. Wir können Satz 4.4.4 auf alle Positionsaufzählungen der Gewichtsbäume verallgemeinern, die diese Eigenschaft besitzen.

Satz 5.0.6 (Erweiterung Satz 4.4.4)**.** *Sei K ein bi-lokal endliches starkes Bimonoid. Ist en eine Positionsaufzählung derart, dass alle en-erkennbaren Baumreihen aus $K\langle\langle T_\Sigma\rangle\rangle$ erkennbare Stufenfunktionen sind, so gilt:*

$$K^{en-emso}\langle\langle T_\Sigma\rangle\rangle = K^{en-wrec}\langle\langle T_\Sigma\rangle\rangle \ .$$

Ist K zudem kommutativ gilt außerdem:

$$K^{en-mso}\langle\langle T_\Sigma\rangle\rangle = K^{en-wrec}\langle\langle T_\Sigma\rangle\rangle \ .$$

Beweis. (Skizze)
Wir zeigen die Behauptungen vollkommen analog zum Beweis von Satz 4.4.4. In dessen Ringschluss wird Satz 4.3.13 genutzt, wobei *en* die Positionsaufzählung TS oder $en^<$ ist:

$$K^{en-emso}\langle\langle T_\Sigma\rangle\rangle \subseteq K^{en-wrec}\langle\langle T_\Sigma\rangle\rangle \ .$$

Der Beweis dieser Inklusion nutzt im Induktionsschritt die Aussage, dass in bi-lokal endlichen Bimonoiden alle TS- und alle $<$-erkennbaren Baumreihen erkennbare Stufenfunktionen sind. Das gilt per Voraussetzung ebenfalls für *en* und somit gilt die Inklusion auch für beliebige Positionsaufzählungen *en* mit dieser Eigenschaft. □

Es bleibt zu untersuchen, welche Positionsaufzählungen neben TS und $en^<$ diese Eigenschaft besitzen. Sind die Ergebnisse beispielsweise für Tiefensuchen übertragbar, die nicht beim ersten Kind eines Knotens beginnen?

In Lemma 3.2.12 haben wir gezeigt, dass für alle Positionsaufzählungen, in denen die Wurzel stets die letzte Stelle ist, Erkennbarkeit und schwache Erkennbarkeit zusammenfallen. Aus jeder Positionsaufzählung en_1 kann man eine Positionsaufzählung en_2 derart konstruieren, dass die Wurzel stets die letzte Position ist und die sich sonst wie en_1 verhält. Wir definieren dazu für alle $t \in T_K$, $w \in \mathrm{pos}(t)$ und $r = en_1(t)^{-1}(\varepsilon)$

$$en_2(t)^{-1}(w) := \begin{cases} |\mathrm{pos}(t)| & \text{falls } w = \varepsilon \ , \\ en_1(t)^{-1}(w) & \text{falls } en_1(t)^{-1}(w) < r \ , \\ en_1(t)^{-1}(w) - 1 & \text{falls } en_1(t)^{-1}(w) > r \ . \end{cases}$$

Für die derart veränderte Positionsaufzählung gelten offensichtlich Lemma 3.2.12 und Korollar 4.4.3. Gibt es jedoch neben TS Aufzählungen en_1 und nicht-kommutative starke Bimonoide K für die die folgende Gleichheit, wobei en_2 wie oben konstruiert sein möge, gilt?

$$K^{en_1-rec}\langle\langle T_\Sigma \rangle\rangle = K^{en_2-rec}\langle\langle T_\Sigma \rangle\rangle$$

Möglicherweise existieren weitere Fragmente von wMSO, die mit gewissen Eigenschaften von Positionsaufzählungen korrespondieren.

