

Otto-von-Guericke-Universität Magdeburg
Fakultät für Informatik
Institut für Wissens- und Sprachverarbeitung



Zur Endlichkeit von Bildsprachen synchroner deterministischer Ketten-Code-Bild-Systeme

Diplomarbeit
zur Erlangung des akademischen Grades
Diplom-Informatiker

eingereicht von

Bianca Truthe

geboren am 10. Januar 1975
in Magdeburg

Betreuer: Prof. Dr. Jürgen Dassow

Magdeburg, den 15. April 1999

Vorwort

Die vorliegende Arbeit stellt den Abschluß meines bisherigen Studiums dar. Daher möchte ich an dieser Stelle all jenen danken, die mich in den vergangenen Jahren unterstützt haben.

Besonderer Dank gebührt Prof. Hollatz für die vielen anregenden Diskussionen und unzähligen wertvollen Hinweise.

Desweiteren danke ich Prof. Dassow für die unkomplizierte Betreuung während der Diplom-Phase.

Der Studienstiftung des deutschen Volkes danke ich für die finanzielle Unterstützung und ideelle Förderung.

Ein herzlicher Dank gilt auch meinen Eltern.

Inhaltsverzeichnis

Abbildungsverzeichnis	vi
Tabellenverzeichnis	vi
Notation	vi
1 Einleitung	1
2 Grundlagen	5
2.1 Strukturen über einem Alphabet	5
2.2 Graphen	11
2.3 Ketten-Code-Bild-Systeme	21
3 Endlichkeitsuntersuchungen	26
3.1 <i>sDOL</i> -Systeme mit $k < 1$	27
3.2 <i>sDOL</i> -Systeme mit $k > 1$	30
3.3 <i>sDOL</i> -Systeme mit $k = 1$	31
4 Zusammenfassung und Ausblick	47
Literatur	49
Erklärung	51
Thesen	53

Abbildungsverzeichnis

1	Ketten-Code-Anwendungen	1
2	Ableitung eines Bildes	3
3	Verfeinerung eines Bildes	3
4	Abstrahierungshierarchie	20
5	Ableitung eines Graphen	22

Tabellenverzeichnis

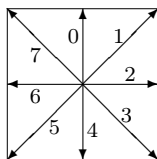
1	Beispiele für Ketten-Code-Bild-Systeme	2
2	Zusammenhang	20
3	Ableitung – Beispiel 1	33
4	Ableitung – Beispiel 2	34
5	Zusammenfassung	47

Notation

r, u, l, d	Symbole für Zeichenbewegungen (<i>right, up left, down</i>)
λ	Leerwort (Wort der Länge 0)
\mathcal{A}	Alphabet $\{r, u, l, d\}$
\mathcal{A}^*	Menge aller Wörter über dem Alphabet \mathcal{A} einschließlich λ
\mathcal{A}^+	Menge aller Wörter über dem Alphabet \mathcal{A} ohne λ
\mathcal{A}_n^*	Menge aller Wörter aus \mathcal{A}^* mit der Länge n
u, v, w, x, y, \dots	Wörter aus \mathcal{A}^*
x, y, \dots	Variable für Buchstaben aus \mathcal{A} , evtl. mit Indices
$\#w$	Länge eines Wortes $w \in \mathcal{A}^*$
$\#_x w$	Anzahl der Vorkommen eines Buchstaben x im Wort w
$[w]$	Buchstaben-Menge eines Wortes w
\mathbb{Z}^2	Menge aller geordneten Paare ganzer Zahlen
\mathbf{o}	Nullpunkt des \mathbb{Z}^2 : $(0, 0)$
$\mathbf{a}, \mathbf{p}, \mathbf{q}, \mathbf{r}, \mathbf{v}$	Punkte des \mathbb{Z}^2 , evtl. mit Indices
$g^{\mathbf{a}}(w)$	gerichteter Graph zu einem Wort w und einem Bezugspunkt \mathbf{a}
$sg^{\mathbf{a}}(w)$	schlichter gerichteter Graph zu $g^{\mathbf{a}}(w)$
$\ {}^{\mathbf{a}}w$	Kantenmenge des schlichten gerichteten Graphen $sg^{\mathbf{a}}(w)$
$p^{\mathbf{a}}(w)$	Bild zu einem Wort w und einem Bezugspunkt \mathbf{a}
\mathbb{N}_0	Menge der natürlichen Zahlen einschließlich 0
\mathbb{N}	Menge der natürlichen Zahlen ohne 0
$w', w'', w''', w^{(n)}$	1., 2., 3., n -te Ableitung eines Wortes w

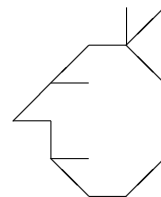
1 Einleitung

Zentrale Aufgaben der Bildverarbeitung liegen beim Beschreiben, Erzeugen, Speichern und Erkennen von Bildern. Mit Ketten-Codes lieferte FREEMAN in den 60er Jahren eine Beschreibungsmöglichkeit für Strichgraphiken [Fre61, Fre74]. Dabei entsteht ein Bild durch eine Folge von Zeichenbewegungen, die durch Symbole, z. B. Buchstaben repräsentiert sind. Ein Wort beschreibt ein Bild, das durch Nacheinanderausführung der Zeichenschritte seiner Buchstaben entsteht. FREEMAN benutzt ein achtelementiges Alphabet $\{0, \dots, 7\}$, dessen Elemente entsprechend folgender Skizze interpretiert werden:



Zum Beispiel entsteht das nebenstehende Bild aus dem Wort 1261204153445672606:

(Beim Nachvollziehen beginne man an der Nasenspitze.)



Dieser Zusammenhang von Wörtern und Bildern legt es nahe, Beziehungen zwischen formalen Sprachen und Bildmengen zu suchen. Für sprachentheoretische Betrachtungen genügen die vier Richtungen $\{0, 2, 4, 6\}$, da die restlichen vier keine andersartigen Resultate liefern [DH89]. In Anlehnung an Plotter-Befehle schreibt man u, r, d, l für die Richtungen *up*, *right*, *down*, *left*. Auch andere Befehlssätze, z. B. $\{F, +, -\}$ mit den Bedeutungen

- F : zeichne eine Strecke in Blickrichtung,
- $+$: drehe Blickrichtung um einen gewissen Winkel nach links,
- $-$: drehe Blickrichtung um einen gewissen Winkel nach rechts

können als Ketten-Codes verstanden werden [PL90]. Eine zusätzlich eingeführte Klammerung ermöglicht es, Verzweigungen zu modellieren (s. Abbildung 1, rechtes Bild). Desweiteren kann ein Alphabet Buchstaben ohne graphische Bedeutung enthalten; sie dienen zum Ableiten, werden aber beim Zeichnen übergangen (s. Tabelle 1, Beispiel 3). Mittels Ketten-Codes lassen sich Muster – wie Fraktale, Kurven, Folkloremuster, Pflanzen – beschreiben:

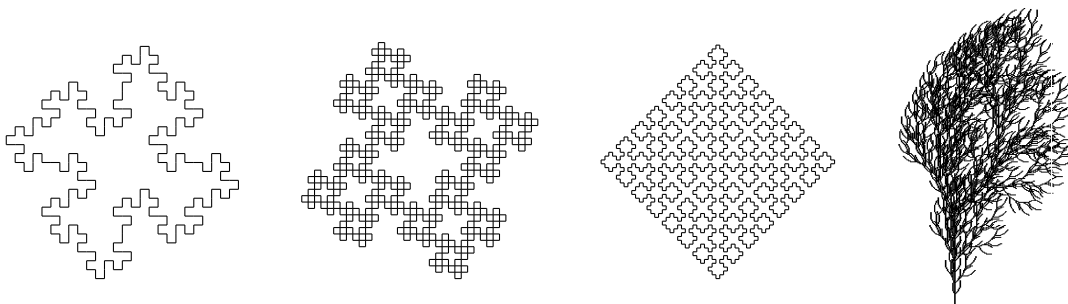


Abbildung 1: Ketten-Code-Anwendungen

Zur Erzeugung von Wortmengen stehen im wesentlichen zwei formale Mechanis-


men zur Verfügung: CHOMSKY-Grammatiken [Cho56] und LINDENMAYER-Systeme [Lin68]. Grammatiken eignen sich durch ihr sequentielles Ableitungskonzept zum Modellieren von natürlichen Sprachen und ermöglichen die Entwicklung von Programmiersprachen. LINDENMAYER-Systemen (L -Systemen) liegt ein Konzept des parallelen Ableitens zugrunde. Dadurch lassen sich mit ihnen Entwicklungsvorgänge bei Pflanzen beschreiben und simulieren [PL90]. Ketten-Code-Bild-Grammatiken bzw. Ketten-Code-Bild-Systeme sind Grammatiken bzw. L -Systeme über Ketten-Codes; in diesem Zusammenhang interessieren die generierten/generierbaren Bildsprachen. Beispiele für Ketten-Code-Bild-Systeme sind

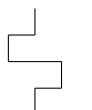
1. $S_1 = (\{r, u, l, d\}, P, \omega)$ mit dem Startwort $\omega = ruld$ und der Regelmenge $P = \{r \rightarrow rdruurdr, u \rightarrow urulluru, l \rightarrow lulddlul, d \rightarrow dldrrdld\}$. Das Bild der 2. Ableitung von ω ist in Abbildung 1 – links – zu sehen.
2. $S_2 = (\{r, u, l, d\}, P, \omega)$ mit dem Startwort $\omega = rdlu$ und der Regelmenge $P = \{d \rightarrow lulur, l \rightarrow dldlu, r \rightarrow ururd, u \rightarrow rdrdl\}$. Abbildung 1 zeigt das Bild der 4. Ableitung von ω (2. von links).
3. $S_3 = (\{X, F, +, -\}, P, \omega, \delta)$ mit dem Startwort $\omega = F+XF+F+XF$, dem Winkel $\delta = 90^\circ$ und der Regelmenge $P = \{X \rightarrow XF-F+F-XF+F+XF-F+F-X, F \rightarrow F, + \rightarrow +, - \rightarrow -\}$. Das Bild der 4. Ableitung von ω ist in der 1. Abbildung (2. von rechts) zu sehen.
4. $S_4 = (\{F, +, -, [,]\}, P, F, \delta)$ mit dem Winkel $\delta = 22.5^\circ$ und der Regelmenge $P = \{F \rightarrow FF+[+F-F-F]-[-F+F+F], + \rightarrow +, - \rightarrow -, [\rightarrow [,] \rightarrow]\}$. Abbildung 1 zeigt rechts das Bild der 4. Ableitung vom Startwort F .

Tabelle 1: Beispiele für Ketten-Code-Bild-Systeme

Diese und weitere Beispiele sind in [PL90, DH89] zu finden.

Mittels Grammatiken und L -Systemen lassen sich aus einem Wort nach gewissen Regeln andere Wörter ableiten. Einer Ersetzung von Buchstaben durch Wörter entspricht bei Ketten-Code-Bild-Grammatiken und Ketten-Code-Bild-Systemen eine Ersetzung von Bildkanten durch Bilder. Von besonderem Interesse sind Bildmengen synchroner Ketten-Code-Bild-Systeme. Dabei sind jene Bilder, die eine Kante ersetzen, in gewisser Weise einander ähnlich. Dies sei am Beispielsystem S_1 aus Tabelle 1 demonstriert. Das Startwort (Axiom) ist $ruld$; das ihm zugeordnete Bild ist links in Abbildung 2 dargestellt. Die Bilder der ersetzenden Wörter sind

 für die Wörter $rdruurdr$ und $lulddlul$ sowie

 für die Wörter $urulluru$ und $dldrrdld$.

Das Bild der 1. Ableitung des Axioms setzt sich aus diesen ersetzenden Bildern zu-

sammen und ist in Abbildung 2 (rechts) zu sehen.

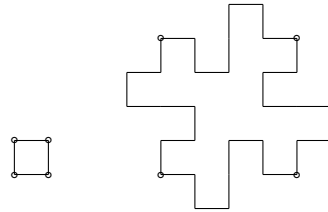


Abbildung 2: Ableitung eines Bildes

Die Kreise gehören nicht zu den Bildern; sie sollen korrespondierende Eckpunkte markieren. In diesem Beispiel entsteht bei jedem Ableitungsschritt ein flächenmäßig größeres Bild; die Länge einer Kante ist jedoch konstant. Man kann einen Ableitungsschritt auch so verstehen, daß der Abstand korrespondierender Eckpunkte gleich bleibt. Dies hat unter Umständen eine Verkürzung der Kantenlängen zur Folge. Damit kann die Ableitung eines Bildes als dessen Verfeinerung angesehen werden, was die ersten Bilder des obigen Beispielsystems in folgender Abbildung zeigen sollen:

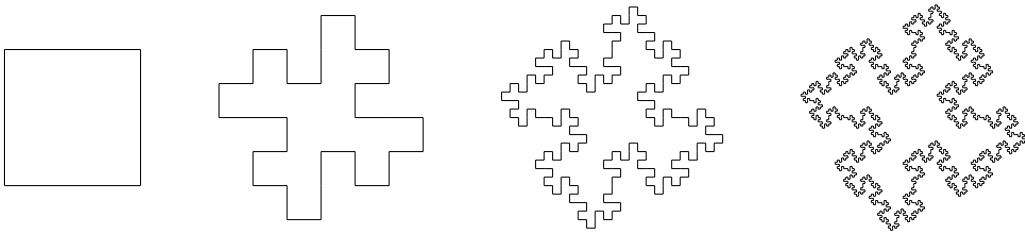


Abbildung 3: Verfeinerung eines Bildes

Solche Sichtweise findet sich bei hierarchischen Entwicklungen, z. B. von VLSI-Strukturen wieder [LW87].

