

Institut für Formale Methoden der Informatik

Universität Stuttgart
Universitätsstraße 38
D-70569 Stuttgart

Diplomarbeit Nr. 3722

**Separierbarkeit über endlichen
Wörtern bei einer
Quantorenalternierung
(Separation over finite words for
one quantifier alternation)**

Amir Abdelaziz

Studiengang: Informatik

Prüfer: PD Dr. Manfred Kufleitner

Betreuer: PD Dr. Manfred Kufleitner

Beginn am: 13. April 2015

Beendet am: 13. Oktober 2015

CR-Klassifikation: F.4.3,F.4.1

Kurzfassung

Das Separierbarkeitsproblem entspricht der Fragestellung ob für zwei Mengen X und Y ein sogenannter Separator S existiert mit $X \subseteq S$ und $Y \cap S = \emptyset$. Man sagt dann, dass S die Menge X von Y trennt. Formale Sprachen können durch prädikatenlogische Formeln definiert werden. Ein bekanntes Logikfragment der prädikatenlogischen Formeln ist Σ_2 . Diese Diplomarbeit beschäftigt sich mit der Σ_2 -Separierbarkeit von regulären Sprachen, d.h. mit der Entscheidbarkeit ob für zwei reguläre Sprachen L_1 und L_2 eine dritte Sprache S existiert die durch eine Formel in Σ_2 definiert werden kann und L_1 von L_2 trennt. Grundlage dafür ist der Artikel *Going Higher in the First-Order Quantifier Alternation Hierarchy on Words* von Thomas Place und Marc Zeitoun.

Inhaltsverzeichnis

1	Einleitung	7
2	Grundlagen	8
3	Σ_2 -Ketten	10
4	Kompatible Mengen von Σ_2 -Ketten	11
5	Algorithmus zur Berechnung von Σ_2 -Ketten	13
6	Ehrenfeucht-Fraisse-Spiele	14
7	Simons Factorization Forest Theorem	16
8	Korrektheit des Algorithmus	16
9	Vollständigkeit des Algorithmus	18
10	Zusammenfassung	21

Literaturverzeichnis	23
-----------------------------	-----------

1 Einleitung

Formale Sprachen stellen ein zentrales Gebiet in der theoretischen Informatik dar. Neben der Komplexität der Mittel zur Beschreibung einer Sprache wie etwa die Chomsky-Hierarchie für Grammatiken werden viele verschiedene äquivalente Charakterisierungen einer Klasse von Sprachen verwendet. So ist etwa wohlbekannt, dass die regulären Sprachen nicht nur durch reguläre Grammatiken und reguläre Ausdrücke definiert werden können, sondern auch diejenigen sind, die sich durch einen nichtdeterministischen Automaten oder durch endliche Monoide beschreiben lassen. Eine weitere Möglichkeit zur Beschreibung von formalen Sprachen ist etwa die Prädikatenlogik. Dabei werden Wörter als logische Strukturen und die durch eine Formel erzeugte Sprache als die Menge aller Wörter aufgefasst, die eine Formel erfüllen. So bewies Büchi etwa, dass die regulären Sprachen genau diejenigen sind, welche durch eine Formel in der monadischen Prädikatenlogik zweiter Stufe definiert werden können. Ein weiteres Resultat ist etwa Schützenbergers Theorem, dass die sternfreien Sprachen mit den Sprachen übereinstimmen, welche durch aperiodische Monoide erkannt werden, welche wiederum mit den Sprachen übereinstimmen, die sich durch das Logikfragment der prädikatenlogischen Formeln erster Stufe mit dem zweistelligen Prädikat $<$, welches häufig mit $FO[<]$ oder FO falls die Verwendung des Prädikates $<$ klar ist bezeichnet wird, für die lineare Ordnung ausdrücken lassen [9] [5]. Eine weitere wichtige Hierarchie von Logikfragmenten in FO wird durch die Anzahl der Quantoralternierungen einer Formel $\phi \in FO$ konstruiert. So ist eine Formel im Logikfragment Σ_i , falls sie semantisch äquivalent zu einer Formel in Pränexnormalform mit $i - 1$ alternierenden Quantorenblöcken ist und mit einem existenziellen Quantorenblock beginnt. Analog werden die Fragmente Π_i bestehend aus den Negationen von Formeln Σ_i , sowie $\Delta_i = \Sigma_i \cap \Pi_i$ und $B\Sigma_i$ als boolescher Abschluss von Σ_i definiert. Ein wichtiger Untersuchungsgegenstand in der theoretischen Informatik, ist die Ausdrucksfähigkeit von Logikfragmenten. Nach Simon [10] entspricht etwa $B\Sigma_1$ den stückweisen testbaren Sprachen und eine Charakterisierung von Σ_2 findet sich in [1]. Unabhängig davon stellen Thomas Place und Marc Zeitoun in [8] einen Algorithmus zur Entscheidbarkeit der Mitgliedschaft einer Sprache in Σ_2 vor. Dieser basiert auf einer Verallgemeinerung des Mitgliedschaftsproblems dem sogenannten Separierbarkeitsproblem.

Das Separierbarkeitsproblem beschäftigt sich mit der Fragestellung, ob für zwei Mengen X und Y aus einer Klasse C eine Menge S aus einer Teilklasse $C_0 \subseteq C$ existiert, für die $X \subseteq S$ und $Y \cap S = \emptyset$ gilt [7]. S heißt dann Separator und man sagt, dass S die Menge L_1 von L_2 trennt. Da formale Sprachen über endlichen Wörtern als Teilmengen eines endlich erzeugten freien Monoides A^* über einem endlichen Alphabet A definiert werden, stellt das Separierbarkeitsproblem eine Verallgemeinerung des Mitgliedschaftsproblems einer Sprache $L \subseteq A^*$ zu einer Klasse C_0 dar. Denn es gilt $L \in C_0$ gdw. es einen Separator $S \in C_0$ gibt, der L von seinem Komplement $A^* \setminus L$ trennt. [8] zeigten, dass ein Algorithmus existiert, der für zwei beliebige reguläre Sprachen L_1 und L_2 entscheidet, ob ein Separator $S \in \Sigma_2$ existiert, der L_1 von L_2 trennt. Die Idee des Algorithmus ist es, innerhalb eines Monoides M der zwei regulären Sprachen L_1 und L_2 genug Informationen zu berechnen, um das Σ_2 -Separierbarkeitsproblem lösen zu können. Dazu verwenden sie sogenannte Σ_i -Ketten. In der vorliegenden Arbeit wird der Algorithmus von Thomas Place und Marc Zeitoun [8] vereinfacht, sowie die Beweise für dessen Korrektheit und Vollständigkeit. Des Weiteren werden

einige Beweise die in [8] nicht ausgeführt werden erbracht. In Abschnitt 2 werden benötigte Grundlagen eingeführt. In Abschnitt 3 werden Σ_2 -Ketten und in Abschnitt 4 kompatible Mengen von Σ_2 -Ketten eingeführt. Abschnitt 5 beinhaltet den Algorithmus, der Beweis für die Korrektheit ist Abschnitt 8 und für die Vollständigkeit Abschnitt 9 vorbehalten. Für die Korrektheit benötigen wir Ehrenfeucht-Fraisse-Spiele, die in Abschnitt 6, und für die Vollständigkeit factorization forests, die in Abschnitt 7, eingeführt werden. Die Notation und Beweise halten sich eng an [8].

2 Grundlagen

Wir bezeichnen mit \mathbb{N} immer die Menge der natürlichen Zahlen inklusive der 0. Für jede endliche Halbgruppe S existiert eine kleinste Zahl $\omega \in \mathbb{N} \setminus \{0\}$, der sogenannte Exponent von S , so dass für alle $s \in S$ die Potenz s^ω idempotent ist, d.h. $s^\omega = s^\omega \cdot s^\omega$. Wir schreiben $\omega(S)$ für den Exponenten von S bzw. ω wenn die Halbgruppe S aus dem Kontext bekannt ist.

Mit A bezeichnen wir im Folgenden immer ein endliches Alphabet und mit A^* das endlich erzeugte freie Monoid über A . Eine (formale) **Sprache** ist eine Teilmenge eines endlich erzeugten freien Monoides und deren Elemente heißen **Wörter**. Für ein Wort $w \in A^*$ bezeichne $|w|$ die Länge des Wortes und $\text{alph}(w)$ das Alphabet von w , d.h. die Menge aller Buchstaben aus A die in w mindestens einmal vorkommen.

Eine Sprache $L \subseteq A^*$ heißt **erkennbar** gdw. es einen Monoidhomomorphismus α in ein endliches Monoid M mit $\alpha^{-1}(\alpha(L)) = L$ gibt. Man sagt dann auch, dass M die Sprache L erkennt. Durch Ersetzen von M durch das Monoid $\alpha(A^*)$ in der Definition von α kann o.B.d.A. α als surjektiv angenommen werden. Die regulären Sprachen stimmen genau mit den erkennbaren Sprachen überein und werden zwei reguläre Sprachen L_1 und L_2 durch zwei Monoidhomomorphismen $\alpha_1 : A^* \rightarrow M_1$ und $\alpha_2 : A^* \rightarrow M_2$ erkannt, so erkennt der Monoidhomomorphismus $\alpha : A^* \rightarrow M_1 \times M_2$ mit $\alpha(w) = (\alpha_1(w), \alpha_2(w))$, wobei $M_1 \times M_2$ mit der komponentenweisen Multiplikation versehen ist, sowohl L_1 als auch L_2 . Siehe dazu etwa [2].