Eine mögliche Abstraktion von der Gewichtsberechnung von Automatenläufen durch Multiplikation haben Droste und Meinecke für gewichtete Wortautomaten vorgeschlagen [14]. Einem Tupel von Gewichten aus einem kommutativen Monoid $(K, +, 0)$ wird dabei nicht das Produkt aller Elemente, sondern das Bild unter einer Bewertungsfunktion $Val : K^+ \to K$ zugeordnet. Diese genüge den Bedingungen $Val((k)) = k$ und $Val((k_1, \ldots, k_n)) = 0 \iff k_1 = 0 \vee \ldots \vee k_n = 0$ für alle $k, k_1, \ldots, k_n \in K$. Es bleibt zu untersuchen, ob die damit gezeigten Charakterisierungen durch eine entsprechend interpretierte wMSO-Logik ebenfalls auf entsprechend gewichtete Baumautomaten übertragbar sind.

In [26] hat Rahonis gewichtete Muller Baumautomaten, die auf unendlichen Bäumen arbeiten, betrachtet. Droste und Vogler haben in [15] die Charakterisierung erkennbarer Wortreihen über starken Bimonoiden durch wMSO-Logik ebenfalls auf unendliche Worte erweitert. Diese Ergebnisse können möglicherweise auch auf erkennbare Potenzreihen über unendlichen Bäumen übertragen werden.

Ebenfalls offen ist die Übertragbarkeit von Kleene-Sätzen für rationale Wortreihen über bi-lokal endlichen starken Bimonoiden [15] auf Baumreihen. Sie würde zu einer Erweiterung der Ergebnisse von Droste und Vogler in [16] führen.

Literaturverzeichnis

[1] Belnap, Nuel: A useful four-valued logic. In: Dunn, M. (Hrsg.) ; Epstein, G. (Hrsg.): *Modern Uses of Multiple-Valued Logic*. Dordrecht: Reidel, 1977 (8-37)

[2] Birkhoff, Garrett: *Colloquium Publications*. Bd. 25: *Lattice Theory*. Third (New) Edition. American Mathematical Society, 1967. – ISBN 0821810251

[3] Birkhoff, Garrett ; Neumann, John v.: The Logic of Quantum Mechanics. In: *The Annals of Mathematics* 37 (1936), Nr. 4, S. 823–843. – ISSN 0003486X

[4] Bollig, Benedikt ; Gastin, Paul: Weighted versus Probabilistic Logics. In: *DLT '09: Proceedings of the 13th International Conference on Developments in Language Theory*. Berlin, Heidelberg : Springer-Verlag, 2009. – ISBN 978–3–642–02736–9, S. 18–38

[5] Borchardt, Björn ; Maletti, Andreas ; Šešelja, Branimir ; Tepavčević, Andreja ; Vogler, Heiko: Cut sets as recognizable tree languages. In: *Fuzzy Sets Syst.* 157 (2006), Nr. 11, S. 1560–1571. – ISSN 0165–0114

[6] Bozapalidis, Symeon: Representatial tree series. In: *Fundamenta Informaticae* 21 (1994), S. 367–389

[7] Büchi, Richard J.: Weak Second-Order Arithmetic and Finite Automata. In: *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik* 6 (1960), Nr. 1-6, S. 66–92

[8] Buchsbaum, Adam L. ; Giancarlo, Raffaele ; Westbrook, Jeffery R.: On the Determinization of Weighted Finite Automata. In: *Automata, Languages and Programming*. 1998, S. 482+

[9] Clark, James ; Murata, Makoto: *Relax NG Specification*. `http://www.relaxng.org/spec-20011203.html`. Version: 12 2001

[10] Culik, Karel ; Kari, Jarkko: Image compression using weighted finite automata. In: *Computers & Graphics* 17 (1993), Nr. 3, S. 305 – 313. – ISSN 0097–8493

[11] Doner, John: Tree acceptors and some of their applications. In: *Journal of Computer and System Sciences* 4 (1970), Nr. 5, S. 406–451. – ISSN 0022–0000

[12] Droste, Manfred ; Gastin, Paul: Weighted automata and weighted logics. In: *Theor. Comput. Sci.* 380 (2007), Nr. 1-2, S. 69–86. – ISSN 0304–3975

[13] Droste, Manfred (Hrsg.) ; Kuich, Werner (Hrsg.) ; Vogler, Heiko (Hrsg.): *Handbook of Weighted Automata*. Springer-Verlag Berlin, 2009 (EATCS Monographs in Theoretical Computer Science)