Die Ergebnisse zu Problemen aus der Theorie der formalen Sprachen sind i. a. nicht von Wortmengen auf Bildmengen übertragbar. Ein Wort beschreibt ein Bild eindeutig, aber nicht notwendigerweise umkehrbar eindeutig: Unterschiedlichen Wörtern können übereinstimmende Bilder zugeordnet sein (z. B. repräsentieren die Wörter *rul*d, *ur*d*l*, *ud*r*ul*d*r* das Bild: \square). Insbesondere können unendliche Wortmengen auch endliche Bildmengen darstellen.

Probleme zu Bildsprachen formaler Grammatiken sind in zahlreichen Arbeiten untersucht worden, z. B. in

[M⁺82] Hierarchien, grundlegende Entscheidbarkeitsprobleme sowie eine Beschreibungskomplexität von Bildern,

[We183] Entscheidbarkeitsprobleme und deren Komplexität,

[SW85] das Mitgliedsproblem für reguläre Bildsprachen und Entscheidungsprobleme,

[Das88] Konvex- und Einfachheit von Ketten-Code-Bildsprachen,

[DH93] Unterbild- und Mitgliedsprobleme für Mengen nichtzusammenhängender Bilder.

Kontextfreie L -Systeme werden nach [RS80] in folgende Klassen eingeteilt: DOL (deterministisches Ersetzen von Buchstaben), OL (nichtdeterministisches Ersetzen), $DTOL$ (nichtdeterministische Auswahl einer Ersetzungstafel, deterministisches Ersetzen entsprechend der Tafel) und TOL (sowohl die Tafelwahl als auch das Ersetzen erfolgen nichtdeterministisch). Über die Möglichkeit einer bildlichen Interpretation von Wörtern bzw. Wortsprachen ist in [RS80] jedoch nichts ausgesagt.

Im Gegensatz zu Grammatiken gibt es für L -Systeme noch keine Theorie der Ketten-Code-Bildsprachen.

Die Arbeit [DHR92] zu synchronen Ketten-Code-Bild-Sprachen ordnet die Bildsprachen-Familien $\mathcal{B}(sDOL)$, $\mathcal{B}(sOL)$, $\mathcal{B}(sDTOL)$, $\mathcal{B}(sTOL)$ hinsichtlich ihrer Mächtigkeit in die Hierarchie der Familien $\mathcal{B}(REG)$ regulärer, $\mathcal{B}(CF)$ kontextfreier und $\mathcal{B}(CS)$ kontextabhängiger Bildsprachen ein. Außerdem liefert sie Lösungen zu einigen Entscheidungsproblemen. Das Endlichkeitsproblem für $sTOL$ -Systeme kann mit jener Arbeit nicht als gelöst betrachtet werden, da ein logischer Schluß im Beweis von Satz 4 in [DHR92] falsch ist. Genaueres dazu steht im Abschnitt 3 (S. 26) der vorliegenden Arbeit.

Für allgemeine synchrone Ketten-Code-Bild-Systeme ($sTOL$) stellt es sich als schwierig heraus, die Entscheidbarkeit oder Unentscheidbarkeit zu beweisen. Die Entscheidbarkeit der Endlichkeit ist jedoch grundlegend für weitere Entscheidungsprobleme.

Die vorliegende Arbeit entwickelt eine Hierarchie, in der die Interpretation eines Wortes als Bild einen mehrstufigen Abstrahierungsprozeß durchläuft. Auf unterster Ebene werden Wörter einer Sprache betrachtet. Ein Interpretieren der Wörter als gerichtete Graphen mit Mehrfachkanten führt auf die nächsthöhere Stufe. Auf die dritte Ebene gelangt man durch Abstrahieren von den Mehrfachkanten, d. h. jedem gerichteten Graphen der zweiten Ebene ist sein schlichter gerichteter Graph auf der dritten zugeordnet. Durch Abstrahieren von den Kantenrichtungen gelangt man schließlich auf die Ebene der Bilder. Da jeder Abstraktionsschritt eine eindeutige Abbildung ist, gibt es zu jedem Wort der untersten Schicht einen eindeutigen „Interpretationsweg“ zu einem Bild auf der obersten Stufe. Anhand dieses Modells wird die Entscheidbarkeit der Endlich-/Unendlichkeit von Bildsprachen für eine Untermenge der $sTOL$ -Systeme, die deterministischen, synchronen Ketten-Code-Bild-Systeme ($sDOL$) bewiesen. Das angegebene vollständige System von Endlichkeitsbedingungen liefert einen Algorithmus, der zu einem gegebenen $sDOL$ -System entscheidet, ob die erzeugte Bildsprache endlich oder unendlich ist.

2 Grundlagen

Die Endlichkeitsuntersuchungen zu Bildsprachen synchroner, deterministischer Ketten-Code-Bild-Systeme im Abschnitt 3 beruhen auf einer Abstrahierungshierarchie. Die unterste Schicht bilden Wörter über dem Alphabet $\{r, l, u, d\}$. Ihnen sind Graphen unterschiedlich hoher Abstraktion zugeordnet, die verschiedene Interpretationen der Wörter darstellen. Zu jedem Ketten-Code-Bild-System über dem Alphabet $\{r, l, u, d\}$ existiert eine solche Hierarchie. Die unterste Ebene umfaßt jeweils die von dem System erzeugte Wortmenge. Die zugeordnete Graphenmenge der obersten Schicht wird als generierte Bildsprache des Systems betrachtet.

2.1 Strukturen über einem Alphabet

Es sei $\mathcal{A} = \{r, l, u, d\}$ ein Alphabet. Die Menge \mathcal{A}^* aller Zeichenketten über dem Alphabet \mathcal{A} umfaßt jene Wörter, die durch Aneinanderreihung (Konkatenation) von Buchstaben aus \mathcal{A} entstehen. Ein Operationssymbol für die Aneinanderreihung wird im allgemeinen nicht geschrieben. Die Wortmenge \mathcal{A}^* sei induktiv definiert:

1. Die atomaren Elemente sind das Leerwort λ und die Buchstaben des Alphabets \mathcal{A} :

$$\lambda, r, l, u, d \in \mathcal{A}^*.$$

2. Wenn v und w Wörter in \mathcal{A}^* sind, so liegt auch das zusammengesetzte Wort vw in der Menge \mathcal{A}^* :

$$v, w \in \mathcal{A}^* \implies vw \in \mathcal{A}^*.$$

3. Jedes Wort aus \mathcal{A}^* entsteht durch wiederholtes Anwenden von 2. aus den atomaren Elementen.

Aus dieser Definition folgt sofort die Abgeschlossenheit der Menge \mathcal{A}^* unter der Konkatenation, sowie die Assoziativität der Aneinanderreihung. Durch Verknüpfung eines Wortes mit dem Leerwort ändert sich das Wort nicht, d. h. das Leerwort λ ist neutral. Damit ist (\mathcal{A}, \cdot) eine freie Struktur über dem Alphabet \mathcal{A} mit der binären Operation der Konkatenation \cdot , und es gilt

Lemma 2.1 *Die freie Struktur (\mathcal{A}, \cdot) ist ein Monoid.*

Die Menge \mathcal{A}^* ohne Leerwort sei durch \mathcal{A}^+ symbolisiert:

$$\mathcal{A}^+ = \mathcal{A}^* \setminus \{\lambda\}.$$

Die Länge $\#\mathbf{w}$ eines Wortes $\mathbf{w} = w_1 \dots w_n$ ($w_i \in \mathcal{A}, i = 1, \dots, n$) ist die Anzahl der Buchstaben in \mathbf{w} :

$$\mathbf{w} = w_1 \dots w_n \implies \#\mathbf{w} = n.$$

Es sei \mathcal{A}_n^* die Menge aller Wörter aus \mathcal{A}^* mit der Länge n . Ein Wort $\mathbf{w} \in \mathcal{A}_n^*$ sei, wenn nicht anders angegeben, aus Buchstaben w_1, \dots, w_n zusammengesetzt:

$$\mathbf{w} = w_1 \dots w_n.$$

Zu einem Wort $\mathbf{w} \in \mathcal{A}_n^*$ und einem Buchstaben $x \in \mathcal{A}$ sei $\#_x \mathbf{w}$ die Anzahl der Vorkommen von x in \mathbf{w} :

$$\mathbf{w} = w_1 \dots w_n \implies \#_x \mathbf{w} = |\{i \mid 1 \leq i \leq n \text{ und } w_i = x\}|.$$

Folgerung 2.2 *Es sei $\mathbf{u} = \mathbf{vw}$ ein zusammengesetztes Wort mit $\mathbf{v}, \mathbf{w} \in \mathcal{A}^*$. Die Anzahl der Vorkommen eines Buchstaben x in \mathbf{u} ist die Summe der Vorkommen von x in \mathbf{v} und \mathbf{w} :*

$$\#_x \mathbf{u} = \#_x \mathbf{vw} = \#_x \mathbf{v} + \#_x \mathbf{w}.$$

Beweis: Es seien $\mathbf{v} = x_1 \dots x_n$ und $\mathbf{w} = x_{n+1} \dots x_{n+m}$ ($x_i \in \mathcal{A}, i = 1, \dots, n+m$). Dann gilt für die Gesamtzahl der Vorkommen eines Buchstaben x in $\mathbf{u} = \mathbf{vw}$

$$\begin{aligned} \#_x \mathbf{u} &= |\{i \mid 1 \leq i \leq n+m \text{ und } x_i = x\}| \\ &= |\{i \mid 1 \leq i \leq n \text{ und } x_i = x\} \cup \{i \mid n+1 \leq i \leq n+m \text{ und } x_i = x\}| \\ &= |\{i \mid 1 \leq i \leq n \text{ und } x_i = x\}| + |\{i \mid n+1 \leq i \leq n+m \text{ und } x_i = x\}| \\ &= \#_x \mathbf{v} + \#_x \mathbf{w}. \end{aligned}$$

✱

Die Summe aller Buchstabenvorkommen bildet die Länge eines Wortes:

$$\#\mathbf{w} = \#_r \mathbf{w} + \#_l \mathbf{w} + \#_u \mathbf{w} + \#_d \mathbf{w}.$$

Zu einem Wort $\mathbf{w} \in \mathcal{A}^*$ sei $[\mathbf{w}]$ die Menge der in \mathbf{w} auftretenden Buchstaben:

$$[\mathbf{w}] = \{x \mid \#_x \mathbf{w} \geq 1\}.$$

Aus der Definition von $\#_x \mathbf{w}$ resultiert

Folgerung 2.3 *Für die Buchstaben-Mengen der Wörter aus \mathcal{A}^* gilt*

1. Die Buchstaben-Menge der Leerwortes ist leer

$$[\lambda] = \emptyset.$$

2. Die Buchstaben-Menge eines Buchstaben enthält nur den Buchstaben selbst

$$\forall x \in \mathcal{A} : [x] = \{x\}.$$

3. Die Buchstaben-Menge eines zusammengesetzten Wortes $\mathbf{vw} \in \mathcal{A}^*$ ist die Vereinigung der Buchstaben-Mengen von \mathbf{v} und \mathbf{w} :

$$\forall \mathbf{v}, \mathbf{w} \in \mathcal{A}^* : [\mathbf{vw}] = [\mathbf{v}] \cup [\mathbf{w}].$$

Beweis:

1. Im Leerwort λ tritt kein Buchstabe auf ($\#\lambda = 0$).
2. Für alle Buchstaben $x \in \mathcal{A}$ gilt $\#_x x = 1$ und für alle Buchstaben $y \in \mathcal{A}$ mit $y \neq x$ gilt $\#_y x = 0$.
3. Es gilt

$$\begin{aligned} [\mathbf{vw}] &= \{x \mid \#_x \mathbf{vw} \geq 1\} \\ &= \{x \mid \#_x \mathbf{v} + \#_x \mathbf{w} \geq 1\} && \text{(Folgerung 2.2)} \\ &= \{x \mid \#_x \mathbf{v} \geq 1 \vee \#_x \mathbf{w} \geq 1\} \\ &= \{x \mid \#_x \mathbf{v} \geq 1\} \cup \{x \mid \#_x \mathbf{w} \geq 1\} \\ &= [\mathbf{v}] \cup [\mathbf{w}]. \end{aligned}$$

✱

Mit den Elementen aus \mathcal{A}^* seien Abbildungen auf dem \mathbb{Z}^2 assoziiert:

$$\mathbf{w} : \mathbb{Z}^2 \longrightarrow \mathbb{Z}^2 \quad (\mathbf{w} \in \mathcal{A}^*).$$

Die atomaren Abbildungen r, l, u, d ordnen einem Punkt $\mathbf{q} \in \mathbb{Z}^2$ seine Nachbarn zu:

$$\begin{aligned} r(\mathbf{q}) &= \mathbf{q} + (1, 0) \\ l(\mathbf{q}) &= \mathbf{q} - (1, 0) \\ u(\mathbf{q}) &= \mathbf{q} + (0, 1) \\ d(\mathbf{q}) &= \mathbf{q} - (0, 1) \end{aligned} .$$

(Die Funktionsnamen r, l, u, d stammen von den Richtungen right, left, up, down.)