Wir bezeichnen mit FO das Logikfragment der prädikatenlogischen Formeln erster Stufe mit einem zweistelligen Prädikatensymbol $<$ und einstelligen Prädikatensymbolen P_a für alle $a \in A$. Statt den atomaren Formeln $P_a(x)$ werden wir auch $\lambda(x) = a$ für alle $a \in A$ schreiben. Die **Quantorentiefe** $qr(\phi)$ einer Formel ϕ ist die Maximaltiefe von ineinander verschachtelten Quantoren die in ϕ vorkommt. So gilt in Gleichung (2) etwa $qr(\phi) = 3$. Genauer wird die Quantorentiefe folgendermaßen definiert.

$$\begin{aligned} qr(\phi) &= 0 \text{ falls } \phi \text{ quantorenfrei} \\ qr(\phi \wedge \psi) &= \max\{qr(\phi), qr(\psi)\} \\ qr(\neg\phi) &= qr(\phi) \\ qr(\exists x\phi) &= qr(\forall x\phi) = qr(\phi) + 1 \end{aligned}$$

Mit Σ_i bezeichnen wir das Logikfragment, dass aus den Formeln in FO besteht, welche eine Pränexnormalform aus i alternierenden Quantorenblöcken beginnend mit einem Existenzblock besitzen. D.h. ψ ist eine Σ_i -Formel falls

$$\psi \equiv Q_{1,1}x_{1,1} \cdots Q_{1,n_1}x_{1,n_1} Q_{2,1}x_{2,1} \cdots Q_{2,n_2}x_{2,n_2} \cdots Q_{i,1}x_{i,1} \cdots Qx_{i,n_i}\psi' \quad (1)$$

gilt, wobei $n_j \in \mathbb{N}$ für alle j , sowie $qr(\psi') = 0$ und $Q_{j,j'} = \exists$ für ungerades j und $Q_{j,j'} = \forall$ für gerades j gilt. So ist z.B.

$$\begin{aligned} \phi &= \exists x \exists y (x < y \wedge \forall z (x < z \implies y < z \vee y = z) \wedge P_a(x) \wedge P_b(y) \wedge \exists u (y < u \wedge P_a(u))) \quad (2) \\ &\equiv \exists x \exists y \exists u \forall z (x < y \wedge (x < z \implies y < z \vee y = z) \wedge P_a(x) \wedge P_b(y) \wedge (y < u \wedge P_a(u))) \end{aligned}$$

eine Σ_2 -Formel. Beachte, dass $\Sigma_i \subseteq \Sigma_{i+1}$ für alle $i \in \mathbb{N}$ gilt, da etwa $n_{i+1} = 0$ gewählt werden kann. Wir definieren des Weiteren $\Sigma_i[k] = \{\phi \in \Sigma_i \mid qr(\phi) = k\}$.

Ein Wort $w \in A^*$ wird als logische Struktur über einer Menge von Positionen $U_w = \{1, \dots, |w|\}$ aufgefasst, wobei für alle $a \in A$ die einstelligen Prädikatensymbole P_a als die Menge der Positionen in w interpretiert werden an denen der Buchstabe a vorkommt und $<$ als die übliche Ordnungsrelation für \mathbb{N} eingeschränkt auf U_w . Für eine Formel ϕ bezeichne $L(\phi) = \{w \in A^* \mid w \models \phi\}$ die von ϕ definierte Sprache. Beachte, dass $L(\phi)$ nur definiert ist falls ϕ eine Aussage ist, d.h. falls keine freien Variablen in ϕ vorkommen. So ist etwa die in Gleichung (2) definierte Sprache $L(\phi) = A^*abA^*aA^*$.

Je nach Kontext bezeichnet man mit Σ_i auch die Menge aller Sprachen $L \subseteq A^*$, die durch eine Σ_i -Formel definierbar sind. Wir schreiben dann $L \in \Sigma_i$ und sagen dass L eine Σ_i -Sprache ist. Analog sagen wir L ist eine $\Sigma_i[k]$ -Sprache und schreiben $L \in \Sigma_i[k]$ falls sie durch eine Formel $\phi \in \Sigma_i[k]$ definierbar ist. Für zwei Formeln $\phi, \psi \in \Sigma_i[k]$ gilt offensichtlich $\phi \vee \psi \in \Sigma_i[k]$ und $\phi \wedge \psi \in \Sigma_i[k]$ mit $L(\phi \vee \psi) = L(\phi) \cup L(\psi)$ und $L(\phi \wedge \psi) = L(\phi) \cap L(\psi)$. Daher ist die Sprachklasse $\Sigma_i[k]$ und damit auch Σ_i bezüglich endlichen Schnitten und Vereinigungen abgeschlossen. Sei $\phi \in \Sigma_i[k]$ und x eine Variable die nicht frei in ϕ vorkommt, dann ist $\phi' = \exists x \phi$ eine Formel mit $\phi' \equiv \phi$ und $\phi' \in \Sigma_i[k+1]$, d.h. für jede Sprache $L \in \Sigma_i[k]$ gilt $L \in \Sigma_i[k+1]$.

Auf A^* wird eine Quasiordnung \leq_i^k eingeführt. Für zwei Wörter $w_1, w_2 \in A^*$ gilt $w_1 \leq_i^k w_2$ gdw. jede Sprache $L \in \Sigma_i[k]$ die w_1 enthält notwendigerweise auch w_2 enthalten muss, d.h. für jede Formel $\phi \in \Sigma_i[k]$ gilt $w_1 \models \phi \implies w_2 \models \phi$. Des Weiteren definieren wir die Äquivalenzrelation \equiv_i^k durch $w_1 \equiv_i^k w_2$ gdw. $w_1 \leq_i^k w_2 \wedge w_2 \leq_i^k w_1$. Offensichtlich gilt dass \leq_i^{k+1} bzw. \equiv_i^{k+1} Verfeinerungen von \leq_i^k bzw. \equiv_i^k sind, d.h. $w_1 \leq_i^{k+1} w_2 \implies w_1 \leq_i^k w_2$ bzw. $w_1 \equiv_i^{k+1} w_2 \implies w_1 \equiv_i^k w_2$ für alle $i, k \in \mathbb{N}$ und $w_1, w_2 \in A^*$. Es ist wohlbekannt, dass es, bei gegebener endlicher Signatur ohne Funktionssymbole, bis auf semantische Äquivalenz nur endlich viele Aussagen $\phi \in FO$ mit $qr(\phi) \leq k$ gibt. Bezeichne $w_1 \equiv_{FO}^k w_2$, die Äquivalenzrelation auf A^* , so dass w_1 und w_2 genau dann äquivalent sind wenn sie dieselben prädikatenlogischen Formeln ϕ erster Stufe mit $qr(\phi) \leq k$ erfüllen dann besitzt \equiv_{FO}^k einen endlichen Index. Siehe dazu etwa [6] [3].

Da $\Sigma_i \subset FO$ für alle $i \in \mathbb{N}$ gilt und \equiv_{FO}^k einen endlichen Index hat, hat nun auch \equiv_i^k einen endlichen Index. Wir bezeichnen mit $[w]_i^k = \{w' \in A^* \mid w' \equiv_i^k w\}$ die Äquivalenzklasse von w . Die Quasiordnung \leq_i^k ist nach Lemma 8 stabil bzgl. der Konkatenation und damit ist

die Verknüpfung $[w_1]_i^k [w_2]_i^k = [w_1 w_2]_i^k$ wohldefiniert und \leq_i^k induziert eine stabile partielle Ordnung \preceq_i^k auf A^*/\equiv_i^k durch $[w_1]_i^k \preceq_i^k [w_2]_i^k$ gdw. $w_1 \leq_i^k w_2$. Falls das Level i der Quantorenalternierung bei den Relationen nicht angegeben wird soll im folgenden immer $i = 2$ angenommen werden.

Die Idee des Algorithmus ist es innerhalb eines Monoides M der zwei reguläre Sprachen L_1 und L_2 erkennt genug Informationen zu berechnen um das Σ_2 -Separierbarkeitsproblem lösen zu können. Dazu werden Σ_2 -Ketten in Abschnitt 3 und kompatible Mengen von Σ_2 -Ketten in Abschnitt 4 eingeführt.

3 Σ_2 -Ketten

Sei $\alpha : A^* \rightarrow M$ ein Monoidhomomorphismus und $k \in \mathbb{N}$. Eine $\Sigma_2[k]$ -**Kette für** α ist ein Element $(s_1, s_2) \in M^2$, wobei M^2 das direkte Produkt mit der komponentenweisen Multiplikation bezeichne, für das zwei Wörter $w_1, w_2 \in A^*$ existieren, so dass $\alpha(w_1) = s_1$ und $\alpha(w_2) = s_2$ und $w_1 \leq_2^k w_2$ gelten. Die Menge aller $\Sigma_2[k]$ -Ketten bezeichnen wir mit $\mathcal{C}^k[\alpha]$ und für $s \in \mathcal{C}^k[\alpha]$ sei $s \in \mathcal{C}^k[\alpha, B]$ für ein Alphabet $B \subseteq A$ gdw. w_1 und w_2 mit der Bedingung $\mathbf{alph}(w_1) = \mathbf{alph}(w_2) = B$ gewählt werden können.

Ein Element $s \in M^2$ heißt Σ_2 -**Kette für** α falls $s \in \mathcal{C}^k[\alpha]$ für alle k und wir schreiben $\mathcal{C}[\alpha]$ für die Menge der Σ_2 -Ketten für α , d.h. $\mathcal{C}[\alpha] = \bigcap_k \mathcal{C}^k[\alpha]$. Des Weiteren definieren wir $\mathcal{C}[\alpha, B] = \bigcap_k \mathcal{C}^k[\alpha, B]$.

Proposition 1. *Sei $k \geq 1$, dann gilt $\mathcal{C}^k[\alpha] = \bigcup_{B \subseteq A} \mathcal{C}^k[\alpha, B]$. Des Weiteren gilt $\mathcal{C}[\alpha] = \bigcup_{B \subseteq A} \mathcal{C}[\alpha, B]$.*

Beweis. Nach Definition gilt $\mathcal{C}^k[\alpha, B] \subseteq \mathcal{C}^k[\alpha]$ für alle $B \subseteq A$ und $k \in \mathbb{N}$ und damit $\mathcal{C}[\alpha, B] \subseteq \mathcal{C}[\alpha]$ für alle $B \subseteq A$. Es verbleibt also $\mathcal{C}^k[\alpha] \subseteq \bigcup_{B \subseteq A} \mathcal{C}^k[\alpha, B]$ für $k \geq 1$ und $\mathcal{C}[\alpha] \subseteq \bigcup_{B \subseteq A} \mathcal{C}[\alpha, B]$ zu zeigen.