[14] Droste, Manfred ; Meinecke, Ingmar: Describing Average- and Longtime-Behaviour by Weighted MSO Logics. In: Hliněný, Petr (Hrsg.) ; Kučera, Antonín (Hrsg.): *Mathematical Foundations of Computer Science* Bd. 6281. Springer-Verlag, 2010, S. 537–548

[15] Droste, Manfred ; Stüber, Torsten ; Vogler, Heiko: Weighted finite automata over strong bimonoids. In: *Information Sciences* 180 (2010), Nr. 1, S. 156–166. – ISSN 0020–0255. – Special Issue on Collective Intelligence

[16] Droste, Manfred ; Vogler, Heiko: A Kleene theorem for weighted tree automata. In: *Theory of Computing Systems* 38 (2002), S. 1–38

[17] Droste, Manfred ; Vogler, Heiko: Weighted tree automata and weighted logics. In: *Theor. Comput. Sci.* 366 (2006), Nr. 3, S. 228–247. – ISSN 0304–3975

[18] Droste, Manfred ; Vogler, Heiko: Kleene and Büchi theorems for weighted automata and multi-valued logics over arbitrary bounded lattices. In: Ésik, Zoltán (Hrsg.) ; Fülöp, Zoltán (Hrsg.): *Developments in Language Theory* Bd. 6224. Springer-Verlag Berlin, 2010, S. 160–172

[19] Elgot, Calvin C.: Decision Problems of Finite Automata Design and Related Arithmetics. In: *Transactions of the American Mathematical Society* 98 (1961), Nr. 1, pp. 21-51. `http://www.jstor.org/stable/1993511`. – ISSN 00029947

[20] Fülöp, Zoltán ; Vogler, Heiko: Weighted tree automata and tree transducer. In: *Handbook of Weighted Automata [13]*. Springer-Verlag Berlin, 2009, Kapitel 9, S. 313–400

[21] Gödel, Kurt: Zum intuitionistischen Aussagenkalkül. In: *Anzeiger Akademie der Wissenschaften Wien* 69 (1932), S. 65–66

[22] Gottwald, Siegfried: Many-Valued Logic. Version: Spring 2010, 2010. `http://plato.stanford.edu/archives/spr2010/entries/logic-manyvalued/`. In: Zalta, Edward N. (Hrsg.): *The Stanford Encyclopedia of Philosophy*. Spring 2010. The Metaphysics Research Lab, Center for the Study of Language and Information, Stanford University, 2010

[23] Grätzer, George: *General Lattice Theory*. Second. Basel : Birkhäuser Verlag, 2003. – ISBN 3764369965

[24] Jenei, Sándor: How to construct left-continuous triangular norms–state of the art. In: *Fuzzy Sets and Systems* 143 (2004), Nr. 1, S. 27 – 45. – ISSN 0165–0114. – Advances in Fuzzy Logic

[25] Kuich, Werner: Tree transducers and formal tree series. In: *Acta Cybern.* 14 (1999), Nr. 1, S. 135–149. – ISSN 0324–721X

[26] Rahonis, George: Weighted Muller Tree Automata and Weighted Logics. In: *Journal of Automata, Languages and Combinatorics* 12 (2007), Nr. 4, S. 455–483

[27] SCHÜTZENBERGER, Marcel-Paul: On the definition of a family of automata. In: *Information and Control* 4 (1961), Nr. 2–3, S. 245–270. – ISSN 0019–9958

[28] THATCHER, James W. ; WRIGHT, Jesse B.: Generalized finite automata theory with an application to a decision problem of second-order logic. In: *Theory of Computing Systems* 2 (1968), March, Nr. 1, S. 57–81

[29] VLIST, Eric von d.: *RELAX NG*. O'Reilly Media Inc., 2003. – ISBN 0596004214

Erklärung

Ich versichere, dass ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel angefertigt habe, insbesondere sind wörtliche oder sinngemäße Zitate als solche gekennzeichnet. Mir ist bekannt, dass Zuwiderhandlung auch nachträglich zur Aberkennung des Abschlusses führen kann.

Leipzig, 3. Dezember 2010

Unterschrift