Die Verschiebungen $x(\mathbf{q}) - \mathbf{q}$ eines beliebigen Punktes $\mathbf{q} \in \mathbb{Z}^2$ zu seinen Nachbarn $x(\mathbf{q})$ seien mit $\mathbf{v}_x \in \mathbb{Z}^2$ notiert:

$$\mathbf{v}_x = \begin{cases} (1, 0), & \text{falls } x = r \\ (-1, 0), & \text{falls } x = l \\ (0, 1), & \text{falls } x = u \\ (0, -1), & \text{falls } x = d \end{cases} .$$

Damit ergibt sich für den x -Nachbarn $x(\mathbf{q})$ eines Punktes $\mathbf{q} \in \mathbb{Z}^2$:

Folgerung 2.4 Für alle Abbildungen $x \in \mathcal{A}$ und Punkte $\mathfrak{q} \in \mathbb{Z}^2$ gilt

$$x(\mathfrak{q}) = \mathfrak{q} + \mathfrak{v}_x.$$

Die Abbildungen x aus \mathcal{A} sind Translationen. Daher gilt

Lemma 2.5 Für jede Abbildung $x \in \mathcal{A}$ und zwei Punkte $\mathfrak{p}, \mathfrak{q} \in \mathbb{Z}^2$ gilt

$$x(\mathfrak{p}) + \mathfrak{q} = x(\mathfrak{p} + \mathfrak{q}).$$

Beweis: Es seien $x \in \mathcal{A}$ eine Abbildung auf dem \mathbb{Z}^2 und $\mathfrak{p}, \mathfrak{q}$ zwei Punkte im \mathbb{Z}^2 . Dann gilt

$$\begin{aligned} x(\mathfrak{p}) + \mathfrak{q} &= \mathfrak{p} + \mathfrak{v}_x + \mathfrak{q} && \text{(Definition)} \\ &= \mathfrak{p} + \mathfrak{q} + \mathfrak{v}_x && \text{(Kommutativität des } \mathbb{Z}^2 \text{)} \\ &= x(\mathfrak{p} + \mathfrak{q}) && \text{(Definition),} \end{aligned}$$

womit das Lemma bewiesen ist. ✱

Jede Abbildung $x \in \mathcal{A}$ ist surjektiv (der Wertebereich stimmt mit \mathbb{Z}^2 überein), injektiv (aus $x(\mathfrak{p}) = x(\mathfrak{q})$ folgt stets $\mathfrak{p} = \mathfrak{q}$) und damit bijektiv (umkehrbar eindeutig). Zwei Abbildungen $x, y : D \rightarrow W$ heißen disjunktiv, wenn ihre Funktionswerte für jedes Argument verschieden sind.

Je zwei verschiedene Abbildungen $x, y \in \mathcal{A}$, ($x \neq y$), liefern niemals denselben Nachbarn: $\forall \mathfrak{q} \in \mathbb{Z}^2 : x(\mathfrak{q}) \neq y(\mathfrak{q})$. Das bedeutet

Folgerung 2.6 Die Abbildungen in \mathcal{A} sind paarweise disjunktiv.

Beweis: Für je zwei verschiedene Abbildungen $x, y \in \mathcal{A}$ gilt $\mathfrak{v}_x \neq \mathfrak{v}_y$, woraus

$$\mathfrak{q} + \mathfrak{v}_x \neq \mathfrak{q} + \mathfrak{v}_y$$

für alle $\mathfrak{q} \in \mathbb{Z}^2$ und damit

$$x(\mathfrak{q}) \neq y(\mathfrak{q})$$

für alle $\mathfrak{q} \in \mathbb{Z}^2$ folgt. ✱

Dem Leerwort entspricht die identische Abbildung

$$\lambda : \mathbb{Z}^2 \longrightarrow \mathbb{Z}^2 \quad \text{mit } \mathfrak{q} \mapsto \mathfrak{q}.$$

Ein zusammengesetztes Wort $\mathfrak{vw} \in \mathcal{A}^*$ symbolisiert die verkettete Abbildung $\mathfrak{v} \circ \mathfrak{w}$:

$$\mathfrak{v} \circ \mathfrak{w} : \mathbb{Z}^2 \longrightarrow \mathbb{Z}^2 \quad \text{mit } \mathfrak{q} \mapsto \mathfrak{w}(\mathfrak{v}(\mathfrak{q})).$$

Der Nullpunkt des \mathbb{Z}^2 sei mit \mathfrak{o} notiert:

$$\mathfrak{o} = (0, 0).$$

Auf verknüpfte Abbildungen erweitert, liefert das Lemma 2.5

Folgerung 2.7 Für jede verkettete Abbildung $w \in \mathcal{A}^*$ und zwei Punkte $\mathfrak{p}, \mathfrak{q} \in \mathbb{Z}^2$ gilt

$$w(\mathfrak{p} + \mathfrak{q}) = w(\mathfrak{p}) + \mathfrak{q}.$$

Beweis: Es sei $w = w_1 \circ \dots \circ w_n$ ($w_i \in \mathcal{A}, i = 1, \dots, n$). Dann gilt

$$\begin{aligned} w(\mathfrak{p} + \mathfrak{q}) &= (w_1 \circ \dots \circ w_n)(\mathfrak{p} + \mathfrak{q}) \\ &= (w_2 \circ \dots \circ w_n)(w_1(\mathfrak{p} + \mathfrak{q})) \\ &= (w_2 \circ \dots \circ w_n)(w_1(\mathfrak{p}) + \mathfrak{q}) && \text{(Lemma 2.5)} \\ &= (w_3 \circ \dots \circ w_n)(w_2(w_1(\mathfrak{p}) + \mathfrak{q})) \\ &= (w_3 \circ \dots \circ w_n)((w_1 \circ w_2)(\mathfrak{p}) + \mathfrak{q}) && \text{(Lemma 2.5)} \\ &\vdots \\ &= (w_1 \circ \dots \circ w_n)(\mathfrak{p}) + \mathfrak{q} \\ &= w(\mathfrak{p}) + \mathfrak{q}. \end{aligned}$$

✱

Insbesondere gilt damit auch

$$\begin{aligned} (v \circ w)(\mathfrak{o}) &= w(v(\mathfrak{o})) \\ &= w(\mathfrak{o} + v(\mathfrak{o})) \\ &= w(\mathfrak{o}) + v(\mathfrak{o}) && \text{(Folgerung 2.7)} \\ &= v(\mathfrak{o}) + w(\mathfrak{o}) && \text{(Kommutativität des } \mathbb{Z}^2 \text{)}. \end{aligned}$$

Die dargelegte Interpretation von Wörtern als Abbildungen auf dem \mathbb{Z}^2 ist ein Homomorphismus von der freien Struktur (\mathcal{A}, \cdot) in die freie Struktur (\mathcal{A}, \circ) . Damit ist (\mathcal{A}, \circ) ebenfalls ein Monoid. Zu jeder Abbildung $w \in \mathcal{A}^*$ existiert eine inverse Abbildung $w^{-1} \in \mathcal{A}^*$ mit $w^{-1} \circ w = w \circ w^{-1} = \lambda$. Das bedeutet

Lemma 2.8 Die algebraische Struktur (\mathcal{A}, \circ) ist eine Gruppe.

Beweis: Obigen Überlegungen zufolge ist nur noch zu zeigen, daß zu jedem Element aus \mathcal{A}^* ein Inverses bezüglich der Verkettung \circ existiert.

- Die Inverse der identischen Abbildung ist die identische Abbildung

$$\lambda^{-1} = \lambda,$$

denn $\lambda \circ \lambda = \lambda$.

- Die Inversen der atomaren Abbildungen sind

$$r^{-1} = l, \quad l^{-1} = r, \quad u^{-1} = d, \quad d^{-1} = u,$$

denn damit gilt

$$\begin{aligned}
 \mathbf{v}_{r^{-1}} &= \mathbf{v}_l & \mathbf{v}_{u^{-1}} &= \mathbf{v}_d \\
 &= -\mathbf{v}_r & &= -\mathbf{v}_u & \text{(Definition)} \\
 &= -\mathbf{v}_{l^{-1}} & &= -\mathbf{v}_{d^{-1}} & \text{(Definition)}
 \end{aligned}$$

bzw. allgemein

$$\mathbf{v}_x = -\mathbf{v}_{x^{-1}} \quad (x \in \mathcal{A}),$$

und daher gilt für alle Abbildungen $x \in \mathcal{A}$ und Punkte $\mathbf{q} \in \mathbb{Z}^2$:

$$\begin{aligned}
 (x \circ x^{-1})(\mathbf{q}) &= x^{-1}(x(\mathbf{q})) \\
 &= x^{-1}(\mathbf{q} + \mathbf{v}_x) && \text{(Folgerung 2.4)} \\
 &= x^{-1}(\mathbf{q}) + \mathbf{v}_x && \text{(Lemma 2.5)} \\
 &= \mathbf{q} + \mathbf{v}_{x^{-1}} + \mathbf{v}_x && \text{(Folgerung 2.4)} \\
 &= \mathbf{q} + \mathbf{v}_{x^{-1}} - \mathbf{v}_{x^{-1}} \\
 &= \mathbf{q} \\
 &= \mathbf{q} + \mathbf{v}_{x^{-1}} - \mathbf{v}_{x^{-1}} \\
 &= \mathbf{q} + \mathbf{v}_{x^{-1}} + \mathbf{v}_x \\
 &= x(\mathbf{q} + \mathbf{v}_{x^{-1}}) && \text{(Folgerung 2.4)} \\
 &= x(x^{-1}(\mathbf{q})) && \text{(Folgerung 2.4)} \\
 &= (x^{-1} \circ x)(\mathbf{q}).
 \end{aligned}$$

Daraus folgt für alle $x \in \mathcal{A}$: Die verknüpften Abbildungen $x \circ x^{-1}$ und $x^{-1} \circ x$ bilden genau dann die identische Abbildung λ , wenn $\mathbf{v}_x = -\mathbf{v}_{x^{-1}}$ gilt, d. h. die inverse Abbildung x^{-1} zu $x \in \mathcal{A}$ ist jene Abbildung $y \in \mathcal{A}$ mit $\mathbf{v}_x = -\mathbf{v}_y$. Zu jedem $x \in \mathcal{A}$ gibt es genau ein $y \in \mathcal{A}$, das diese Bedingung erfüllt.

- Es sei $\mathbf{w} = w_1 \circ \dots \circ w_n$ ($w_i \in \mathcal{A}, i = 1, \dots, n$) eine verkettete Abbildung. Dann ist die inverse Abbildung

$$\mathbf{w}^{-1} = w_n^{-1} \circ \dots \circ w_1^{-1},$$

denn es gilt

$$\begin{aligned}
 \mathbf{w} \circ \mathbf{w}^{-1} &= w_1 \circ \dots \circ w_n \circ w_n^{-1} \circ \dots \circ w_1^{-1} \\
 &= w_1 \circ \dots \circ w_{n-1} \circ w_{n-1}^{-1} \circ \dots \circ w_1^{-1} \\
 &= \vdots \\
 &= w_1 \circ w_1^{-1} \\
 &= \lambda \\
 &= w_n^{-1} \circ w_n \\
 &= w_n^{-1} \circ w_{n-1}^{-1} \circ w_{n-1} \circ w_n \\
 &= \vdots \\
 &= w_n^{-1} \circ \dots \circ w_1^{-1} \circ w_1 \circ \dots \circ w_n \\
 &= \mathbf{w}^{-1} \circ \mathbf{w}.
 \end{aligned}$$



Der Operator \circ muß nicht geschrieben werden, wenn aus dem Zusammenhang hervorgeht, um welche Operation es sich handelt. So impliziert beispielsweise $(x_1x_2)(\mathbf{o})$, daß x_1x_2 die verkettete Abbildung $x_1 \circ x_2$ symbolisiert, wogegen x_1x_2 in $[x_1x_2]$ für das zusammengesetzte Wort x_1x_2 steht.

Zu $x \in \mathcal{A}$ seien \bar{x} , x^\perp , \bar{x}^\perp wie folgt erklärt:

x	\bar{x}	x^\perp	\bar{x}^\perp
r	l	u	d
l	r	d	u
u	d	r	l
d	u	l	r

Als Abbildungen interpretiert, sind x und \bar{x} bzw. x^\perp und \bar{x}^\perp zueinander invers. Die Abbildungen xx^\perp und $x^\perp x$ bilden einen Punkt $\mathbf{q} \in \mathbb{Z}^2$ auf den Diagonalnachbarn $\mathbf{q} + (1, 1)$ (falls $x = r$) bzw. $\mathbf{q} - (1, 1)$ (falls $x = l$) ab.

2.2 Graphen

Für Graphen existieren unterschiedliche Definitionen. Mit den meisten sind jedoch nur spezielle Graphen – wie reine gerichtete bzw. ungerichtete oder solche ohne Mehrfachkanten und Schlingen – erklärt (z. B. in [M⁺94]). Hinzu kommt, daß eine gewisse Vereinigung von Graphen nicht oder nur umständlich beschrieben werden kann. In [Hol99] ist eine allgemeinere Definition zu finden, die auch gemischte Graphen mit Mehrfachkanten und Schlingen zuläßt, sowie auf einfache Weise ermöglicht, Graphen um neue Kanten zu erweitern. Aus diesem Grunde wird sie in der vorliegenden Arbeit verwendet. Im folgenden seien die hier benötigten Begriffe zusammengestellt. Ein Graph ist eine Struktur

$$G = (V, R_u, R_g)$$

mit einer endlichen Menge V , einer endlichen Menge R_u von symmetrischen Relationen auf V und einer endlichen Menge R_g von asymmetrischen Relationen auf V . Die Elemente aus V heißen Knoten. Eine Zweiermenge $\{(x, y), (y, x)\}$ heißt ungerichtete Kante, wenn die Paare (x, y) und (y, x) in der gleichen symmetrischen Relation stehen. Eine Einermenge $\{(x, y)\}$ heißt gerichtete Kante, wenn das Paar (x, y) aus einer asymmetrischen Relation ist. Sind alle definierenden Relationen symmetrisch, heißt der Graph ungerichtet; sind alle definierenden Relationen asymmetrisch, heißt der Graph gerichtet. Ein Graph heißt schlicht, wenn alle definierenden Relationen irreflexiv und je zwei von ihnen elementfremd sind. Der Adjazenzgrad $a(x, y)$ ist die Anzahl der Kanten vom Knoten x zum Knoten y , d. h. die Anzahl der definierenden Relationen, die das Paar (x, y) enthalten. Sollte es zu einem Knoten x keine Kante geben, die zu x hin- oder von x wegführt, heißt der Knoten x isoliert. Ein Graph heißt zusammenhängend, wenn er genau einen Knoten oder keine isolierten Knoten

enthält. Der Schatten G^u eines Graphen G ist ein ungerichteter Graph mit der gleichen Knotenmenge wie G , wobei für die Anzahl $a^u(x, y)$ der Kanten zwischen zwei beliebigen Knoten x, y gilt

$$a^u(x, y) = \max \{ a(x, y), a(y, x) \},$$

wobei $a(p, q)$ die Anzahl der Kanten vom Knoten p zum Knoten q im Graphen G darstellt.