Sei nun $k \geq 1$ und $(s_1, s_2) \in \mathcal{C}^k[\alpha]$, d.h. es existieren nach Definition von $\mathcal{C}^k[\alpha]$ zwei Wörter $w_1, w_2 \in A^*$ mit $\alpha(w_1) = s_1$ sowie $\alpha(w_2) = s_2$ und $w_1 \leq_2^k w_2$. Aus $w_1 \leq_2^k w_2$ folgt sofort, dass ein Alphabet $B \subseteq A$ existiert mit $\mathbf{alph}(w_1) = \mathbf{alph}(w_2) = B$. Denn wäre $\mathbf{alph}(w_1) \neq \mathbf{alph}(w_2)$ dann ließe sich im Widerspruch zur Definition von \leq_2^k eine Formel $\phi \in \Sigma_2$ durch $\phi = \bigwedge_{a \in \mathbf{alph}(w_1)} \exists x P_a(x) \wedge \bigwedge_{a \in A \setminus \mathbf{alph}(w_1)} \neg \exists x P_a(x)$ mit $qr(\phi) = 1$ konstruieren, für die $w_1 \models \phi$ und $w_2 \not\models \phi$ gilt. Damit gilt $(s_1, s_2) \in \mathcal{C}^k[\alpha, B]$ für ein Alphabet $B \subseteq A$ und damit $\mathcal{C}^k[\alpha] \subseteq \bigcup_{B \subseteq A} \mathcal{C}^k[\alpha, B]$.

Sei nun $(s_1, s_2) \in \mathcal{C}[\alpha]$. Dann gilt nach Definition $\forall k \in \mathbb{N} \quad (s_1, s_2) \in \mathcal{C}^k[\alpha]$ und damit existiert für alle $k \geq 1$ ein $B_k \subseteq A$ derart, dass $(s_1, s_2) \in \mathcal{C}^k[\alpha, B_k]$ gilt. Da A ein endliches Alphabet ist, gilt auch dass ihre Potenzmenge 2^A endlich ist und nach Schubfachschluss muss nun ein Alphabet $B \subseteq A$ existieren mit $B = B_k$ für unendlich viele k . Sei dieses B nun gemäß dieser Eigenschaft fest. Nach Definition von \leq_i^k folgt offensichtlich $w_1 \leq_2^{k+1} w_2 \implies w_1 \leq_2^k w_2$, d.h. $\mathcal{C}^{k+1}[\alpha, B] \subseteq \mathcal{C}^k[\alpha, B]$, für alle $k \in \mathbb{N}$ und $w_1, w_2 \in A^*$ und damit gilt $(s_1, s_2) \in \mathcal{C}[\alpha, B]$. \square

Für ein Monoid M ist M^2 mit der komponentenweisen Verknüpfung ein Monoid. Analog definieren wir eine Verknüpfung \cdot in 2^{M^2} durch $S \cdot T = \{st \in M^2 \mid s \in S, t \in T\}$ für alle $S, T \in 2^{M^2}$. Dadurch wird 2^{M^2} wieder zum Monoid mit $\{(1, 1)\}$ als neutralem Element.

Proposition 2. *Sei $k \in \mathbb{N}$ und $C, D \subseteq A$ beliebig aber fest und $\alpha : A^* \rightarrow M$ ein Monoidhomomorphismus dann gilt*

$$\mathcal{C}^k[\alpha, C] \cdot \mathcal{C}^k[\alpha, D] \subseteq \mathcal{C}^k[\alpha, C \cup D] \quad (3)$$

sowie dass $\mathcal{C}^k[\alpha]$ ein Untermonoid und für alle $B \subseteq A$ die Mengen $\mathcal{C}^k[\alpha, B]$ Unterhalbgruppen von M^2 sind. Analog gilt

$$\mathcal{C}[\alpha, C] \cdot \mathcal{C}[\alpha, D] \subseteq \mathcal{C}[\alpha, C \cup D] \quad (4)$$

sowie dass $\mathcal{C}[\alpha]$ ein Untermonoid und für alle $B \subseteq A$ die Mengen $\mathcal{C}[\alpha, B]$ Unterhalbgruppen von M^2 sind.

Beweis. Sei $(s_1, s_2) \in \mathcal{C}^k[\alpha, C]$, $(s'_1, s'_2) \in \mathcal{C}^k[\alpha, D]$ dann existieren $w_1, w_2, w'_1, w'_2 \in A^*$ mit $\text{alph}(w_1) = \text{alph}(w_2) = C$ und $\text{alph}(w'_1) = \text{alph}(w'_2) = D$ und $w_1 \leq_2^k w_2$ und $w'_1 \leq_2^k w'_2$ und $\alpha(w_1) = s_1$ und $\alpha(w_2) = s_2$ und $\alpha(w'_1) = s'_1$ und $\alpha(w'_2) = s'_2$. Nach Lemma 8 gilt dann $w_1 w'_1 \leq_2^k w_2 w'_2$ und da α ein Homomorphismus ist $\alpha(w_1 w'_1) = \alpha(w_1) \alpha(w'_1) = s_1 s'_1$ und $\alpha(w_2 w'_2) = \alpha(w_2) \alpha(w'_2) = s_2 s'_2$. Damit gilt $(s_1, s_2)(s'_1, s'_2) = (s_1 s'_1, s_2 s'_2) \in \mathcal{C}^k[\alpha]$ und aus $\text{alph}(w_1 w'_1) = \text{alph}(w_2 w'_2) = C \cup D$ folgt $(s_1 s'_1, s_2 s'_2) \in \mathcal{C}^k[\alpha, C \cup D]$. Sei jetzt $B \subseteq A$ und $s, s' \in \mathcal{C}^k[\alpha, B]$ dann folgt nun $ss' \in \mathcal{C}^k[\alpha, B]$ und damit das $\mathcal{C}^k[\alpha, B]$ abgeschlossen ist. Die Verknüpfung in dem Monoid M ist nach Definition assoziativ und damit auch die Verknüpfung im direkten Produkt M^2 sowie deren Einschränkung auf $\mathcal{C}^k[\alpha, B]$. Daraus folgt das $\mathcal{C}^k[\alpha, B]$ eine Halbgruppe ist. Wir zeigen nun, dass $\mathcal{C}^k[\alpha]$ ein Monoid ist. Sei $k = 0$, dann ist $\Sigma_2[k] = \{\emptyset, A^*\}$, da die Tautologie (leere Konjunktion) und deren Negation (leere Disjunktion) die einzigen Aussagen in $\Sigma_2[k]$ sind. Damit ist $\mathcal{C}^0[\alpha] = \alpha(A^*) \times \alpha(A^*)$, wobei $\alpha(A^*)$ als homomorphes Bild eines Monoides ein Monoid ist und damit auch das direkte Produkt $\mathcal{C}^0[\alpha]$ ein Monoid ist. Für $k \geq 1$ gilt nach Proposition 1 $\mathcal{C}^k[\alpha] = \bigcup_{B \subseteq A} \mathcal{C}^k[\alpha, B]$ und damit ist $\mathcal{C}^k[\alpha]$ eine Halbgruppe. Da α ein Monoidhomomorphismus ist gilt für das leere Wort $\epsilon \in \alpha^{-1}(1_M) \neq \emptyset$ und damit ist offensichtlich das neutrale Element $(1_M, 1_M) \in \mathcal{C}^k[\alpha]$ und $\mathcal{C}^k[\alpha]$ ein Monoid. Sei nun $s \in \mathcal{C}[\alpha, C]$, $s' \in \mathcal{C}[\alpha, D]$, dann gilt für alle k nach Definition $s \in \mathcal{C}^k[\alpha, C]$ und $s' \in \mathcal{C}^k[\alpha, D]$ und damit $ss' \in \mathcal{C}^k[\alpha, C \cup D]$. Nach Definition folgt nun $ss' \in \mathcal{C}[\alpha, C \cup D]$. Analog wird nun $\mathcal{C}[\alpha, B]$ für alle $B \subseteq A$ wiederum zur Halbgruppe und $\mathcal{C}[\alpha]$ zu einem Monoid. \square

4 Kompatible Mengen von Σ_2 -Ketten

Eine Menge von $\Sigma_2[k]$ -Ketten $T \subseteq \mathcal{C}^k[\alpha, B]$ heißt **kompatibel** falls gilt: Es existiert ein $w_1 \in A^*$, so dass für alle $(s_1, s_2) \in T$ ein Wort $w_2 \in A^*$ existiert mit $w_1 \leq_2^k w_2$ und $\alpha(w_1) = s_1$ und $\alpha(w_2) = s_2$ und $\text{alph}(w_1) = \text{alph}(w_2) = B$.

Damit „synchronisieren“ wir die $\Sigma_2[k]$ -Ketten in der ersten Komponente. Aus dieser Definition folgt sofort das alle $\Sigma_2[k]$ -Ketten in T dieselbe erste Komponente s_1 haben und dass jede Teilmenge $T' \subseteq T$ selbst wieder kompatibel ist.

Wir definieren des Weiteren die Menge der kompatiblen Mengen von $\Sigma_2[k]$ -Ketten (für ein Alphabet B) durch

$$\mathfrak{C}^k[\alpha, B] = \{T \subseteq \mathcal{C}^k[\alpha, B] \mid T \text{ kompatibel}\} \quad (5)$$

$$\mathfrak{C}^k[\alpha] = \{T \subseteq \mathcal{C}^k[\alpha] \mid T \text{ kompatibel}\} \quad (6)$$

Analog heißt eine Menge von Σ_2 -Ketten $T \subseteq \mathcal{C}[\alpha, B]$ **kompatibel** falls $T \in \mathfrak{C}^k[\alpha, B]$ für alle k und wir definieren

$$\mathfrak{C}[\alpha, B] = \bigcap_k \mathfrak{C}^k[\alpha, B] \quad (7)$$

$$\mathfrak{C}[\alpha] = \bigcap_k \mathfrak{C}^k[\alpha] \quad (8)$$

Proposition 3. *Sei $k \geq 1$, dann gilt $\mathfrak{C}^k[\alpha] = \bigcup_{B \subseteq A} \mathfrak{C}^k[\alpha, B]$. Des Weiteren gilt $\mathfrak{C}[\alpha] = \bigcup_{B \subseteq A} \mathfrak{C}[\alpha, B]$.*