Zu jedem Punkt $\mathbf{a} \in \mathbb{Z}^2$ sei $g^{\mathbf{a}}$ eine Funktion, die einem Wort $w \in \mathcal{A}_n^*$ den gerichteten Graphen

$$g^{\mathbf{a}}(w) = (\{\mathbf{a}, w_1(\mathbf{a}), (w_1w_2)(\mathbf{a}), \dots, w(\mathbf{a})\}, \\ \{\{\mathbf{a}, w_1(\mathbf{a})\}, \{(w_1(\mathbf{a}), (w_1w_2)(\mathbf{a}))\}, \dots, \{(w_1 \dots w_{n-1})(\mathbf{a}), w(\mathbf{a})\}\})$$

zuordnet.

Als Beispiel sei der Graph $g^{\mathbf{a}}(w)$ zu dem Wort $w = rudlrud$ und dem Bezugspunkt $\mathbf{a} = \mathbf{o}$ angegeben:

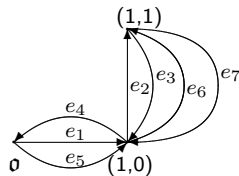
Die Knoten sind

$$\begin{aligned} k_1 &= \lambda(\mathbf{o}) &= \mathbf{o}, \\ k_2 &= r(\mathbf{o}) &= (1, 0), \\ k_3 &= (ru)(\mathbf{o}) &= (1, 1), \\ k_4 &= (rud)(\mathbf{o}) &= (1, 0), \\ k_5 &= (rudl)(\mathbf{o}) &= \mathbf{o}, \\ k_6 &= (rudlr)(\mathbf{o}) &= (1, 0), \\ k_7 &= (rudlru)(\mathbf{o}) &= (1, 1), \\ k_8 &= (rudlrud)(\mathbf{o}) &= (1, 0), \end{aligned}$$

d. h. die Knotenmenge ist $\{\mathbf{o}, (1, 0), (1, 1)\}$. Die Kanten entstehen durch aufeinanderfolgendes Verbinden der Knoten: $e_1 = (k_1, k_2)$, $e_2 = (k_2, k_3)$, \dots , $e_7 = (k_7, k_8)$. Entsprechend der obigen Schreibweise kann der Graph angegeben werden durch

$$\begin{aligned} g^{\mathbf{o}}(rudlrud) &= (\{k_1, k_2, k_3\}, \{e_1\}, \dots, \{e_7\}) \\ &= (\{\mathbf{o}, (1, 0), (1, 1)\}, \{\{\mathbf{o}, (1, 0)\}\}, \{(1, 0), (1, 1)\}\}, \\ &\quad \{(1, 1), (1, 0)\}\}, \{(1, 0), \mathbf{o}\}\}, \{\{\mathbf{o}, (1, 0)\}\}, \\ &\quad \{(1, 0), (1, 1)\}\}, \{(1, 1), (1, 0)\}\}) \end{aligned}$$

oder auch bildlich:



Die textuale Angabe ist keine Minimalbeschreibung im Sinne von [Hol99]; eine Zusammenlegung von Einerrelationen ist im allgemeinen nicht verlustfrei (Kanten zwischen gleichen Knoten gehen verloren oder werden zu ungerichteten Kanten). Das vorstehende Beispiel soll die Definition veranschaulichen; daher bildet auch hier jede Kante eine Relation.

Aus der obigen Definition erkennt man sofort

Folgerung 2.9 *Zu jedem Punkt $\mathbf{a} \in \mathbb{Z}^2$ und jedem Wort $\mathbf{w} \in \mathcal{A}^*$ ist der Graph $g^{\mathbf{a}}(\mathbf{w})$ zusammenhängend.*

Nach Folgerung 2.7 kann man für einen Graphen $g^{\mathbf{a}}(\mathbf{w})$ auch schreiben:

$$g^{\mathbf{a}}(\mathbf{w}) = (\{\mathbf{o} + \mathbf{a}, w_1(\mathbf{o}) + \mathbf{a}, (w_1w_2)(\mathbf{o}) + \mathbf{a}, \dots, \mathbf{w}(\mathbf{o}) + \mathbf{a}\}, \\ \{(\mathbf{o} + \mathbf{a}, w_1(\mathbf{o}) + \mathbf{a})\}, \{(w_1(\mathbf{o}) + \mathbf{a}, (w_1w_2)(\mathbf{o}) + \mathbf{a})\}, \dots, \\ \{((w_1 \dots w_{n-1})(\mathbf{o}) + \mathbf{a}, \mathbf{w}(\mathbf{o}) + \mathbf{a})\}),$$

d. h. $g^{\mathbf{a}}(\mathbf{w})$ geht aus $g^{\mathbf{o}}(\mathbf{w})$ durch Translation um \mathbf{a} hervor. Stimmt der Bezugspunkt mit dem Nullpunkt überein, muß er nicht angegeben werden: $g^{\mathbf{o}} = g$.

Die Vereinigung zweier gerichteter Graphen

$$g = (V, R_1, \dots, R_n) \quad \text{und} \quad f = (U, Q_1, \dots, Q_m)$$

sei wie folgt erklärt:

$$g \cup f = (V \cup U, R_1, \dots, R_n, Q_1, \dots, Q_m).$$

Dies ermöglicht eine Verkettung zweier gerichteter Graphen:

Folgerung 2.10 *Der Verknüpfung von Wörtern ist eine Vereinigung von gerichteten Graphen zugeordnet: Für jeden Punkt $\mathbf{a} \in \mathbb{Z}^2$ und zwei beliebige Wörter $\mathbf{v}, \mathbf{w} \in \mathcal{A}^*$ gilt*

$$g^{\mathbf{a}}(\mathbf{vw}) = g^{\mathbf{a}}(\mathbf{v}) \cup g^{\mathbf{v}(\mathbf{a})}(\mathbf{w}).$$

Beweis: Es seien \mathbf{v} ein Wort aus \mathcal{A}_n^* und \mathbf{w} ein Wort aus \mathcal{A}_m^* . Für die Vereinigung von $g^{\mathbf{a}}(\mathbf{v})$ und $g^{\mathbf{v}(\mathbf{a})}(\mathbf{w})$ gilt:

$$\begin{aligned} & g^{\mathbf{a}}(\mathbf{v}) \cup g^{\mathbf{v}(\mathbf{a})}(\mathbf{w}) \\ &= (\{\mathbf{a}, v_1(\mathbf{a}), (v_1v_2)(\mathbf{a}), \dots, \mathbf{v}(\mathbf{a})\} \\ & \quad \cup \{\mathbf{v}(\mathbf{a}), w_1(\mathbf{v}(\mathbf{a})), (w_1w_2)(\mathbf{v}(\mathbf{a})), \dots, \mathbf{w}(\mathbf{v}(\mathbf{a}))\}, \\ & \quad \{(\mathbf{a}, v_1(\mathbf{a}))\}, \{(v_1(\mathbf{a}), (v_1v_2)(\mathbf{a}))\}, \dots, \{((v_1 \dots v_{n-1})(\mathbf{a}), \mathbf{v}(\mathbf{a}))\}, \\ & \quad \{(\mathbf{v}(\mathbf{a}), w_1(\mathbf{v}(\mathbf{a})))\}, \{(w_1(\mathbf{v}(\mathbf{a})), (w_1w_2)(\mathbf{v}(\mathbf{a})))\}, \dots, \\ & \quad \{((w_1 \dots w_{m-1})(\mathbf{v}(\mathbf{a})), \mathbf{w}(\mathbf{v}(\mathbf{a})))\}) \\ &= (\{\mathbf{a}, v_1(\mathbf{a}), (v_1v_2)(\mathbf{a}), \dots, \mathbf{v}(\mathbf{a}), (\mathbf{vw}_1)(\mathbf{a}), (\mathbf{vw}_1w_2)(\mathbf{a}), \dots, (\mathbf{vw})(\mathbf{a})\}, \\ & \quad \{(\mathbf{a}, v_1(\mathbf{a}))\}, \{(v_1(\mathbf{a}), (v_1v_2)(\mathbf{a}))\}, \dots, \{((v_1 \dots v_{n-1})(\mathbf{a}), \mathbf{v}(\mathbf{a}))\}, \\ & \quad \{(\mathbf{v}(\mathbf{a}), (\mathbf{vw}_1)(\mathbf{a}))\}, \{((\mathbf{vw}_1)(\mathbf{a}), (\mathbf{vw}_1w_2)(\mathbf{a}))\}, \dots, \\ & \quad \{((\mathbf{vw}_1 \dots w_{m-1})(\mathbf{a}), (\mathbf{vw})(\mathbf{a}))\}) \\ &= g^{\mathbf{a}}(\mathbf{vw}), \end{aligned}$$

womit die Folgerung bewiesen ist. ✱

Man kann sich den Zusammenhang anhand der Arbeitsweise eines Plotters veranschaulichen: Der Zeichenstift steht auf dem Punkt \mathfrak{a} eines Rasters. Nacheinander werden die Buchstaben eines Eingabewortes in Befehle umgesetzt und diese ausgeführt:

- r : zeichne einen Pfeil nach rechts zum nächsten Rasterpunkt,
- l : zeichne einen Pfeil nach links zum nächsten Rasterpunkt,
- u : zeichne einen Pfeil nach oben zum nächsten Rasterpunkt,
- d : zeichne einen Pfeil nach unten zum nächsten Rasterpunkt.

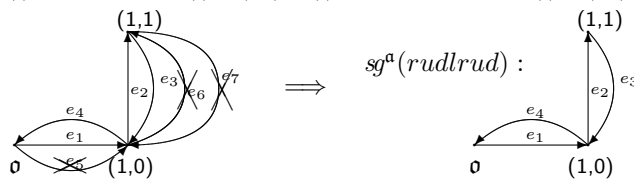
Je öfter der Stift über einen bereits gezeichneten Pfeil geführt wird, desto dicker ist der Pfeil. Dies entspricht den Mehrfachkanten. Angenommen, das Eingabewort ist *rudlrud*; der Plotterstift möge auf dem Nullpunkt stehen (s. obiges Beispiel). Zuerst zeichnet der Stift einen Pfeil vom Nullpunkt \mathfrak{o} nach rechts zum Punkt $r(\mathfrak{o}) = (1, 0)$; dann von dort nach oben zum Punkt $u(1,0) = u(r(\mathfrak{o})) = (ru)(\mathfrak{o}) = (1,1)$, danach nach unten zum Punkt $d(1,1) = (1,0)$ usw. Nachdem alle Befehle ausgeführt wurden, steht der Stift auf dem Punkt $(rudlrud)(\mathfrak{o}) = (1,0)$, oder allgemein: Nach der Abarbeitung eines Wortes w steht der Zeichenstift auf dem Punkt $w(\mathfrak{a})$. Angenommen das Wort w besteht aus zwei Teilwörtern u, v . Der Graph zum Wort u wird im Punkte \mathfrak{a} beginnend gezeichnet; der Zeichenstift steht am Ende auf dem Punkt $u(\mathfrak{a})$. Dies ist der Startpunkt für den Graphen von v . Am Ende des Wortes w steht der Stift auf dem Punkt $v(u(\mathfrak{a})) = (uv)(\mathfrak{a}) = w(\mathfrak{a})$. Damit sind einem Graphen ein wohlbestimmter Start- und ein wohlbestimmter Endpunkt zugeordnet. Der Graph eines zusammengesetzten Wortes uv entsteht, indem die Graphen der Einzelwörter u, v so zueinander verschoben werden, daß der Endpunkt vom ersten Graphen mit dem Startpunkt des zweiten Graphen zusammenfällt.

Zu jedem Punkt $\mathfrak{a} \in \mathbb{Z}^2$ sei $sg^{\mathfrak{a}}$ eine Funktion, die einem Wort $w \in \mathcal{A}^*$ jenen schlichten gerichteten Graphen $sg^{\mathfrak{a}}(w)$ zuordnet, der aus dem Graphen $g^{\mathfrak{a}}(w)$ durch Streichen der Kantenwiederholungen entsteht, d. h. $sg^{\mathfrak{a}}(w)$ hat die gleiche Knotenmenge wie $g^{\mathfrak{a}}(w)$, und für die Anzahl $a^{sg}(x, y)$ der Kanten von einem beliebigen Knoten x zu einem Knoten y gilt

$$a^{sg}(x, y) = \begin{cases} 0, & \text{falls } a^g(x, y) = 0 \\ 1, & \text{sonst} \end{cases} = \min \{ 1, a^g(x, y) \} \leq 1,$$

wobei $a^g(x, y)$ die Anzahl der Kanten von x nach y im Graphen $g^{\mathfrak{a}}(w)$ ist. Der Bezugspunkt muß nur angegeben werden, wenn er vom Nullpunkt abweicht: $sg^{\mathfrak{o}} = sg$.