Beweis. Der Beweis wird analog zum Beweis in Proposition 1 für Σ_2 -Ketten geführt. Für alle $B \subseteq A$ und $k \in \mathbb{N}$ folgt aus $\mathcal{C}^k[\alpha, B] \subseteq \mathcal{C}^k[\alpha]$, dass $\mathfrak{C}^k[\alpha, B] \subseteq \mathfrak{C}^k[\alpha]$ gilt und damit $\mathfrak{C}[\alpha, B] \subseteq \mathfrak{C}[\alpha]$ für alle $B \subseteq A$. Es verbleibt also $\mathfrak{C}^k[\alpha] \subseteq \bigcup_{B \subseteq A} \mathfrak{C}^k[\alpha, B]$ für $k \geq 1$ und $\mathfrak{C}[\alpha] \subseteq \bigcup_{B \subseteq A} \mathfrak{C}[\alpha, B]$ zu zeigen. Sei $k \geq 1$ und $T \in \mathfrak{C}^k[\alpha]$, nach Definition gilt, dass T kompatibel und $T \subseteq \mathcal{C}^k[\alpha]$ ist. Nach Proposition 1 folgt, dass ein $B \subseteq A$ existiert mit $T \in \mathcal{C}^k[\alpha, B]$ und damit gilt nach Definition $T \in \mathfrak{C}^k[\alpha, B]$. Sei also nun $T \in \mathfrak{C}[\alpha]$, dann gilt $T \in \mathfrak{C}^k[\alpha]$ für alle $k \in \mathbb{N}$ und damit existieren nun wieder für alle $k \geq 1$ Alphabete B_k mit $T \in \mathfrak{C}^k[\alpha, B_k]$. Nach Schubfachschluss gilt wieder, dass ein $B \subseteq A$ existiert, so dass $B = B_k$ für unendliche viele k . Sei $B \subseteq A$ nun wieder gemäß dieser Eigenschaft fest. Für alle $k \in \mathbb{N}$ gilt $\mathcal{C}^{k+1}[\alpha, B] \subseteq \mathcal{C}^k[\alpha, B]$ und damit $\mathfrak{C}^{k+1}[\alpha, B] \subseteq \mathfrak{C}^k[\alpha, B]$ und damit $T \in \mathfrak{C}[\alpha, B]$. \square

Analog zur Verknüpfung \cdot in 2^{M^2} definieren wir nun die Verknüpfung \cdot in $2^{2^{M^2}}$ durch $S \cdot T = \{st \in 2^{M^2} \mid s \in S, t \in T\}$ für alle $S, T \in 2^{2^{M^2}}$. Dadurch wird $2^{2^{M^2}}$ wieder zum Monoid und analog zu Proposition 2 gilt nun Proposition 4.

Proposition 4. *Sei $k \in \mathbb{N}$ und $C, D \subseteq A$ beliebig aber fest und $\alpha : A^* \rightarrow M$ ein Monoidhomomorphismus dann gilt*

$$\mathfrak{C}^k[\alpha, B] \cdot \mathfrak{C}^k[\alpha, C] \subseteq \mathfrak{C}^k[\alpha, B \cup C] \quad \mathfrak{C}[\alpha, B] \cdot \mathfrak{C}[\alpha, C] \subseteq \mathfrak{C}[\alpha, B \cup C]$$

Des Weiteren gilt, dass $\mathfrak{C}^k[\alpha, B]$ und $\mathfrak{C}[\alpha, B]$ Unterhalbgruppen und $\mathfrak{C}^k[\alpha]$ und $\mathfrak{C}[\alpha]$ Untermonoide von $2^{2^{M^2}}$ sind.

Beweis. Folgt direkt aus den Definitionen in den Gleichungen (5) bis (8) und Proposition 2 analog zu Beweis in Proposition 2. Dabei wird verwendet, dass für zwei kompatible Mengen T, T' die mit w bzw. w' synchronisiert werden ww' zur Synchronisierung der Ketten in $T \cdot T'$ verwendet werden kann. \square

Wir definieren den Abschluss einer Menge \mathfrak{T} bezüglich der Mengeninklusion durch

$$\downarrow \mathfrak{T} = \{T \mid \exists S \in \mathfrak{T}, T \subseteq S\}$$

Offenbar gilt dass alle Mengen aus den Definitionen in den Gleichungen (5) bis (8) bereits bezüglich der Mengeninklusion abgeschlossen sind.

Theorem 5. Sei $\alpha : A^* \rightarrow M$ ein Monoidhomomorphismus der zwei reguläre Sprachen L_1, L_2 erkennt dann gilt: L_1 ist genau dann von L_2 durch eine Sprache $S \in \Sigma_2$ separierbar wenn $(\alpha(L_1) \times \alpha(L_2)) \cap \mathcal{C}[\alpha] = \emptyset$

Beweis. Sei $(s_1, s_2) \in (\alpha(L_1) \times \alpha(L_2)) \cap \mathcal{C}[\alpha]$ und S ein Separator der durch eine Σ_2 -Formel der Quantorentiefe k für ein beliebiges $k \in \mathbb{N}$ definiert wird. Aus der Definition von $\mathcal{C}[\alpha]$ folgt das zwei Wörter $w_1, w_2 \in A^*$ mit $w_1 \leq_2^k w_2$ und $\alpha(w_1) = s_1$ und $\alpha(w_2) = s_2$ existieren. Da α die beiden regulären Sprachen L_1 und L_2 erkennt folgt $w_1 \in L_1$ und $w_2 \in L_2$. Da $L_1 \subseteq S$ gelten soll folgt $w_1 \in S$ und damit nach Definition von \leq_2^k automatisch $w_2 \in S \cap L_2 \neq \emptyset$ im Widerspruch zur Annahme das ein Separator existiert. Um die Rückrichtung zu zeigen nehmen wir nun $(\alpha(L_1) \times \alpha(L_2)) \cap \mathcal{C}[\alpha] = \emptyset$ an. Nach Definition der Erkennbarkeit von Sprachen gilt $|M| < \infty$ und damit $|\alpha(L_1)| < \infty$ und $|\alpha(L_2)| < \infty$. Sei also o.B.d.A. $\alpha(L_1) = \{s_1, \dots, s_n\}$ und $\alpha(L_2) = \{t_1, \dots, t_m\}$. Wir zeigen, dass sich für beliebige $1 \leq i \leq n$ und $1 \leq j \leq m$ eine Σ_2 -Sprache S_{ij} existiert die $\alpha^{-1}(s_i)$ von $\alpha^{-1}(t_j)$ trennt. Da Σ_2 gegenüber endlichen Schnitten und Vereinigungen abgeschlossen ist gilt offensichtlich dass $\bigcup_{1 \leq i \leq n} \bigcap_{1 \leq j \leq m} S_{ij}$ eine Σ_2 -Sprache ist die L_1 von L_2 trennt. Seien also $1 \leq i \leq n$ und $1 \leq j \leq m$ fest. Da $(s_i, t_j) \notin \mathcal{C}[\alpha]$ folgt das ein $k \in \mathbb{N}$ existiert so dass für alle Wörter $w \in \alpha^{-1}(s_i)$ und $w' \in \alpha^{-1}(t_j)$ die Beziehung $w \not\leq_2^k w'$ gilt und damit nach Definition von \leq_2^k , dass eine Sprache $S_{[w]_2^k [w']_2^k} \in \Sigma_2$ mit $w \in S_{[w]_2^k [w']_2^k}$ und $w' \notin S_{[w]_2^k [w']_2^k}$ existiert. Da es nur endlich viele Äquivalenzklassen gibt lässt sich der Separator $S_{ij} \in \Sigma_2$ als endliche boolesche Kombination

$$S_{ij} = \bigcup_{\{[w]_2^k \mid w \in \alpha^{-1}(s_i)\}} \bigcap_{\{[w']_2^k \mid w' \in \alpha^{-1}(t_j)\}} S_{[w]_2^k [w']_2^k}$$

angeben. □

5 Algorithmus zur Berechnung von Σ_2 -Ketten

Sei $\alpha : A^* \rightarrow M$ ein Monoidhomomorphismus der zwei reguläre Sprachen L_1 und L_2 erkennt. Nach Theorem 5 reicht es aus $\mathcal{C}[\alpha]$ zu berechnen um das Separierbarkeitsproblem für Σ_2 zu entscheiden. Der hier vorgestellte Algorithmus berechnet stattdessen $\mathfrak{C}[\alpha, B]$ für alle $B \subseteq A$. Dies genügt jedoch, da $\mathcal{C}[\alpha] = \bigcup_{B \subseteq A} \mathfrak{C}[\alpha, B]$ nach Proposition 1 gilt und $s \in \mathfrak{C}[\alpha, B]$ gdw. $\{s\} \in \mathfrak{C}[\alpha, B]$. Der Algorithmus ist ein Fixpunktalgorithmus und startet mit Mengen von trivialen kompatiblen Mengen $\mathfrak{J}_B^0 \subseteq \mathfrak{C}[\alpha, B]$ der Form $\mathfrak{J}_B^0 = \{\{(\alpha(w), \alpha(w))\} \mid \text{alph}(w) = B\}$

für alle $B \subseteq A$. Dann werden iterativ für $j \geq 0$ die Mengen \mathfrak{J}_B^{j+1} durch $\mathfrak{J}_B^{j+1} = \mathfrak{J}_B^j \cup M_B^j \cup D_B^j$ mit

$$M_B^j = \bigcup_{C \cup D = B} \mathfrak{J}_C^j \cdot \mathfrak{J}_D^j \quad (9)$$

$$D_B^j = \left\{ T^\omega \cdot \{(1, \alpha(w)) \mid \mathbf{alph}(w) = B\} \cdot T^\omega \mid T \in \mathfrak{J}_B^j \right\} \quad (10)$$

berechnet. Dabei bezeichne ω den Exponenten der Halbgruppe 2^{M^2} . Offensichtlich gilt $|2^{M^2}| < \infty$ und $\mathfrak{J}_B^j \subseteq \mathfrak{J}_B^{j+1} \subseteq 2^{M^2}$ für alle $j \geq 0$ und $B \subseteq A$. Daher existiert für alle $B \subseteq A$ ein $j \geq 0$, so dass $\mathfrak{J}_B^{j+1} = \mathfrak{J}_B^j$ und von nun an bezeichnen wir diese Mengen mit \mathfrak{J}_B^* .

Theorem 6. *Sei $l \geq 3 \cdot |M| \cdot 2^{|A|} \cdot 2^{2^{|M|^2}}$. Dann gilt*

$$\mathfrak{C}[\alpha, B] = \mathfrak{C}^l[\alpha, B] = \downarrow \mathfrak{J}_B^*$$

Theorem 6 behauptet also sowohl die Vollständigkeit als auch die Korrektheit des Algorithmus. Für die Korrektheit genügt es $\downarrow \mathfrak{J}_B^* \subseteq \mathfrak{C}[\alpha, B]$ zu zeigen, d.h. dass der Algorithmus tatsächlich nur Mengen kompatibler Mengen von Σ_2 -Ketten berechnet. Da $\mathfrak{C}[\alpha, B] \subseteq \mathfrak{C}^l[\alpha, B]$ nach der Definition in Gleichung (7) für alle $l \in \mathbb{N}$ und $B \subseteq A$ gilt, genügt es für die Vollständigkeit nur noch $\mathfrak{C}^l[\alpha, B] \subseteq \downarrow \mathfrak{J}_B^*$ für $l \geq 3 \cdot |M| \cdot 2^{|A|} \cdot 2^{2^{|M|^2}}$ zu zeigen. In Abschnitt 8 wird die Korrektheit und in Abschnitt 9 die Vollständigkeit bewiesen.