Der gerichtete Graph zum obigen Beispielwort *rudlrud* enthält Mehrfachkanten; diese sind die doppelt auftretenden Kanten $(\mathfrak{o}, (1,0))$, $((1,0), (1,1))$ und $((1,1), (1,0))$. Der schlichte gerichtete Graph $sg^{\mathfrak{a}}(\textit{rudlrud})$ entsteht daher durch Streichen einer Kante $(\mathfrak{o}, (1,0))$, einer Kante $((1,0), (1,1))$ und einer Kante $((1,1), (1,0))$:



Von jeder Kante in einem Graphen $g^{\mathbf{a}}(\mathbf{w})$ bleibt beim Abstrahieren zu $sg^{\mathbf{a}}(\mathbf{w})$ ein Repräsentant erhalten. Damit gilt

Folgerung 2.11 *Zu jedem Punkt $\mathbf{a} \in \mathbb{Z}^2$ und einem beliebigen Wort $\mathbf{w} \in \mathcal{A}^*$ stimmen die Kantenmengen der Graphen $g^{\mathbf{a}}(\mathbf{w})$ und $sg^{\mathbf{a}}(\mathbf{w})$ überein.*

Beweis: Es seien \mathbf{w} ein Wort aus \mathcal{A}^* , $g^{\mathbf{a}}(\mathbf{w})$ der gerichtete Graph zu \mathbf{w} und $R^{\mathbf{a}}(\mathbf{w})$ dessen Kantenmenge:

$$g^{\mathbf{a}}(\mathbf{w}) = (V, R_1, \dots, R_n), \quad R^{\mathbf{a}}(\mathbf{w}) = \bigcup_{i=1}^n R_i,$$

sowie $sg^{\mathbf{a}}(\mathbf{w})$ der schlichte gerichtete Graph zu \mathbf{w} mit der Kantenmenge $Q^{\mathbf{a}}(\mathbf{w})$:

$$sg^{\mathbf{a}}(\mathbf{w}) = (V, Q_1, \dots, Q_m), \quad Q^{\mathbf{a}}(\mathbf{w}) = \bigcup_{i=1}^m Q_i.$$

Jedes Knotenpaar (x, y) , das in einer Relation R_i ($i = 1, \dots, n$) steht, ist durch eine Kante verbunden, d. h. $a^g(x, y) \geq 1$. Damit ist $a^{sg}(x, y) = 1$, d. h. im Graphen $sg^{\mathbf{a}}(\mathbf{w})$ gibt es eine Kante von x nach y : $(x, y) \in Q^{\mathbf{a}}(\mathbf{w})$. Das bedeutet, daß jede Kante von $g^{\mathbf{a}}(\mathbf{w})$ auch in $sg^{\mathbf{a}}(\mathbf{w})$ auftritt:

$$R^{\mathbf{a}}(\mathbf{w}) \subseteq Q^{\mathbf{a}}(\mathbf{w}).$$

Angenommen, es gibt ein Element $(x, y) \in Q^{\mathbf{a}}(\mathbf{w})$, welches nicht in $R^{\mathbf{a}}(\mathbf{w})$ vorkommt. Dann gibt es eine Relation Q_j , $j \in \{1, \dots, m\}$, aber keine Relation R_i , $i = 1, \dots, n$, die das Knotenpaar (x, y) enthält:

$$a^{sg}(x, y) = 1 \quad \text{und} \quad a^g(x, y) = 0.$$

Dies ist ein Widerspruch zur Definition von $sg^{\mathbf{a}}$; somit gilt

$$Q^{\mathbf{a}}(\mathbf{w}) \subseteq R^{\mathbf{a}}(\mathbf{w})$$

und insgesamt

$$Q^{\mathbf{a}}(\mathbf{w}) = R^{\mathbf{a}}(\mathbf{w}).$$

✱

Es sei $sg^{\mathbf{a}}(\mathbf{w}) = (V, R_1, \dots, R_n)$ der schlichte gerichtete Graph zu einem Wort $\mathbf{w} \in \mathcal{A}^*$ und einem Bezugspunkt $\mathbf{a} \in \mathbb{Z}^2$. Im Falle $\mathbf{w} = \lambda$ besteht der Graph $sg^{\mathbf{a}}(\mathbf{w})$ aus dem Knoten \mathbf{a} ($n = 0$); in jedem anderen Falle enthält er mindestens zwei Knoten, und keiner ist isoliert (nach Folgerung 2.9). Von einem Knoten zu einem anderen in $sg^{\mathbf{a}}(\mathbf{w})$ führt höchstens eine Kante. Folglich wird der Graph vollständig durch seine Kantenmenge

$$K^{\mathbf{a}}(\mathbf{w}) = \bigcup_{i=1}^n R_i$$

beschrieben. Die Kantenmenge $K^{\mathbf{a}}(\mathbf{w})$ ist genau dann leer, wenn \mathbf{w} das Leerwort ist:

$$K^{\mathbf{a}}(\mathbf{w}) = \emptyset \iff \mathbf{w} = \lambda.$$

Alle Kanten in $K^{\mathbf{a}}(\mathbf{w})$ haben die Form $(\mathbf{q}, x(\mathbf{q}))$ mit $\mathbf{q} \in V$ und $x \in [\mathbf{w}]$ (folgt aus den Definitionen von $g^{\mathbf{a}}$ und $sg^{\mathbf{a}}$). Das bedeutet: Zu jeder Kante $(\mathbf{q}, \tilde{\mathbf{q}}) \in K^{\mathbf{a}}(\mathbf{w})$ gibt es mindestens ein $x \in \mathcal{A}$ mit $x(\mathbf{q}) = \tilde{\mathbf{q}}$. Auf Grund der Disjunktivität der Abbildungen in \mathcal{A} (Folgerung 2.6) gibt es höchstens ein $x \in \mathcal{A}$ mit $x(\mathbf{q}) = \tilde{\mathbf{q}}$. Somit läßt sich jeder Kante $(\mathbf{q}, \tilde{\mathbf{q}}) \in K^{\mathbf{a}}(\mathbf{w})$ umkehrbar eindeutig das Paar (\mathbf{q}, x) mit $x(\mathbf{q}) = \tilde{\mathbf{q}}$ zuordnen. Dem Graphen $sg^{\mathbf{a}}(\mathbf{w})$ ist daher die Menge

$$\|{}^{\mathbf{a}}\mathbf{w} = \{ (\mathbf{q}, x) \mid (\mathbf{q}, x(\mathbf{q})) \in K^{\mathbf{a}}(\mathbf{w}) \}$$

umkehrbar eindeutig zugeordnet. Diese Menge wird im folgenden Kantenmenge von \mathbf{w} bezüglich \mathbf{a} genannt (eine Kante ist durch ihren Anfangspunkt \mathbf{q} und die Richtung \mathbf{v}_x bestimmt). Der Bezugspunkt muß nicht angegeben werden, wenn es sich um den Nullpunkt handelt: $\|{}^{\circ} = \|\cdot$.

Es sei \mathbf{w} ein Wort aus \mathcal{A}_n^* . Dann sind

$$g^{\mathbf{a}}(\mathbf{w}) = \{ (\mathbf{a}, w_1(\mathbf{a})), (w_1 w_2)(\mathbf{a}), \dots, \mathbf{w}(\mathbf{a}) \}, \\ \{ (\mathbf{a}, w_1(\mathbf{a})) \}, \{ (w_1(\mathbf{a}), (w_1 w_2)(\mathbf{a})) \}, \dots, \{ ((w_1 \dots w_{n-1})(\mathbf{a}), \mathbf{w}(\mathbf{a})) \}$$

der zugeordnete gerichtete Graph und

$$\|{}^{\mathbf{a}}\mathbf{w} = \{ (\mathbf{a}, w_1), (w_1(\mathbf{a}), w_2), ((w_1 w_2)(\mathbf{a}), w_3), \dots, ((w_1 \dots w_{n-1})(\mathbf{a}), w_n) \}$$

die zugeordnete Kantenmenge bezüglich $\mathbf{a} \in \mathbb{Z}^2$.

Die Kantenmenge zum obigen Beispiel $\mathbf{w} = rudlrud$, $\mathbf{a} = \mathbf{o}$ lautet

$$\|{}^{\mathbf{o}}rudlrud = \{ (\mathbf{o}, r), ((1, 0), u), ((1, 1), d), ((1, 0), l) \}.$$

Eine Graphenverkettung schlichter gerichteter Graphen soll einen schlichten gerichteten Graphen liefern. Bei der oben eingeführten Graphenvereinigung ist dies nicht gewährleistet; es entstehen möglicherweise Mehrfachkanten (Folgerung 2.10). Das geschieht nicht bei der Vereinigung von Kantenmengen. Somit wird eine Verkettung schlichter gerichteter Graphen als Vereinigung der entsprechenden Kantenmengen erklärt:

Folgerung 2.12 *Der Verknüpfung von Wörtern ist eine Vereinigung von schlichten gerichteten Graphen zugeordnet: Für zwei beliebige Wörter $\mathbf{v}, \mathbf{w} \in \mathcal{A}^*$ und jeden Punkt $\mathbf{a} \in \mathbb{Z}^2$ gilt*

$$\|{}^{\mathbf{a}}\mathbf{vw} = \|{}^{\mathbf{a}}\mathbf{v} \cup \|{}^{\mathbf{v}(\mathbf{a})}\mathbf{w}.$$

Beweis: Es seien \mathbf{v} ein Wort aus \mathcal{A}_n^* und \mathbf{w} ein Wort aus \mathcal{A}_m^* . Für die Vereinigung der Kantenmengen zu \mathbf{v} bezüglich \mathbf{a} und \mathbf{w} bezüglich $\mathbf{v}(\mathbf{a})$ gilt

$$\begin{aligned}
& \|\mathbf{a}\mathbf{v} \cup \|\mathbf{v}(\mathbf{a})\mathbf{w} \\
&= \{(\mathbf{a}, v_1), (v_1(\mathbf{a}), v_2), ((v_1v_2)(\mathbf{a}), v_3), \dots, ((v_1 \dots v_{n-1})(\mathbf{a}), v_n)\} \\
&\quad \cup \{(\mathbf{v}(\mathbf{a}), w_1), (w_1(\mathbf{v}(\mathbf{a})), w_2), ((w_1w_2)(\mathbf{v}(\mathbf{a})), w_3), \dots, \\
&\quad \quad ((w_1 \dots w_{m-1})(\mathbf{v}(\mathbf{a})), w_m)\} \\
&= \{(\mathbf{a}, v_1), (v_1(\mathbf{a}), v_2), ((v_1v_2)(\mathbf{a}), v_3), \dots, ((v_1 \dots v_{n-1})(\mathbf{a}), v_n), \\
&\quad (\mathbf{v}(\mathbf{a}), w_1), (\mathbf{v}w_1)(\mathbf{a}), w_2), ((\mathbf{v}w_1w_2)(\mathbf{a}), w_3), \dots, ((\mathbf{v}w_1 \dots w_{m-1})(\mathbf{a}), w_m)\} \\
&= \|\mathbf{a}\mathbf{v}\mathbf{w}.
\end{aligned}$$

✱

Dieser Abstraktion entspricht ein Plotter P_{sg} , der so arbeitet wie der Plotter P_g auf der vorherigen Stufe (der die Graphen $g^{\mathbf{a}}(\mathbf{w})$ zeichnet) – P_{sg} zeichnet genau dann einen Pfeil, wenn P_g es tut (Folgerung 2.11). Bei P_{sg} sind die Pfeile jedoch gleich dick, d. h. an einem Graphen ist nicht mehr zu erkennen, wie oft der Stift einen Pfeil gezeichnet hat. Unterschiedliche Zeichenrichtungen auf einer Rasterstrecke sind aber nach wie vor zu sehen (die Strecke hat an beiden Enden eine Spitze).

Zu jedem Punkt $\mathbf{a} \in \mathbb{Z}^2$ sei $p^{\mathbf{a}}$ eine Funktion, die einem Wort $\mathbf{w} \in \mathcal{A}^*$ den Schatten $p^{\mathbf{a}}(\mathbf{w})$ des schlichten gerichteten Graphen $sg^{\mathbf{a}}(\mathbf{w})$ zuordnet, d. h. $p^{\mathbf{a}}(\mathbf{w})$ ist ein ungerichteter Graph mit der gleichen Knotenmenge wie $sg^{\mathbf{a}}(\mathbf{w})$, wobei für die Anzahl $a^p(x, y)$ der Kanten zwischen zwei beliebigen Knoten x, y gilt

$$\begin{aligned}
a^p(x, y) &= \max \{ a^{sg}(x, y), a^{sg}(y, x) \} \\
&= \begin{cases} 0, & \text{falls } a^{sg}(x, y) = 0 \text{ und } a^{sg}(y, x) = 0 \\ 1, & \text{sonst} \end{cases} .
\end{aligned}$$

Der Graph $p^{\mathbf{a}}(\mathbf{w})$ heißt Bild von \mathbf{w} bezüglich \mathbf{a} . Im Falle $\mathbf{a} = \mathbf{o}$ muß der Index nicht geschrieben werden: $p^{\mathbf{o}} = p$.

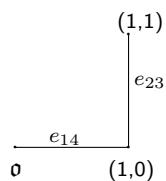
Es sei \mathbf{w} ein Wort aus \mathcal{A}_n^* . Dann ist das zugeordnete Bild bezüglich $\mathbf{a} \in \mathbb{Z}^2$:

$$\begin{aligned}
p^{\mathbf{a}}(\mathbf{w}) &= (\{\mathbf{a}, w_1(\mathbf{a}), (w_1w_2)(\mathbf{a}), \dots, \mathbf{w}(\mathbf{a})\}, \\
&\quad \{\mathbf{a}, w_1(\mathbf{a}), (w_1(\mathbf{a}), (w_1w_2)(\mathbf{a})), \dots, ((w_1 \dots w_{n-1})(\mathbf{a}), \mathbf{w}(\mathbf{a})), \\
&\quad (\mathbf{w}(\mathbf{a}), (w_1 \dots w_{n-1})(\mathbf{a})), \dots, ((w_1w_2)(\mathbf{a}), w_1(\mathbf{a})), (w_1(\mathbf{a}), \mathbf{a})\}).
\end{aligned}$$

Das Bild zum obigen Beispiel $\mathbf{w} = \text{rudlrud}$, $\mathbf{a} = \mathbf{o}$ besteht aus den Knoten \mathbf{o} , $(1, 0)$ und $(1, 1)$, sowie den (ungerichteten) Kanten $e_{14} = \{e_1, e_4\} = \{(\mathbf{o}, (1, 0)), ((1, 0), \mathbf{o})\}$ und $e_{23} = \{e_2, e_3\} = \{((1, 0), (1, 1)), ((1, 1), (1, 0))\}$:

$$\begin{aligned}
p(\text{rudlrud}) &= (\{\mathbf{o}, (1, 0), (1, 1)\}, \\
&\quad \{(\mathbf{o}, (1, 0)), ((1, 0), \mathbf{o}), ((1, 0), (1, 1)), ((1, 1), (1, 0))\})
\end{aligned}$$

oder bildlich:



Die Vereinigung zweier Bilder $p = (V, R)$, $q = (U, Q)$ sei wie folgt erklärt

$$p \cup q = (V \cup U, R \cup Q).$$

Analog zu den Vereinigungen gerichteter und schlichter gerichteter Graphen können auch zwei Bilder verkettet werden:

Folgerung 2.13 *Der Verknüpfung von Wörtern ist eine Vereinigung von Bildern zugeordnet: Für jeden Punkt $\mathbf{a} \in \mathbb{Z}^2$ und zwei beliebige Wörter $\mathbf{v}, \mathbf{w} \in \mathcal{A}^*$ gilt*

$$p^{\mathbf{a}}(\mathbf{vw}) = p^{\mathbf{a}}(\mathbf{v}) \cup p^{\mathbf{v}(\mathbf{a})}(\mathbf{w}).$$

Beweis: Es seien \mathbf{v} ein Wort aus \mathcal{A}_n^* und \mathbf{w} ein Wort aus \mathcal{A}_m^* . Dann gilt für die Verknüpfung der Bilder $p^{\mathbf{a}}(\mathbf{v})$ und $p^{\mathbf{v}(\mathbf{a})}(\mathbf{w})$:

$$\begin{aligned} p^{\mathbf{a}}(\mathbf{v}) \cup p^{\mathbf{v}(\mathbf{a})}(\mathbf{w}) &= (\{\mathbf{a}, v_1(\mathbf{a}), (v_1v_2)(\mathbf{a}), \dots, \mathbf{v}(\mathbf{a})\}, \\ &\quad \{\mathbf{a}, v_1(\mathbf{a}), \dots, ((v_1 \dots v_{n-1})(\mathbf{a}), \mathbf{v}(\mathbf{a})), \\ &\quad (\mathbf{v}(\mathbf{a}), (v_1 \dots v_{n-1})(\mathbf{a})), \dots, (v_1(\mathbf{a}), \mathbf{a})\}) \\ &\cup (\{\mathbf{v}(\mathbf{a}), w_1(\mathbf{v}(\mathbf{a})), (w_1w_2)(\mathbf{v}(\mathbf{a})), \dots, \mathbf{w}(\mathbf{v}(\mathbf{a}))\}, \\ &\quad \{\mathbf{v}(\mathbf{a}), w_1(\mathbf{v}(\mathbf{a})), \dots, ((w_1 \dots w_{m-1})(\mathbf{v}(\mathbf{a})), \mathbf{w}(\mathbf{v}(\mathbf{a}))), \\ &\quad (\mathbf{w}(\mathbf{v}(\mathbf{a})), (w_1 \dots w_{m-1})(\mathbf{v}(\mathbf{a}))), \dots, (w_1(\mathbf{v}(\mathbf{a})), \mathbf{v}(\mathbf{a}))\}) \\ &= (\{\mathbf{a}, v_1(\mathbf{a}), \dots, \mathbf{v}(\mathbf{a}), (\mathbf{vw}_1)(\mathbf{a}), \dots, (\mathbf{vw})(\mathbf{a})\}, \\ &\quad \{\mathbf{a}, v_1(\mathbf{a}), \dots, ((v_1 \dots v_{n-1})(\mathbf{a}), \mathbf{v}(\mathbf{a})), \\ &\quad (\mathbf{v}(\mathbf{a}), (\mathbf{vw}_1)(\mathbf{a})), \dots, ((\mathbf{vw}_1 \dots w_{m-1})(\mathbf{a}), (\mathbf{vw})(\mathbf{a})), \\ &\quad ((\mathbf{vw})(\mathbf{a}), (\mathbf{vw}_1 \dots w_{m-1})(\mathbf{a})), \dots, ((\mathbf{vw}_1)(\mathbf{a}), \mathbf{v}(\mathbf{a})), \\ &\quad (\mathbf{v}(\mathbf{a}), (v_1 \dots v_{n-1})(\mathbf{a})), \dots, (v_1(\mathbf{a}), \mathbf{a})\}) \\ &= p^{\mathbf{a}}(\mathbf{vw}), \end{aligned}$$

womit die Folgerung bewiesen ist. ✱

Das Plotter-Bild $p^{\mathbf{a}}(\mathbf{w})$ zu einem Wort \mathbf{w} sieht so aus wie der gezeichnete Graph $sg^{\mathbf{a}}(\mathbf{w})$ ohne Spitzen, d. h. am Bild ist nicht zu erkennen, wie oft und in welcher Richtung der Stift zwei Rasterpunkte verbunden hat.

Einen Zusammenhang zwischen Kantenmengen und Bildern zweier Wörter stellt das folgende Lemma heraus:

Lemma 2.14 *Stimmen die Kantenmengen $\|\mathbf{a}\mathbf{v}$ und $\|\mathbf{a}\mathbf{w}$ zweier Wörter $\mathbf{v}, \mathbf{w} \in \mathcal{A}^*$ bezüglich eines Punktes $\mathbf{a} \in \mathbb{Z}^2$ überein, so stimmen auch die Bilder $p^{\mathbf{a}}(\mathbf{v})$ und $p^{\mathbf{a}}(\mathbf{w})$ überein.*

Beweis: Es seien \mathbf{a} ein Bezugspunkt aus \mathbb{Z}^2 und \mathbf{v}, \mathbf{w} zwei Wörter aus \mathcal{A}^* . Die Kantenmengen $\|\mathbf{a}\mathbf{v}, \|\mathbf{a}\mathbf{w}$ sind umkehrbar eindeutig den schlichten gerichteten Graphen $sg^{\mathbf{a}}(\mathbf{v}), sg^{\mathbf{a}}(\mathbf{w})$ zugeordnet; folglich gilt

$$\|\mathbf{a}\mathbf{v} = \|\mathbf{a}\mathbf{w} \iff sg^{\mathbf{a}}(\mathbf{v}) = sg^{\mathbf{a}}(\mathbf{w}).$$

Per Definition stimmen die Knotenmengen von $sg^{\mathbf{a}}(\mathbf{v})$ und $p^{\mathbf{a}}(\mathbf{v})$ bzw. $sg^{\mathbf{a}}(\mathbf{w})$ und $p^{\mathbf{a}}(\mathbf{w})$ überein. Es sei $a_{\mathbf{v}}^s(x, y), a_{\mathbf{w}}^s(x, y), a_{\mathbf{v}}^p(x, y)$ bzw. $a_{\mathbf{w}}^p(x, y)$ die Anzahl der Kanten von x nach y in den Graphen $sg^{\mathbf{a}}(\mathbf{v}), sg^{\mathbf{a}}(\mathbf{w}), p^{\mathbf{a}}(\mathbf{v})$ bzw. $p^{\mathbf{a}}(\mathbf{w})$. Stimmen $sg^{\mathbf{a}}(\mathbf{v})$ und $sg^{\mathbf{a}}(\mathbf{w})$ überein, so stimmen die Knotenmengen von $p^{\mathbf{a}}(\mathbf{v})$ und $p^{\mathbf{a}}(\mathbf{w})$ überein, und für die Kantenzahlen in den Bildern gilt

$$\begin{aligned} a_{\mathbf{v}}^p(x, y) &= \max \{ a_{\mathbf{v}}^s(x, y), a_{\mathbf{v}}^s(y, x) \} && \text{(Definition)} \\ &= \max \{ a_{\mathbf{w}}^s(x, y), a_{\mathbf{w}}^s(y, x) \} && (sg^{\mathbf{a}}(\mathbf{v}) = sg^{\mathbf{a}}(\mathbf{w})) \\ &= a_{\mathbf{w}}^p(x, y) && \text{(Definition)} \end{aligned}$$

für beliebige Knoten x, y ; d. h. $p^{\mathbf{a}}(\mathbf{v})$ und $p^{\mathbf{a}}(\mathbf{w})$ stimmen überein:

$$sg^{\mathbf{a}}(\mathbf{v}) = sg^{\mathbf{a}}(\mathbf{w}) \implies p^{\mathbf{a}}(\mathbf{v}) = p^{\mathbf{a}}(\mathbf{w}).$$

Damit gilt

$$\|\mathbf{a}\mathbf{v} = \|\mathbf{a}\mathbf{w} \implies p^{\mathbf{a}}(\mathbf{v}) = p^{\mathbf{a}}(\mathbf{w}),$$

was das Lemma gerade besagt. ✱

Die Knotenmenge der Graphen $g^{\mathbf{a}}(\mathbf{w}), sg^{\mathbf{a}}(\mathbf{w}), p^{\mathbf{a}}(\mathbf{w})$ zu einem Wort $\mathbf{w} \in \mathcal{A}^*$ und einem Bezugspunkt $\mathbf{a} \in \mathbb{Z}^2$ sei mit $\odot^{\mathbf{a}}(\mathbf{w})$ symbolisiert. Der Index kann entfallen, wenn sich die Graphen auf den Nullpunkt beziehen: $\odot^{\mathbf{0}} = \odot$.

Die Beispiele in Tabelle 2 sollen die Zusammenhänge veranschaulichen. Der umrandete Knoten markiert jeweils den Punkt $\mathbf{w}(\mathbf{a})$.

Die Mengen

- \mathcal{A}^* von Wörtern,
- $\mathcal{G} = \{ g^{\mathbf{a}}(\mathbf{w}) \mid \mathbf{w} \in \mathcal{A}^*, \mathbf{a} \in \mathbb{Z}^2 \}$ von gerichteten Graphen,
- $\mathcal{S} = \{ sg^{\mathbf{a}}(\mathbf{w}) \mid \mathbf{w} \in \mathcal{A}^*, \mathbf{a} \in \mathbb{Z}^2 \}$ von schlichten gerichteten Graphen und
- $\mathcal{P} = \{ p^{\mathbf{a}}(\mathbf{w}) \mid \mathbf{w} \in \mathcal{A}^*, \mathbf{a} \in \mathbb{Z}^2 \}$ von Bildern

w	λ	r	$rudl$	$rudlrud$
$g^a(w)$	\circ \mathfrak{a}	$\xrightarrow{\quad} \circ$ $\mathfrak{a} \quad \mathfrak{a}+(1,0)$	$\xrightarrow{\quad} \circ$ $\mathfrak{a} \quad \mathfrak{a}+(1,0)$ \uparrow $\mathfrak{a}+(1,1)$	$\xrightarrow{\quad} \circ$ $\mathfrak{a} \quad \mathfrak{a}+(1,0)$ \uparrow $\mathfrak{a}+(1,1)$
$sg^a(w)$	\circ \mathfrak{a}	$\xrightarrow{\quad} \circ$ $\mathfrak{a} \quad \mathfrak{a}+(1,0)$	$\xrightarrow{\quad} \circ$ $\mathfrak{a} \quad \mathfrak{a}+(1,0)$ \uparrow $\mathfrak{a}+(1,1)$	$\xrightarrow{\quad} \circ$ $\mathfrak{a} \quad \mathfrak{a}+(1,0)$ \uparrow $\mathfrak{a}+(1,1)$
$\dots\dots$	\dots	$\dots\dots$	$\dots\dots$	$\dots\dots$
$\ {}^a w$	\emptyset	$\{(\mathfrak{a}, r)\}$	$\{(\mathfrak{a}, r),$ $(\mathfrak{a} + (1, 0), u),$ $(\mathfrak{a} + (1, 1), d),$ $(\mathfrak{a} + (1, 0), l)\}$	$\{(\mathfrak{a}, r),$ $(\mathfrak{a} + (1, 0), u),$ $(\mathfrak{a} + (1, 1), d),$ $(\mathfrak{a} + (1, 0), l)\}$
$p^a(w)$	\circ \mathfrak{a}	$\xrightarrow{\quad} \circ$ $\mathfrak{a} \quad \mathfrak{a}+(1,0)$	$\xrightarrow{\quad} \circ$ $\mathfrak{a} \quad \mathfrak{a}+(1,0)$ \uparrow $\mathfrak{a}+(1,1)$	$\xrightarrow{\quad} \circ$ $\mathfrak{a} \quad \mathfrak{a}+(1,0)$ \uparrow $\mathfrak{a}+(1,1)$

Tabelle 2: Zusammenhang

bilden unterschiedliche Ebenen einer Abstrahierungshierarchie. Auf unterster Ebene werden Wörter einer Sprache betrachtet. Ein Interpretieren der Wörter als gerichtete Graphen mit Mehrfachkanten führt auf die nächsthöhere Stufe. Auf die dritte Ebene gelangt man durch Abstrahieren von den Mehrfachkanten, d. h. jedem gerichteten Graphen der zweiten Ebene ist sein schlichter gerichteter Graph auf der dritten zugeordnet. Durch Abstrahieren von den Kantenrichtungen gelangt man schließlich auf die Ebene der Bilder. Da jeder Abstraktionsschritt eine eindeutige Abbildung ist, gibt es zu jedem Wort der untersten Schicht einen eindeutigen „Interpretationsweg“ zu einem Bild auf der obersten Schicht. Das folgende Diagramm zeigt einen Ausschnitt der Hierarchie:

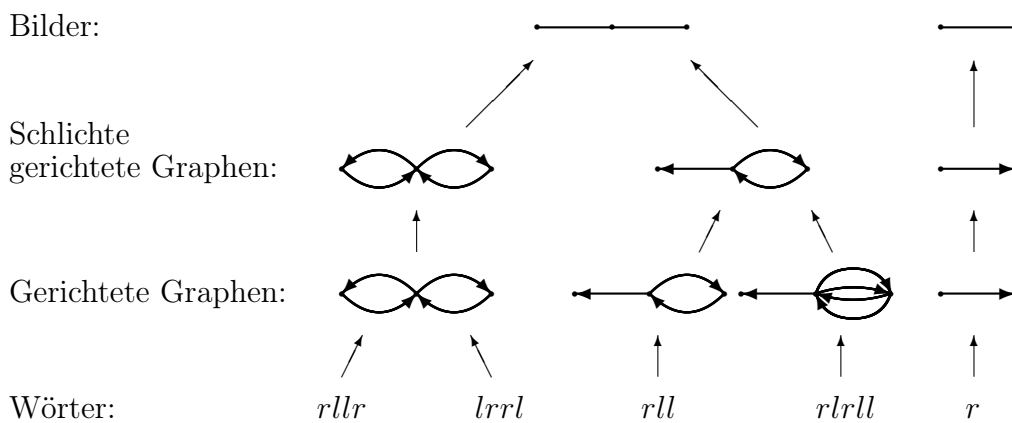


Abbildung 4: Abstrahierungshierarchie

2.3 Ketten-Code-Bild-Systeme

Allgemeine Definitionen von LINDENMAYER-Systemen sind in [RS80]; von Ketten-Code-Bild-Systemen in [DHr92] zu finden. Die vorliegende Arbeit betrachtet die spezielle Klasse *sDOL* der synchronen, deterministischen, kontextfreien Ketten-Code-Bild-Systeme.