6 Ehrenfeucht-Fraïsse-Spiele

Die Ausdrucksfähigkeit von Logikfragmenten kann durch sogenannte Ehrenfeucht-Fraïsse-Spiele ausgedrückt werden. Siehe dazu auch [6] [3]. Wir definieren hier Ehrenfeucht-Fraïsse-Spiele für die Logikfragmente Σ_i , wie sie in Abschnitt 2 definiert wurden.

Sei $i \in \mathbb{N}$ fest. Das Ehrenfeucht-Fraïsse-Spiel für Σ_i wird von zwei Spielern, im folgenden Spieler I und Spieler II genannt, auf zwei Wörtern $w, w' \in A^*$ gespielt. Jeder der beiden Spieler hat k Marken zur Verfügung, die sie in k Runden auf Positionen in den beiden Wörtern verteilen. In jeder Runde ist eines der beiden Wörter w, w' das sogenannte aktive Wort. Des Weiteren existiert ein Zähler c für die Quantorenalternierung der zu Beginn auf 0 gesetzt wird. Jede Runde j läuft folgendermaßen ab

- Falls $c < i - 1$ ist darf Spieler I, das aktive Wort austauschen, d.h. in diesem Fall wird, falls w aktiv war nun w' aktiv bzw. umgekehrt und c wird um 1 inkrementiert.
- Spieler I sucht falls w aktiv ist eine Position x_j in w bzw. falls w' aktiv ist eine Position x'_j in w' und legt seine Marke darauf.
- Spieler II legt analog im Wort das nicht aktiv ist seine Marke an eine Position x'_j in w' bzw. x_j in w .

Falls nach der j -ten Runde für alle $l_1, l_2 \leq j$ die Bedingungen $\lambda(x_{l_1}) = \lambda(x'_{l_1})$ und $x_{l_1} < x_{l_2}$ gdw. $x'_{l_1} < x'_{l_2}$ gelten wird weitergespielt, sonst gewinnt Spieler I. Spieler II gewinnt falls Spieler I nach k Runden noch nicht gewonnen hat.

Lemma 7. Für alle $k, i \in \mathbb{N}$ und $w, w' \in A^*$ gilt $w \leq_i^k w'$ genau dann wenn Spieler II eine Gewinnstrategie für ein Σ_i -Spiel über k Runden besitzt, wobei w zu Anfang das aktive Wort ist.

Beweis. Siehe etwa [6], [3] □

Lemma 8. Sei $k, i \in \mathbb{N}$ und $w_1, w_2, w'_1, w'_2 \in A^*$ so dass $w_1 \leq_i^k w_2$ und $w'_1 \leq_i^k w'_2$ dann gilt $w_1 w'_1 \leq_i^k w_2 w'_2$.

Beweis. Nach Lemma 7 hat Spieler II Gewinnstrategien für die Spiele für w_1 und w_2 bzw. w'_1 und w'_2 mit w_1 bzw. w'_1 als zum Anfang aktives Wort. Spieler II kann diese beiden Strategien zu einer Gewinnstrategie für das Spiel für die beiden Wörter $w_1 w'_1$ und $w_2 w'_2$ mit $w_1 w'_1$ als aktivem Wort kombinieren. □

Lemma 9. Seien $k, k_1, k_2 \in \mathbb{N}$ mit $k_1, k_2 \geq 2^k - 1$. Sei $v \in A^*$ dann gilt

$$\forall i \in \mathbb{N} \quad v^{k_1} \leq_i^k v^{k_2}$$

Beweis. Siehe etwa [12] □

Lemma 10. Sei $i \in \mathbb{N}$, sei $k, l, r, l', r' \in \mathbb{N}$ derart, dass $l, r, l', r' \geq 2^k$ und seien $u, v \in A^*$ derart, dass $u \leq_i^k v$. Dann gilt

$$v^l v^r \leq_{i+1}^k v^{l'} u v^{r'}$$

Beweis. Wir zeigen $v^l v^r \leq_{i+1}^k v^{l'} u v^{r'}$ durch ein Ehrenfeucht-Fraïsse Argument. Der Beweis benutzt eine Induktion über k und eine Fallunterscheidung nach dem Wert des Zählers c nach der ersten Runde. Sei nun $w = v^l v^r$ und $w' = v^{l'} u v^{r'}$.

Fall 1: Sei nach der ersten Runde $c = 1$. Dann hat Spieler I w' ausgewählt und es genügt $w' \leq_i^k w$ zu zeigen. Wir wissen, dass $u \leq_i^k v$ gilt und aus Lemma 9 folgt $v^{l'} \leq_i^k v^l$ und $v^{r'} \leq_i^k v^{r-1}$ und damit nach Lemma 8 $w' \leq_i^k w$.

Fall 2: Sei nach der ersten Runde $c = 0$. Dies bedeutet, dass Spieler I an eine Position x in w gespielt hat. Damit gilt, dass x in einer Kopie von v ist. w enthält mindestens 2^{k+1} Kopien von v und aufgrund der Symmetrie können wir o.B.d.A. annehmen, dass rechts von x noch mindestens 2^k Kopien von v vorhanden sind. Wir definieren also nun die Antwort x' von Spieler II in w' . Wir wählen x' so, dass es zu einer Kopie v in w' gehört und an derselben relativen Position in v wie x in v steht. D.h. wir müssen noch die Kopie von v bestimmen in der x'

liegt. Sei n die Anzahl der Kopien von v links von x , d.h. x ist in der $n + 1$ Kopie von v in w . Falls $n < 2^{k-1} - 1$ ist, dann liegt x' in der $n + 1$ -ten Kopie von v in w' . Andernfalls liegt x' in der 2^{k-1} -ten Kopie von v . Diese Kopien existieren immer da $l' \geq 2^k$ ist. Sei $w = w_p v w_q$ und $w' = w'_p v w'_q$, wobei die Faktoren v den Kopien entsprechen in denen x bzw. x' liegen. Wir müssen also nur noch $w_p \leq_{i+1}^{k-1} w'_p$ und $w_q \leq_{i+1}^{k-1} w'_q$ zeigen. Falls $n < 2^{k-1} - 1$ gilt, dann gilt nach Definition $w_p = w'_p$ und damit offensichtlich $w_p \leq_{i+1}^{k-1} w'_p$. Andernfalls bestehen w_p und w'_p aus einer Konkatenation von mindestens $2^{k-1} - 1$ Kopien von v und damit gilt nach Lemma 9 $w_p \leq_{i+1}^{k-1} w'_p$. Des Weiteren gilt $w_q = v^{l_1} v^r$ und $w'_q = v^{l'_1} v^{r'}$ mit $l_1 + r \geq 2^k$ und $l'_1, r' \geq 2^{k-1}$. Damit gilt nach Induktionsannahme $w_q \leq_{i+1}^{k-1} w'_q$. \square

7 Simons Factorization Forest Theorem

In diesem Abschnitt wird Simons Factorization Forest Theorem vorgestellt. Beweise finden sich etwa in [11] [4].

Sei M ein endliches Monoid und $\alpha : A^* \rightarrow M$ ein Monoidhomomorphismus. Ein α -factorization Forest ist ein geordneter Baum dessen Knoten mit Wörtern aus A^* beschriftet sind, so dass für jeden inneren Knoten x mit Kindern x_1, \dots, x_n gilt, dass wenn die Kinder mit w_1, \dots, w_n beschriftet sind, dass dann x mit $w = w_1 \cdots w_n$ beschriftet ist. Die Knoten lassen sich in drei Typen einteilen.

- Blätter, welche mit einem Buchstaben $a \in A$ oder dem leeren Wort beschriftet sind.
- Binäre Knoten welche genau zwei Kinder haben.
- Idempotente Knoten dürfen eine beliebige Anzahl an Knoten haben, derart, dass für deren Beschriftungen w_1, \dots, w_n die Bedingung $\alpha(w_1) = \cdots = \alpha(w_n) = e$ für ein idempotentes $e \in M$ erfüllt ist.

Ein α -factorization Forest für w ist ein α -factorization Forest dessen Wurzel mit w beschriftet ist.

Theorem 11. *Sei $\alpha : A^* \rightarrow M$ ein Monoidhomomorphismus in ein endliches Monoid, dann existiert für alle $w \in A^*$ ein α -factorization Forest der Höhe $h \leq 3|M| - 1$.*

Beweis. Siehe [11] [4] \square

8 Korrektheit des Algorithmus

In diesem Abschnitt soll die Korrektheit des Algorithmus, d.h. $\downarrow \mathfrak{J}_B^* \subseteq \mathfrak{C}[\alpha, B]$ für alle $B \subseteq A$ gezeigt werden.

Proposition 12. *Sei $B \subseteq A$ dann gilt $\mathfrak{J}_B^* \subseteq \mathfrak{C}^k[\alpha, B]$ für alle $k \in \mathbb{N}$*

Da nach Definition $\mathfrak{C}[\alpha, B] = \bigcap_k \mathfrak{C}^k[\alpha, B]$ gilt, folgt $\mathfrak{J}_B^* \subseteq \mathfrak{C}[\alpha, B]$ aus Proposition 12 und zusammen mit $\downarrow \mathfrak{C}[\alpha, B] = \mathfrak{C}[\alpha, B]$ folgt dann $\downarrow \mathfrak{J}_B^* \subseteq \mathfrak{C}[\alpha, B]$.

Sei also nun $k \in \mathbb{N}$, $B \subseteq A$ und $R \in \mathfrak{J}_B^*$ beliebig aber fest, dann ist $R \in \mathfrak{C}^k[\alpha, B]$ zu zeigen. Nach der Definition von \mathfrak{J}_B^* existiert ein $j \in \mathbb{N}$ so dass $R \in \mathfrak{J}_B^j$ gilt und somit können wir vollständige Induktion nach j anwenden.

Sei $R \in \mathfrak{J}_B^0$ dann gilt offensichtlich $R \in \mathfrak{C}^k[\alpha, B]$. Sei nun die Aussage $\mathfrak{J}_C^j \subseteq \mathfrak{C}^k[\alpha, C]$ für $j \geq 0$ und alle Alphabete $C \subseteq A$ bereits bewiesen. Sei nun $R \in \mathfrak{J}_B^{j+1} = \mathfrak{J}_B^j \cup M_B^j \cup D_B^j$. Falls $R \in \mathfrak{J}_B^j$ dann gilt nach Induktionsannahme $R \in \mathfrak{C}^k[\alpha, B]$.

Sei $R \in M_B^j$ dann gilt nach Gleichung (9), dass zwei Alphabete $C, D \subseteq A$ existieren mit $C \cup D = B$, sowie kompatible Mengen $T_C \in \mathfrak{J}_C^j$ und $T_D \in \mathfrak{J}_D^j$, so dass $R = T_C \cdot T_D$ gilt. Nach Induktionsannahme gilt dann $T_C \in \mathfrak{C}^k[\alpha, C]$ und $T_D \in \mathfrak{C}^k[\alpha, D]$ und damit $R = T_C \cdot T_D \in \mathfrak{C}^k[\alpha, C] \cdot \mathfrak{C}^k[\alpha, D] \subseteq \mathfrak{C}^k[\alpha, B]$ nach Proposition 4.

Sei $R \in D_B^j$, dann existiert nach Definition von D_B^j in Gleichung (10) ein $T \in \mathfrak{J}_B^j$ mit $R = T^\omega \cdot \{(1, \alpha(w)) \mid \mathbf{alph}(w) = B\} \cdot T^\omega$, wobei ω der Exponent der Halbgruppe 2^{M^2} ist. Wir wählen $h = \omega \cdot 2^{2k}$, dann gilt $T^h = T^\omega$ nach Definition von ω und damit $R = T^h \cdot \{(1, \alpha(w)) \mid \mathbf{alph}(w) = B\} \cdot T^h$. Nach Induktionsannahme gilt $T \in \mathfrak{C}^k[\alpha, B]$ und damit haben alle $\Sigma_2[k]$ -Ketten in T dasselbe erste Element. Sei t_1 dieses Element. Nach Definition von $\mathfrak{C}^k[\alpha, B]$ gilt, dass ein $u \in A^*$ existiert, so dass für alle $(t_1, t_2) \in \mathfrak{C}^k[\alpha, B]$ ein $u_2 \in A^*$ existiert mit $u \leq_2^k u_2$ und $\alpha(u) = t_1$ und $\alpha(u_2) = t_2$ und $\mathbf{alph}(u) = \mathbf{alph}(u_2) = B$.

Wir zeigen nun $R \in \mathfrak{C}^k[\alpha, B]$. Sei $r \in R$, dann gilt $r = (r_1, r_2) = (t_1^h t_1', t_2^h s_2'')$, wobei $s_2 = \alpha(w)$ für ein w mit $\mathbf{alph}(w) = B$ und $(t_1^h, t_2'), (t_1', t_2'') \in T^h = T^\omega$.

Sei $w = u^{2h}$ dann gilt nun $\alpha(w) = r_1$. Aus $T \in \mathfrak{C}^k[\alpha, B]$ folgt $T^h \in \mathfrak{C}^k[\alpha, B]$ nach Proposition 4 und damit existieren Wörter $w_2', w_2'', v \in A^*$ mit $\alpha(w_2') = t_2'$ und $\alpha(w_2'') = t_2''$ und $\alpha(v) = s_2$, sowie $\mathbf{alph}(w_2') = \mathbf{alph}(w_2'') = \mathbf{alph}(v) = B$ und $u^h \leq_2^k w_2'$ und $u^h \leq_2^k w_2''$ und damit gilt $\alpha(w_2' v w_2'') = r_2$. Nun folgt Proposition 12 nach Lemma 13.

Lemma 13. $u^h u^h \leq_2^k w_2' v w_2''$

Beweis. Nach Lemma 8 gilt bereits $u^h v u^h \leq_2^k w_2' v w_2''$ und somit ist nur noch $u^h u^h \leq_2^k u^h v u^h$ zu zeigen. Da $\mathbf{alph}(v) = \mathbf{alph}(u) = B$ gilt

$$v \leq_1^k u^{2k} \tag{11}$$

und zusammen mit $h = \omega \cdot 2^{2k}$ und Lemma 10 folgt dann $u^h u^h \leq_2^k u^h v u^h$. \square

9 Vollständigkeit des Algorithmus

Für die Vollständigkeit müssen wir nun für alle $B \subseteq A$ noch $\mathfrak{C}^l[\alpha, B] \subseteq \downarrow \mathfrak{J}_B^*$ für $l \geq 3 \cdot |M| \cdot 2^{|A|} \cdot 2^{2^{2|M|^2}}$ zeigen. Dazu werden kompatible Mengen von $\Sigma_2[k]$ -Ketten definiert die von einem Wort $w \in A^*$ erzeugt werden.

$$G^k(w) = \left\{ (\alpha(w), t_2) \in M^2 \mid \exists w_2 \in A^* (\alpha(w_2) = t_2 \wedge w \leq_2^k w_2) \right\} \quad (12)$$

Aus dieser Definition folgt direkt $G^k(w) \in \mathfrak{C}^k[\alpha, \mathbf{alph}(w)]$. Andererseits folgt für alle $B \subseteq A$ und jede kompatible Menge $T \in \mathfrak{C}^k[\alpha, B]$, dass ein Wort $w \in A^*$ existiert mit $\mathbf{alph}(w) = B$ und $T \subseteq G^k(w)$.

Lemma 14. *Sei $w_1 \cdots w_{m+1} \in A^*$ und $k \in \mathbb{N}$ mit $k > m$ dann gilt:*

$$G^k(w_1 \cdots w_{m+1}) \subseteq G^{k-m}(w_1) \cdots G^{k-m}(w_{m+1})$$

Beweis. Sei $(s_1, s_2) \in G^k(w_1 \cdots w_{m+1})$. Nach Definition existieren $u_1, u_2 \in A^*$ mit $u_1 = w_1 \cdots w_{m+1}$ und $\alpha(u_1) = s_1$ und $\alpha(u_2) = s_2$ und $u_1 \leq_2^k u_2$.

Ein einfaches Ehrenfeucht-Fraisse Argument liefert das u_1 und u_2 in $u_1 = u_{1,1} \cdots u_{1,m+1}$ und $u_2 = u_{2,1} \cdots u_{2,m+1}$ mit $u_{1,1} = w_1, \dots, u_{1,m+1} = w_{m+1}$ und $u_{1,j} \leq_2^k u_{2,j}$ für alle j zerlegt werden können. Für alle i, j sei $\alpha(u_{i,j}) = s_{i,j}$ dann gilt $(s_{1,j}, s_{2,j}) \in G^{k-m}(w_j)$ für alle j und des Weiteren $(s_1, s_2) = (s_{1,1}, s_{2,1}) \cdots (s_{1,m+1}, s_{2,m+1})$ und damit $(s_1, s_2) \in G^{k-m}(w_1) \cdots G^{k-m}(w_{m+1})$ \square

Wir definieren einen Monoidhomomorphismus $\beta : A^* \rightarrow M \times 2^A$ durch $\beta(w) = (\alpha(w), \mathbf{alph}(w))$. Wobei $M \times 2^A$ mit der komponentenweisen Verknüpfung $(m_1, B_1)(m_2, B_2) = (m_1 m_2, B_1 \cup B_2)$ versehen sei. Des Weiteren definieren wir $\kappa(h) = h \cdot 2^{2^{2|M|^2}}$.

Proposition 15. *Sei $B \subseteq A$ und $w \in A^*$ mit $\mathbf{alph}(w) = B$ so, dass ein β -factorization Forest für w der Höhe h existiert. Dann gilt $G^k(w) \in \downarrow \mathfrak{J}_B^*$ für alle $k \geq \kappa(h)$.*

Wir werden Proposition 15 folgendermaßen verwenden. Sei $l \geq 3 \cdot |M| \cdot 2^{|A|} \cdot 2^{2^{2|M|^2}}$ und $T \in \mathfrak{C}^l[\alpha, B]$, dann existiert ein Wort $w \in A^*$ mit $\mathbf{alph}(w) = B$ und $T \subseteq G^l(w)$. Nach Theorem 11 hat w ein β -factorization Forest der Höhe $h \leq 3 \cdot |M| \cdot 2^{|A|}$ und nach Proposition 15 folgt $G^l(w) \in \downarrow \mathfrak{J}_B^*$ und damit $T \in \downarrow \mathfrak{J}_B^*$ und damit $\mathfrak{C}^l[\alpha, B] \subseteq \downarrow \mathfrak{J}_B^*$.

Wir zeigen nun Proposition 15. Sei also $w \in A^*$ mit einem β -factorization Forest der Höhe h sowie $\mathbf{alph}(w) = B$ und $k \geq h \cdot 2^{2^{2|M|^2}}$. Wir zeigen es existiert $T \in \mathfrak{J}_B^*$ mit $G^k(w) \subseteq T$ und damit $G^k(w) \in \downarrow \mathfrak{J}_B^*$. Der Beweis ist ein Induktionsbeweis über die Höhe h des β -factorization Forest. Dazu wird Lemma 14 verwendet um w entsprechend des β -factorization Forest für w zu faktorisieren, wobei die Induktionsannahme für die einzelnen Faktoren von w bereits gilt.