Ein synchrones, deterministisches, kontextfreies Ketten-Code-Bildsystem (*sDOL*-System) ist ein 5-Tupel

$$G = (\mathcal{A}, h, \omega, k, m)$$

mit dem in Abschnitt 2.1 eingeführten Alphabet $\mathcal{A} = \{r, l, u, d\}$, einem Endomorphismus h auf dem Monoid (\mathcal{A}, \cdot) – s. Lemma 2.1 –, einem nichtleeren Startwort (Axiom) $\omega \in \mathcal{A}^+$ und zwei natürlichen Zahlen (Synchronisationsparametern) $k \in \mathbb{N}_0$ und $m \in \mathbb{N}_0$. Dabei unterliegt der Endomorphismus h folgenden Einschränkungen:

- Für alle Buchstaben $x \in \mathcal{A}$ gilt

$$(h(x))(\mathfrak{o}) = k\mathfrak{v}_x.$$

- Für alle Buchstaben $x \in \mathcal{A}$ und alle Punkte $\mathfrak{q}_x \in \odot(h(x))$ des Bildes zu $h(x)$ gilt

$$\begin{aligned} (0, -m) &\leq \mathfrak{q}_x \leq (k, m), & \text{falls } x = r, \\ (-k, -m) &\leq \mathfrak{q}_x \leq (0, m), & \text{falls } x = l, \\ (-m, 0) &\leq \mathfrak{q}_x \leq (m, k), & \text{falls } x = u, \\ (-m, -k) &\leq \mathfrak{q}_x \leq (m, 0), & \text{falls } x = d, \end{aligned}$$

wobei die Relation \leq komponentenweise gemeint ist:

$$(a, b) \leq (c, d) \iff a \leq c \wedge b \leq d.$$

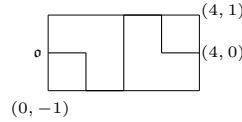
Der Synchronisationsbegriff soll an folgendem Beispiel erläutert werden:

Es sei $G = (\mathcal{A}, h, ruld, 4, 1)$ ein *sDOL*-System mit

$$\begin{aligned} h(r) &= rdruurdr, \\ h(u) &= urulluru, \\ h(l) &= lulddlul, \\ h(d) &= dlldrrld. \end{aligned}$$

Die Bilder vom Axiom *ruld* und dessen erster Ableitung sind in Abschnitt 1, Abbildung 2 dargestellt. Die Synchronisationsbedingungen seien an $x = r$ erklärt:

Das Bild zu r ist $\circ \xrightarrow{(1,0)}$ mit dem Endpunkt $(1,0)$, der eine Einheit nach rechts vom Nullpunkt entfernt liegt. Die erste Synchronisationsbedingung gibt an, wo der Endpunkt des Bildes von $h(r)$ liegt: k Einheiten nach rechts vom Nullpunkt entfernt; in diesem Falle bei $4v_r = (4,0)$. Die zweite Bedingung bewirkt, daß das Bild zu $h(r)$ in einem gewissen Rechteck liegt; in diesem Falle zwischen den Eckpunkten $(0,-1)$ und $(4,1)$:



Die Bedingungen gelten analog auch für die anderen Buchstaben. Damit ist eine „gleichmäßige Entwicklung“ der Bilder vorgeschrieben (die Kanten wachsen „synchron“).

Die n -stellige Verkettung des Endomorphismus h wird kurz h^n geschrieben:

$$h^n = \underbrace{h \circ \dots \circ h}_n.$$

Angewendet auf ein Wort $w \in \mathcal{A}^*$, liefert sie die n -te Ableitung von w ; geschrieben $w^{(n)}$ (w', w'', w''' für die ersten drei Ableitungen, $w^{(0)} = w$). Der Ableitung eines Wortes entspricht eine Ableitung der zugeordneten Graphen: Es sei $w \in \mathcal{A}^*$. Dem Wort w ist ein schlichter gerichteter Graph $sg(w)$ zugeordnet. Diesen skaliere man um den Faktor k , d. h. man bilde jeden Knoten $\mathfrak{q} \in \odot(w)$ auf den Knoten $k\mathfrak{q}$ ab und verbinde zwei Knoten $k\mathfrak{p}$ und $k\mathfrak{q}$ genau dann durch eine Kante, wenn in $sg(w)$ eine Kante von \mathfrak{p} nach \mathfrak{q} führt; eine Kante sei außerdem mit x bezeichnet, wenn $x(\mathfrak{p}) = \mathfrak{q}$ ist. Nun ersetze man jede x -Kante im skalierten Graphen durch den schlichten Graphen von $h(x)$. Der so entstehende Graph ist der schlichte gerichtete Graph zur Ableitung $h(w)$.

Dies sei am vorhergehenden Beispielsystem demonstriert:

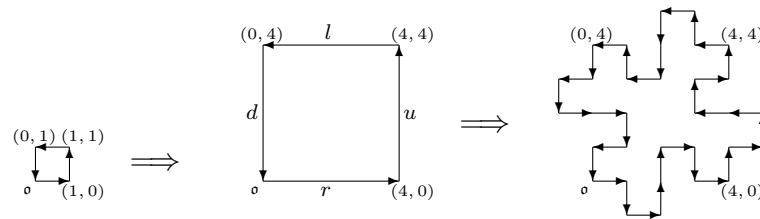


Abbildung 5: Ableitung eines Graphen

Die von einem $sDOL$ -System G erzeugte Bildsprache $B(G)$ ist die Menge aller Bilder von Ableitungen des Axioms ω :

$$B(G) = \{ p(\omega^{(n)}) \mid n \in \mathbb{N}_0 \}.$$

Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein $sDOL$ -System. Aus der ersten Synchronisationsbedingung ergibt sich

Folgerung 2.15 Für die n -te Ableitung ($n \in \mathbb{N}$) eines Buchstaben $x \in \mathcal{A}$ gilt

$$x^{(n)}(\mathfrak{o}) = k^n \mathfrak{v}_x.$$

Beweis: Per Definition ist für jeden Buchstaben $x \in \mathcal{A}$

$$x(\mathfrak{o}) = \mathfrak{v}_x \quad \text{und} \quad x'(\mathfrak{o}) = k \mathfrak{v}_x$$

festgelegt. Für jeden Buchstaben $x \in \mathcal{A}$ gelte $x^{(i)}(\mathfrak{o}) = k^i \mathfrak{v}_x$ für alle Ableitungen höchstens n -ter Stufe. Die n -te Ableitung eines Buchstaben x sei

$$x^{(n)} = x_1 \dots x_l \quad (x_j \in \mathcal{A}, j = 1, \dots, l).$$

Für die $(n+1)$ -te Ableitung gilt damit

$$\begin{aligned} x^{(n+1)}(\mathfrak{o}) &= (h(x^{(n)}))(\mathfrak{o}) \\ &= (h(x_1) \dots h(x_l))(\mathfrak{o}) \quad (\text{Operationstreu von } h) \\ &= (x'_1 \dots x'_l)(\mathfrak{o}) \\ &= \sum_{i=1}^l x'_i(\mathfrak{o}) \quad (\text{Folgerung 2.7}) \\ &= \sum_{i=1}^l k \mathfrak{v}_{x_i} \quad (1. \text{ Synchronisationsbedingung}) \\ &= k \sum_{i=1}^l \mathfrak{v}_{x_i} \\ &= k \sum_{i=1}^l x_i(\mathfrak{o}) \quad (\text{Definition}) \\ &= k(x_1 \dots x_l)(\mathfrak{o}) \quad (\text{Folgerung 2.7}) \\ &= kx^{(n)}(\mathfrak{o}) \\ &= k(k^n \mathfrak{v}_x) \quad (\text{Induktionsvoraussetzung}) \\ &= k^{n+1} \mathfrak{v}_x. \end{aligned}$$

Das bedeutet: Es gilt

$$x^{(n)}(\mathfrak{o}) = k^n \mathfrak{v}_x$$

für alle Buchstaben $x \in \mathcal{A}$ und Ableitungsstufen $n \in \mathbb{N}$. ✱

Daraus ergibt sich

Folgerung 2.16 Jede Ableitung eines Buchstaben $x \in \mathcal{A}$ ist genau dann mit der identischen Abbildung λ assoziiert, wenn $k = 0$ gilt:

$$(\forall \mathfrak{q} \in \mathbb{Z}^2 : x^{(n)}(\mathfrak{q}) = \mathfrak{q}) \iff k = 0 \quad (x \in \mathcal{A}, n \in \mathbb{N}).$$

Beweis: Nach Folgerung 2.7 gilt $x^{(n)}(\mathfrak{q}) = x^{(n)}(\mathfrak{o}) + \mathfrak{q}$, woraus

$$(\forall \mathfrak{q} \in \mathbb{Z}^2 : x^{(n)}(\mathfrak{q}) = \mathfrak{q}) \iff x^{(n)}(\mathfrak{o}) = \mathfrak{o}$$

folgt. Wenn $k = 0$, so ist auch $k^n = 0$ (für $n \in \mathbb{N}$). Nach Folgerung 2.15 gilt damit $x^{(n)}(\mathfrak{o}) = \mathfrak{o}$. Wenn $k \neq 0$, so ist auch $k^n \neq 0$ (für $n \in \mathbb{N}$). Da $\mathfrak{v}_x \neq \mathfrak{o}$ (für $x \in \mathcal{A}$), ist auch $x^{(n)}(\mathfrak{o}) \neq \mathfrak{o}$. Das bedeutet

$$k = 0 \iff x^{(n)}(\mathfrak{o}) = \mathfrak{o}.$$

Insgesamt gilt damit auch die Behauptung. ✱

Aus Folgerung 2.3 zur Entstehung von Buchstaben-Mengen schließt man für die Ableitung eines Wortes \mathfrak{w}

Folgerung 2.17 *Die Buchstaben-Menge $[\mathfrak{w}']$ der Ableitung eines Wortes $\mathfrak{w} \in \mathcal{A}^*$ entsteht durch Vereinigung der Buchstaben-Mengen $[x']$ der Ableitungen jener $x \in \mathcal{A}$, die im Wort $\mathfrak{w} \in \mathcal{A}^*$ auftreten:*

$$[\mathfrak{w}'] = \bigcup_{x \in [\mathfrak{w}]} [x'], \quad \mathfrak{w} \in \mathcal{A}^*.$$

Beweis: Es sei \mathfrak{w} ein Wort aus \mathcal{A}_n^* . Dann ist die Ableitung von \mathfrak{w} :

$$\mathfrak{w}' = w'_1 \dots w'_n.$$

Nach Folgerung 2.3 gilt für die Buchstaben-Menge

$$[\mathfrak{w}'] = \bigcup_{i=1}^n [w'_i].$$

Stimmen zwei Buchstaben w_i und w_j aus dem Wort \mathfrak{w} überein, so stimmen auch ihre Ableitungen überein (auf Grund der Determiniertheit des *sDOL*-Systems G) und damit auch deren Buchstabenmengen. Folglich gilt

$$[\mathfrak{w}'] = \bigcup_{x \in [\mathfrak{w}]} [x']. \quad \text{✱}$$

Diese Folgerung stellt einen Spezialfall des nachstehenden, verallgemeinerten Lemmas dar.