Proposition 16. $\downarrow \mathfrak{J}_B^*$ ist eine Unterhalbgruppe von 2^{M^2}

Beweis. Sind $S_1, S_2 \in \mathfrak{J}_B^*$ dann existieren $j_1, j_2 \geq 0$, so dass $S_1 \in \mathfrak{J}_B^{j_1}$ und $S_2 \in \mathfrak{J}_B^{j_2}$. Sei o.B.d.A. $j_1 \geq j_2$ dann gilt also $S_1, S_2 \in \mathfrak{J}_B^{j_1}$ und damit $S_1 \cdot S_2 \in \mathfrak{J}_B^{j_1+1}$ nach Gleichung (9) und damit $S_1 \cdot S_2 \in \mathfrak{J}_B^*$ nach Definition von \mathfrak{J}_B^* . Aus der Abgeschlossenheit von \mathfrak{J}_B^* bezüglich der Multiplikation folgt nach Definition von \downarrow nun auch, dass $\downarrow \mathfrak{J}_B^*$ bezüglich der Multiplikation abgeschlossen ist. \square

Fall 1: Sei der Wurzelknoten des β -factorization Forest zugleich ein Blattknoten. Dann gilt $w = a$ für ein $a \in A$ und $h = 1$. Damit existiert eine Formel $\phi = \exists x(P_a(x) \wedge \forall y y = x)$ mit $L(\phi) = \{a\}$ und damit $\forall w \in A^*(a \leq_2^k w \implies w = a)$. Damit folgt aber, dass $G^k(w) = \{(\alpha(a), \alpha(a))\} \in \mathfrak{J}_B^0 \subseteq \downarrow \mathfrak{J}_B^*$ für alle $k \geq 2$ und damit auch für alle $k \geq \kappa(h) > 2$ ist.

Fall 2 : Sei nun die Wurzel ein binärer Knoten. Dann lässt sich w in $w = w_1 w_2$ zerlegen mit $h(w_1) \leq h-1$ und $h(w_2) \leq h-1$ und $\text{alph}(w_1) = C$ und $\text{alph}(w_2) = D$ und $C \cup D = B$ für zwei Alphabete $C, D \subseteq A$. Aus $\kappa(h) - 1 \geq \kappa(h-1)$ folgt nach Induktionsannahme $G^{\kappa(h)-1}(w_1) \in \downarrow \mathfrak{J}_C^*$ und $G^{\kappa(h)-1}(w_2) \in \downarrow \mathfrak{J}_D^*$. Damit existieren $T_1 \in \mathfrak{J}_C^*$ und $T_2 \in \mathfrak{J}_D^*$ mit $G^{\kappa(h)-1}(w_1) \subseteq T_1$ und $G^{\kappa(h)-1}(w_2) \subseteq T_2$. Nach Gleichung (9) gilt $T_1 \cdot T_2 \in \mathfrak{J}_B^*$ und damit $G^{\kappa(h)-1}(w_1) \cdot G^{\kappa(h)-1}(w_2) \in \downarrow \mathfrak{J}_B^*$. Schließlich gilt nach Lemma 14 $G^{\kappa(h)}(w) \subseteq G^{\kappa(h)-1}(w_1) \cdot G^{\kappa(h)-1}(w_2)$ und damit $G^{\kappa(h)}(w) \in \downarrow \mathfrak{J}_B^*$.

Fall 3: Sei die Wurzel nun ein idempotenter Knoten. Dann lässt sich w in w_1, \dots, w_m zerlegen mit $\alpha(w_i) = e$ für ein idempotentes $e \in M$ und ein $B \in A$ mit $\text{alph}(w_i) = B$.

Sei $e \in M$ idempotent und $B \subseteq A$ beliebig. Wir sagen $u \in A^*$ besitzt eine (e, B) -Zerlegung u_1, \dots, u_m falls folgendes gilt

- $u = u_1 \cdots u_m$
- für alle $1 \leq i \leq m$ gilt $\alpha(u_i) = e$ und $\text{alph}(u_i) = B$
- für alle $1 \leq i \leq m$ gilt $G^{\kappa(h-1)}(u_i) \in \downarrow \mathfrak{J}_B^*$

Dies bedeutet das $\beta(u_j) = (\alpha(u_j), \text{alph}(u_j))$ für alle j ein konstantes idempotentes Element aus $M \times 2^A$ ist und Proposition 15 für alle Faktoren u_j bereits erfüllt ist.

Lemma 17. *Besitze u eine (e, B) -Zerlegung u_1, \dots, u_m . Dann gilt $G^k(u_j \cdots u_{j'}) \subseteq \{(e, \alpha(w)) \mid w \in A^* \wedge \text{alph}(w) = B\}$ für alle $k \geq 1$ und $1 \leq j \leq j' \leq m$*

Beweis. Offensichtlich gilt nach der Definition einer (e, B) -Zerlegung $\alpha(u_l) = e$ für alle $j \leq l \leq j'$ und da e idempotent und α ein Homomorphismus ist folgt $\alpha(u_j \cdots u_{j'}) = e$ und damit nach der Definition in (12), dass $G^k(u_j \cdots u_{j'}) = \{(e, t_2) \in M^2 \mid \exists w_2 \in A^*(\alpha(w_2) = t_2 \wedge u_j \cdots u_{j'} \leq_2^k w_2) \in M\}$ und aus $u_j \cdots u_{j'} \leq_2^k w_2$ folgt $\text{alph}(w_2) = \text{alph}(u_j \cdots u_{j'}) = B$ für $k \geq 1$. \square

Wir wollen $G^{\kappa(h)}(w) \in \downarrow \mathfrak{J}_B^*$ zeigen. Da die Anzahl der Faktoren m in der (e, B) -Zerlegung von w beliebig groß ist, ist $\kappa(h) - (m-1) < \kappa(h-1)$ möglich und wir können also nicht mehr auf $G^{\kappa(h)}(w) \subseteq G^{\kappa(h-1)}(w_1) \cdots G^{\kappa(h-1)}(w_m)$ schließen.

Deswegen wird w_1, \dots, w_m in eine beschränkte Anzahl Teilerlegungen von w partitioniert. Zum Zweck der Partitionierung werden nachfolgend ν -Sequenz und der Index einer Zerlegung definiert.

Sei $\nu = 2^{|M|^2}$. Sei $u \in A^*$ derart, dass eine (e, B) -Zerlegung u_1, \dots, u_m existiert. Wir definieren die sogenannte ν -**Sequenz** an einer Stelle $1 \leq j \leq m - \nu$ durch die Sequenz $G^{\kappa(h-1)}(u_j), \dots, G^{\kappa(h-1)}(u_j + \nu) \in \downarrow \mathfrak{J}_B^*$. Die Anzahl der unterschiedlichen ν -Sequenzen in u_1, \dots, u_m nennen wir **Index**. Beachte, dass höchstens $\nu^{\nu+1}$ viele paarweise voneinander verschiedene ν -Sequenzen existieren und der Index dadurch durch $\nu^{\nu+1}$ beschränkt ist. Der einfacheren Schreibweise wegen definieren wir nun $\tilde{k} = \kappa(h - 1)$.

Lemma 18. *Sei $u \in A^*$ und besitze u eine (e, B) -Zerlegung u_1, \dots, u_m mit Index g und sei $\hat{k} \geq 2g + 2(\nu + 1) + \tilde{k}$ dann gilt $G^{\hat{k}}(u) \in \downarrow \mathfrak{J}_B^*$.*

Wir werden Lemma 18 für den Beweis in Fall 3 folgendermaßen verwenden. Wir wissen, dass w eine (e, B) -Zerlegung w_1, \dots, w_m besitzt. Da $g \leq \nu^{\nu+1}$ zeigen wir $\kappa(h) \geq 2\nu^{\nu+1} + 2(\nu + 1) + \tilde{k}$, denn aus Lemma 18 folgt dann $G^{\kappa(h)}(w) \in \downarrow \mathfrak{J}_B^*$. Es gilt $2^{2^{|M|^2}} = 2^{\nu^2} \geq 2\nu^{\nu+1} + 2(\nu + 1)$ für $\nu > 2$ und daraus folgt $\kappa(h) = h \cdot 2^{2^{|M|^2}} = 2^{2^{|M|^2}} + (h - 1) \cdot 2^{2^{|M|^2}} \geq 2\nu^{\nu+1} + 2(\nu + 1) + \tilde{k}$

Beweis von Lemma 18. Der Beweis benutzt eine Induktion über den Index g . Es gibt zwei Fälle, je nachdem ob eine ν -Sequenz existiert die doppelt vorkommt oder nicht. Seien also alle ν -Sequenzen zu den Positionen $1 \leq j \leq m - \nu$ paarweise voneinander verschieden, dann gilt $m = g + \nu$. Des Weiteren gilt nach Lemma 14

$$G^{\hat{k}}(u) \subseteq G^{\hat{k}-(m-1)}(u_1) \dots G^{\hat{k}-(m-1)}(u_m)$$

Aus $\hat{k} - (m - 1) \geq \tilde{k}$ folgt $G^{\hat{k}-(m-1)}(u_j) \in \downarrow \mathfrak{J}_B^*$ für $1 \leq j \leq m$ und damit folgt dann mit Proposition 16 $G^{\hat{k}-(m-1)}(u_1) \dots G^{\hat{k}-(m-1)}(u_m) \in \downarrow \mathfrak{J}_B^*$ und damit $G^{\hat{k}}(w) \in \downarrow \mathfrak{J}_B^*$. Komme nun also mindestens eine ν -Sequenz doppelt vor. Dann existieren zwei Indizes $1 \leq j < j' \leq m - \nu$ derart, dass die ν -Sequenzen an den Stellen j und j' identisch sind. Sei nun $R_1, \dots, R_{\nu+1}$ diese Sequenz. Wir können j und j' derart wählen, dass j minimal ist und j' maximal, d.h. das keine Stelle j'' mit $j'' < j$ oder $j' < j''$ existiert an der $R_1, \dots, R_{\nu+1}$ auftritt. Wir zerlegen nun u folgendermaßen

$$\begin{aligned} v_1 &= u_1 \dots u_{j-1} \\ v_2 &= u_j \dots u_{j'+\nu} \\ v_3 &= u_{j'+\nu+1} \dots u_m \end{aligned}$$

Nun gilt nach Lemma 14

$$G^{\hat{k}}(w) \subseteq G^{\hat{k}-2}(v_1)G^{\hat{k}-2}(v_2)G^{\hat{k}-2}(v_3)$$

Wir müssen also noch $G^{\hat{k}-2}(v_1), G^{\hat{k}-2}(v_2), G^{\hat{k}-2}(v_3) \in \downarrow \mathfrak{J}_B^*$ zeigen. Dann gilt nach Proposition 16 $G^{\hat{k}}(w) \in \downarrow \mathfrak{J}_B^*$. Nach Definition von j und j' sind u_1, \dots, u_{j-1} und $u_{j'+\nu+1}, \dots, u_m$

(e, B) -Zerlegungen mit einem kleineren Index als g . Daher gilt nach Induktionsannahme $G^{\hat{k}-2}(v_1), G^{\hat{k}-2}(v_3) \in \downarrow \mathfrak{J}_B^*$.

Wir betrachten nun $G^{\hat{k}-2}(v_2)$. Es gibt wieder 2 Fälle. Sei $j' \leq j + \nu$ dann gilt, dass $u_j, \dots, u_{j'+\nu}$ eine (e, B) -Zerlegung von v_2 der Länge $d < 2(\nu + 1)$ ist. Da $\hat{k} - 2 - (d - 1) \geq \tilde{k}$ und nach Lemma 14 $G^{\hat{k}-2}(v_2) \subseteq G^{\hat{k}-1-d}(u_j) \cdots G^{\hat{k}-1-d}(u_{j'+\nu})$ gilt kann analog zu obigen Fall wieder $G^{\hat{k}-2}(v_2) \in \downarrow \mathfrak{J}_B^*$ gefolgert werden. Sei nun also $j' > j + \nu$. Beachte, dass $g \geq 1$ ist. Damit folgt nach Definition von \hat{k} und \tilde{k} nun $\hat{k} - 2 - 2(\nu + 1) \geq \tilde{k}$ und damit

$$G^{\hat{k}-2}(v_2) \subseteq G^{\tilde{k}}(u_j) \cdots G^{\tilde{k}}(u_{j+\nu}) G^{\tilde{k}}(v) G^{\tilde{k}}(u_{j'}) \cdots G^{\tilde{k}}(u_{j'+\nu})$$

mit $v = u_{j+\nu+1} \cdots u_{j'-1}$ bzw. nach Definition von $R_1, \dots, R_{\nu+1}$

$$G^{\hat{k}-2}(v_2) \subseteq R_1 \cdots R_{\nu+1} G^{\tilde{k}}(v) R_1 \cdots R_{\nu+1} \quad (13)$$

Wir wollen nun Gleichung (10) anwenden. Da höchstens $\nu = 2^{|M|^2}$ viele Elemente des Monoid 2^{M^2} existieren, existieren Indizes $1 \leq j_1 < j_2 \leq \nu + 1$ mit $R_1 \cdots R_{j_1} = R_1 \cdots R_{j_2}$. Sei nun $S_1 = R_1 \cdots R_{j_1}, S_2 = R_{j_1+1} \cdots R_{j_2}$ und $S_3 = R_{j_2+1} \cdots R_{\nu+1}$ Nach Definition von S_1, S_2, S_3 gilt $R_1 \cdots R_{\nu+1} = S_1 \cdot (S_2)^\omega \cdot S_3$.

Durch Ersetzen von $R_1 \cdots R_{\nu+1}$ in Gleichung (13) erhalten wir

$$G^{\hat{k}-2}(v_2) \subseteq S_1 \cdot (S_2)^\omega \cdot S_3 \cdot G^{\tilde{k}}(v) \cdot S_1 \cdot (S_2)^\omega \cdot S_3$$

Da $u_1 \cdots u_m$ eine (e, B) -Zerlegung von u ist gelten $G^{\tilde{k}}(v) \subseteq \{(e, \alpha(w)) \mid \mathbf{alph}(w) = B\}$ und $G^{\tilde{k}}(u_i) \subseteq \{(e, \alpha(w)) \mid \mathbf{alph}(w) = B\}$ für $i \in \{j, \dots, j + \nu\} \cup \{j', \dots, j' + \nu\}$ nach Lemma 17. Damit ist auch $S_3 \cdot G^{\tilde{k}}(v) \cdot S_1 \subseteq \{(e, \alpha(w)) \mid \mathbf{alph}(w) = B\}$. Da die Σ_2 -Ketten in S_2 in der ersten Komponente e stehen haben gilt $(S_2)^\omega \cdot \{(e, \alpha(w)) \mid \mathbf{alph}(w) = B\} \cdot (S_2)^\omega = (S_2)^\omega \cdot \{(1, \alpha(w)) \mid \mathbf{alph}(w) = B\} \cdot (S_2)^\omega$ und damit

$$G^{\hat{k}-2}(v_2) \subseteq S_1 \cdot (S_2)^\omega \cdot \{(1, \alpha(w)) \mid \mathbf{alph}(w) = B\} \cdot (S_2)^\omega \cdot S_3$$

Es gilt $S_1, S_2, S_3 \in \downarrow \mathfrak{J}_B^*$ nach Proposition 16 und damit gilt nach Gleichung (10) $(S_2)^\omega \cdot \{(1, \alpha(w)) \mid \mathbf{alph}(w) = B\} \cdot (S_2)^\omega \in \mathfrak{J}_B^*$ und nach zweimaligen Anwenden von Gleichung (9) $S_1 \cdot (S_2)^\omega \cdot \{(1, \alpha(w)) \mid \mathbf{alph}(w) = B\} \cdot (S_2)^\omega \cdot S_3 \in \downarrow \mathfrak{J}_B^*$ und damit $G^{\hat{k}-2}(v_2) \in \downarrow \mathfrak{J}_B^*$. \square

10 Zusammenfassung

Zusätzlich zu [8] wurden in dieser Arbeit die Beweise für Proposition 1, Proposition 2 und Proposition 3 geführt. Des Weiteren wurde der Algorithmus vereinfacht. Thomas Place und Marc Zeitoun berechnen allgemeiner Σ_2 -Ketten der Länge n , da sie etwa Σ_2 -Ketten der Länge 3 für die Entscheidbarkeit der Mitgliedschaft einer Sprache zu $B\Sigma_2$ benötigen. Durch die Beschränkung auf Ketten der Länge 2 ergeben sich Verbesserungen in der Notation, des Weiteren ergeben sich Vereinfachungen da nicht allgemein die Berechnung von Ketten der

Länge $n + 1$ auf Ketten der Länge n für alle n zurückgeführt werden muss. Des Weiteren ergeben sich in dieser Arbeit Vereinfachungen in der Notation daraus, dass [8] nicht direkt mit Mengen von kompatiblen Mengen von Σ_2 -Ketten im Algorithmus arbeiten, sondern mit Funktionen die jedem Alphabet eine derartige Menge zuordnen. In [8] werden von Thomas Place und Marc Zeitoun analog zu unserer Definition der Σ_2 -Ketten allgemeiner Σ_i -Ketten für $i \in \mathbb{N}$ definiert. Dabei wird \leq_2^k in der Definition durch die Relation \leq_i^k ersetzt. Die Autoren erwähnen, dass ihr Algorithmus nur für $i = 2$ funktioniert. Der Grund ist, dass Gleichung (11) deswegen gilt, da Spieler I im zum \leq_1^k zugehörigen Ehrenfeucht-Fraisse-Spiel, das aktive Wort nicht wechseln kann, d.h. Gleichung (11) ist also i.A. für \leq_j^k mit $j > 1$ nicht mehr erfüllt und somit ist die Korrektheit des Algorithmus nicht mehr gewährleistet. Dies ist intuitiv klar, da jede Σ_i -Kette für $i > 2$ auch eine Σ_2 -Kette sein sollte aber nicht notwendigerweise umgekehrt. Umgekehrt sollte also die Berechnung aller Σ_1 -Ketten durch den Algorithmus an der Vollständigkeit scheitern. Dies passiert genau beim Induktionsanfang in Fall 1 bei der Induktion über die Höhe des β -factorization Forest, da für einen beliebigen Buchstaben $a \in A$ keine Formel $\phi \in \Sigma_1$ mit $L(\phi) = \{a\}$ mehr existiert und damit nicht mehr auf $G^k(a) \in \downarrow \mathfrak{J}_B^*$ für $k \geq \kappa(h)$ geschlossen werden kann. Alles Übrige in der Arbeit entspricht mehr oder weniger exakt Artikel [8].

Literaturverzeichnis

- [1] Mustapha Arfi. Polynomial operations on rational languages. In FranzJ. Brandenburg, Guy Vidal-Naquet, and Martin Wirsing, editors, *STACS 87*, volume 247 of *Lecture Notes in Computer Science*, pages 198–206. Springer Berlin Heidelberg, 1987.
- [2] Volker Diekert, Manfred Kufleitner, and Gerhard Rosenberger. *Diskrete algebraische Methoden: Arithmetik, Kryptographie, Automaten und Gruppen*. de Gruyter, Berlin [u.a.], 2013.
- [3] Martin Huschenbett and Manfred Kufleitner. Ehrenfeucht-Fraissé Games on Omega-Terms. In Ernst W. Mayr and Natacha Portier, editors, *31st International Symposium on Theoretical Aspects of Computer Science (STACS 2014)*, volume 25 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 374–385, Dagstuhl, Germany, 2014. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [4] Manfred Kufleitner. The height of factorization forests. *Mathematical Foundations of Computer Science 2008*, pages 443–454, 2008.
- [5] Robert McNaughton and Seymour A. Papert. *Counter-Free Automata (M.I.T. Research Monograph No. 65)*. The MIT Press, 1971.
- [6] Dominique Perrin and Jean-Eric Pin. *Infinite words : automata, semigroups, logic and games*. Pure and applied mathematics. Academic, London, San Diego (Calif.), 2004.
- [7] Thomas Place, Lorijn van Rooijen, and Marc Zeitoun. Separating regular languages by piecewise testable and unambiguous languages. In Krishnendu Chatterjee and Jiri Sgall, editors, *Mathematical Foundations of Computer Science 2013*, volume 8087 of *Lecture Notes in Computer Science*, pages 729–740. Springer Berlin Heidelberg, 2013.
- [8] Thomas Place and Marc Zeitoun. Going higher in the first-order quantifier alternation hierarchy on words. In Javier Esparza, Pierre Fraigniaud, Thore Husfeldt, and Elias Koutsoupias, editors, *Automata, Languages, and Programming*, volume 8573 of *Lecture Notes in Computer Science*, pages 342–353. Springer Berlin Heidelberg, 2014.
- [9] M.P. Schützenberger. On finite monoids having only trivial subgroups. *Information and Control*, 8(2):190–194, Apr 1965.
- [10] Imre Simon. Piecewise testable events. In H. Brakhage, editor, *Automata Theory and Formal Languages 2nd GI Conference Kaiserslautern, May 20–23, 1975*, volume 33 of *Lecture Notes in Computer Science*, pages 214–222. Springer Berlin Heidelberg, 1975.
- [11] Imre Simon. Factorization forests of finite height. *Theoretical Computer Science*, 72(1):65–94, 1990.

- [12] Howard Straubing. Finite automata, formal logic, and circuit complexity. 1994.

Erklärung

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

Ort, Datum, Unterschrift