Lemma 2.18 *Die Buchstabenmenge der n -ten Ableitung eines Wortes $\mathfrak{w} \in \mathcal{A}^*$ ist die Vereinigung der Buchstaben-Mengen der i -ten Ableitungen ($i \leq n$) jener $x \in \mathcal{A}$, die in der $(n - i)$ -ten Ableitung von \mathfrak{w} auftreten:*

$$[\mathfrak{w}^{(n)}] = \bigcup_{x \in [\mathfrak{w}^{(n-i)}]} [x^{(i)}], \quad \mathfrak{w} \in \mathcal{A}^*, \quad n, i \in \mathbb{N}_0, \quad i \leq n.$$

Beweis: Es sei $w^{(n-i)} \in \mathcal{A}_l^*$ die $(n-i)$ -te Ableitung eines Wortes $w \in \mathcal{A}^*$ ($1 \leq i \leq n$). Dann lautet die n -te Ableitung von w

$$w^{(n)} = w_1^{(i)} \dots w_l^{(i)},$$

und für ihre Buchstaben-Menge gilt

$$\begin{aligned} [w^{(n)}] &= \bigcup_{j=1}^l [w_j^{(i)}] && \text{(Folgerung 2.3)} \\ &= \bigcup_{x \in [w^{(n-i)}]} [x^{(i)}] && \text{(wenn } \hat{x} = \check{x}, \text{ so auch } \hat{x}^{(i)} = \check{x}^{(i)} \text{ und } [\hat{x}^{(i)}] = [\check{x}^{(i)}]). \end{aligned}$$

✱

Mit diesem Ergebnis kann folgende Aussage über die Stabilität von Buchstaben-Mengen getroffen werden:

Lemma 2.19 *Wenn sich bei der Ableitung eines Wortes $w \in \mathcal{A}^*$ die Buchstaben-Menge nicht ändert, so ändert sie sich bei keiner weiteren Ableitung:*

$$[w] = [w'] \implies [w] = [w^{(n)}], \quad w \in \mathcal{A}^*, n \in \mathbb{N}.$$

Beweis: Es mögen die Buchstaben-Mengen $[w]$ und $[w^{(i)}]$ für alle i , $1 \leq i \leq n$, übereinstimmen. Dann gilt

$$\begin{aligned} [w^{(n+1)}] &= \bigcup_{x \in [w^{(n)}]} [x'] && \text{(Lemma 2.18)} \\ &= \bigcup_{x \in [w]} [x'] && \text{(Induktionsvoraussetzung)} \\ &= [w'] && \text{(Folgerung 2.17)} \\ &= [w] && \text{(Induktionsvoraussetzung),} \end{aligned}$$

d. h. aus $[w] = [w']$ folgt $[w] = [w^{(n)}]$ für alle $n \in \mathbb{N}$.

✱

Damit ergibt sich

Lemma 2.20 *Das Leerwort λ stimmt mit seinen Ableitungen überein:*

$$\lambda = \lambda^{(n)}, \quad n \in \mathbb{N}.$$

Beweis: Nach Folgerung 2.17 gilt

$$[\lambda'] = \bigcup_{x \in [\lambda]} [x'] = \emptyset,$$

d. h. $\lambda' = \lambda$, woraus $\lambda^{(n)} = \lambda$ für $n \in \mathbb{N}$ folgt (Lemma 2.19).

✱

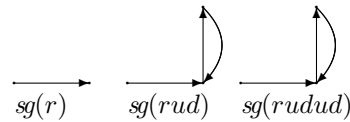
3 Endlichkeitsuntersuchungen

Bei vielen Aufgaben, in denen Ketten-Code-Bild-Systeme eine Rolle spielen, muß zunächst geklärt werden, ob ein gegebenes *sDOL*-System eine endliche oder unendliche Bildmenge erzeugt oder wie ein *sDOL*-System beschaffen sein muß, damit die erzeugte Bildmenge gewisse Eigenschaften besitzt. In [DHR92] sind Bedingungen angegeben, unter denen ein *sTOL*-System eine endliche bzw. unendliche Bildsprache erzeugt (Satz 4). Aus dem zweiten Fall

„If $k_1 = k_2 = \dots = k_r = 1$ and $d_i \geq 1$ for some i , $1 \leq i \leq r$, then we obtain the infinity of $B(G)$ by an analogous argument.“

im Beweis von Satz 4 ([DHR92]) kann man nicht auf die Unendlichkeit der erzeugten Bildsprache schließen; dies zeigt das folgende Beispiel:

Es sei $G = (\mathcal{A}, h, r, 1, 1)$ ein *sDOL*-System mit $h(r) = rud$ und $h(x) = x$ für $x \in \mathcal{A} \setminus \{r\}$. Die Klasse *sTOL* schließt die Klasse *sDOL* ein (s. [RS80]). Damit ist G auch ein *sTOL*-System; der obige Fall trifft zu wegen $k = 1$ und $m = 1$ (der Parameter m entspricht dem Parameter d in der Notation von [DHR92]). Die schlichten gerichteten Graphen $sg(r)$, $sg(h(r))$, $sg(h^2(r))$ sind (ohne Knotenbezeichnungen):



Die Kantenmengen der 1. und 2. Ableitung stimmen überein. Mit Lemma 3.9 der vorliegenden Arbeit wird gezeigt, daß auch die Kantenmengen höherer Ableitungen mit ihnen übereinstimmen. Nach Lemma 2.14 heißt dies, daß ab der 2. Ableitung kein neues Bild entsteht. Folglich ist die erzeugte Bildmenge endlich.

Außerdem wird im Beweis von Satz 4 in [DHR92] nichts über *sTOL*-Systeme mit $k_i = 0$ für ein oder alle i ausgesagt. Da Null als Wert für ein d_i zulässig ist (wird im angesprochenen Beweis benutzt) und d_i laut Definition in der gleichen Menge wie k_i liegt, muß Null auch für die Parameter k_i zulässig sein. Das bedeutet, daß der angegebene Algorithmus nicht zu jedem System die Endlichkeit bzw. Unendlichkeit der Bildsprache feststellt.

Die vorliegende Arbeit leitet Bedingungen her, anhand derer entschieden werden kann, ob ein *sDOL*-System eine endliche Bildmenge erzeugt oder nicht.

Wie in Abschnitt 2.3 vorgestellt, ist der Synchronisationsparameter k ein „Wachstumsfaktor“. Daher ist es naheliegend, die Fälle $k < 1$, $k = 1$ und $k > 1$ gesondert zu betrachten.

3.1 sDOL-Systeme mit $k < 1$

Laut Definition in Abschnitt 2.3 ist k eine natürliche Zahl. Daher gibt es für $k < 1$ nur eine Möglichkeit: $k = 0$. Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein sDOL-System mit $k = 0$. Das System G ist synchron; daraus ergibt sich

Folgerung 3.1 *Für alle $x \in \mathcal{A}$ gilt*

1. $x'(\mathbf{o}) = \mathbf{o}$,
2. $\#_x x' = \#_{\bar{x}} x' = 0$,
3. $\#_{x^\perp} x' = \#_{\bar{x}^\perp} x'$.

Beweis:

1. Die erste Synchronisationsbedingung (Abschnitt 2.3) lautet

$$x'(\mathbf{o}) = k\mathbf{v}_x.$$

Wegen $k = 0$ ist demzufolge

$$x'(\mathbf{o}) = \mathbf{o}.$$

2. Die zweite Synchronisationsbedingung besagt, daß in den Graphen zu r' und l' die erste Koordinate jedes Knotens und in den Graphen zu u' und d' die zweite Koordinate jedes Knotens Null ist. Angenommen, es tritt x oder \bar{x} im Wort x' auf; dieser Buchstabe sei y . Dann enthält die Knotenmenge $\odot(x')$ zwei Knoten \mathbf{q} und $\tilde{\mathbf{q}} = \mathbf{q} + \mathbf{v}_y$. Da $\mathbf{v}_y \neq \mathbf{o}$ ist, hat einer der beiden Knoten $\mathbf{q}, \tilde{\mathbf{q}}$ eine von Null verschiedene Komponente (im Falle $x \in \{r, l\}$ ist es die erste, sonst die zweite). Dies ist ein Widerspruch zur Synchronisationsbedingung; daher treten im Wort x' weder x noch \bar{x} auf.

3. Es sei $x' = x_1 \dots x_n$ ($x_i \in \mathcal{A}, i = 1, \dots, n$). Aus Folgerung 2.7 ergibt sich

$$x'(\mathbf{o}) = \mathbf{v}_{x_1} + \dots + \mathbf{v}_{x_n}.$$

Angenommen, die Aussage gilt nicht; d. h.

$$\#_{x^\perp} x' \neq \#_{\bar{x}^\perp} x'.$$

Dann gilt mit 2.

$$\begin{aligned} x'(\mathbf{o}) &= \#_{x^\perp} x' \mathbf{v}_{x^\perp} + \#_{\bar{x}^\perp} x' \mathbf{v}_{\bar{x}^\perp} \\ &= \#_{x^\perp} x' \mathbf{v}_{x^\perp} + \#_{\bar{x}^\perp} x' (-\mathbf{v}_{x^\perp}) \quad (\text{Lemma 2.8}) \\ &= (\#_{x^\perp} x' - \#_{\bar{x}^\perp} x') \mathbf{v}_{x^\perp} \\ &\neq \mathbf{o}, \end{aligned}$$

was einen Widerspruch zu 1. darstellt. Demzufolge treten x^\perp und \bar{x}^\perp gleich oft auf. ✱

Für ein beliebiges Wort $w \in \mathcal{A}^*$ tritt genau einer der drei Fälle ein:

1. $w' = \lambda$. Nach Lemma 2.20 sind auch alle weiteren Ableitungen leer:

$$w^{(n)} = \lambda \text{ für } n \geq 1.$$

2. $w' \neq \lambda$, $w'' = \lambda$. Nach Lemma 2.20 sind alle weiteren Ableitungen leer:

$$w^{(n)} = \lambda \text{ für } n \geq 2.$$

3. $w' \neq \lambda$, $w'' \neq \lambda$. Aus Folgerung 3.1 ergibt sich für die Buchstaben-Mengen der ersten beiden Ableitungen:

$$[w'] \in \{ \{r, l\}, \{u, d\}, \{r, l, u, d\} \}$$

und

$$[w''] = \begin{cases} \{u, d\}, & \text{falls } [w'] = \{r, l\} \\ \{r, l\}, & \text{falls } [w'] = \{u, d\} \\ \mathcal{A}, & \text{falls } [w'] = \mathcal{A}. \end{cases}$$

Daraus folgt mit $[w'''] = \bigcup_{x \in [w'']} [x']$ (aus Lemma 2.18):

$$[w'''] = \begin{cases} \{r, l\}, & \text{falls } [w'] = \{r, l\} \\ \{u, d\}, & \text{falls } [w'] = \{u, d\} \\ \mathcal{A}, & \text{falls } [w'] = \mathcal{A} \end{cases},$$

d. h. die Buchstaben-Menge der dritten Ableitung stimmt mit der Buchstaben-Menge der ersten überein: $[w'''] = [w']$. Daraus folgt, die Buchstaben-Menge der vierten stimmt mit jener der zweiten Ableitung überein:

$$[w^{(4)}] = \bigcup_{x \in [w''']} [x'] = \bigcup_{x \in [w'']} [x'] = [w''].$$

Es mögen jeweils die Buchstaben-Mengen ungerader und die Buchstaben-Mengen gerader Ableitungen übereinstimmen: Es gelte $[w^{(2i-1)}] = [w']$ und $[w^{(2i)}] = [w'']$ für $2 \leq i \leq n$. Dann gilt

1. für die ungeraden Ableitungen

$$\begin{aligned} [w^{(2n+1)}] &= \bigcup_{x \in [w^{(2n)}]} [x'] \quad (\text{Lemma 2.18}) \\ &= \bigcup_{x \in [w'']} [x'] \quad (\text{Induktionsvoraussetzung}) \\ &= [w'''] \quad (\text{Lemma 2.18}) \\ &= [w'] \quad (\text{Induktionsvoraussetzung}) \end{aligned}$$

und

2. für die geraden Ableitungen

$$\begin{aligned}
[\mathbf{w}^{(2n+2)}] &= \bigcup_{x \in [\mathbf{w}^{(2n+1)}]} [x'] \quad (\text{Lemma 2.18}) \\
&= \bigcup_{x \in [\mathbf{w}']} [x'] \quad (1.) \\
&= [\mathbf{w}'''] \quad (\text{Lemma 2.18}).
\end{aligned}$$

Insgesamt bedeutet dies, wenn die Buchstaben-Menge der zweiten Ableitung eines Wortes nicht leer ist, dann stimmen alle Buchstaben-Mengen ungerader Ableitung ab der ersten und alle Buchstaben-Mengen gerader Ableitung ab der zweiten überein:

Lemma 3.2 *Die Buchstaben-Mengen ab der 2. Ableitung eines Wortes $\mathbf{w} \in \mathcal{A}^*$ sind entweder leer oder alternieren ab der 1. Ableitung:*

$$[\mathbf{w}'] = \emptyset \implies [\mathbf{w}^{(n)}] = \emptyset \quad (n \geq 2),$$

$$[\mathbf{w}'] \neq \emptyset \implies [\mathbf{w}^{(2n-1)}] = [\mathbf{w}'] \wedge [\mathbf{w}^{(2n)}] = [\mathbf{w}'''] \quad (n \geq 2).$$

Ein ähnlicher Zusammenhang läßt sich auch für Kantenmengen finden. Dazu sei $\mathbf{w}^{(n)}$ die n -te Ableitung eines Wortes $\mathbf{w} \in \mathcal{A}^*$:

$$\mathbf{w}^{(n)} = x_1 \dots x_l \quad (x_i \in \mathcal{A}, i = 1, \dots, l).$$

Für die Kantenmenge von $\mathbf{w}^{(n+1)}$ gilt nach Folgerung 2.12:

$$\|\mathbf{w}^{(n+1)}\| = \|x'_1 \cup \|x'_1(\mathfrak{o})x'_2 \cup \dots \cup \|(x'_1 \dots x'_{l-1})(\mathfrak{o})x'_l\|.$$

Aus den Folgerungen 2.7 und 3.1 ergibt sich

$$\begin{aligned}
(x'_1 \dots x'_i)(\mathfrak{o}) &= x'_1(\mathfrak{o}) + \dots + x'_i(\mathfrak{o}) \quad (\text{Folgerung 2.7}) \\
&= \mathfrak{o} \quad (\text{Folgerung 3.1})
\end{aligned}$$

für $i = 1, \dots, l$, d. h.

$$\|\mathbf{w}^{(n+1)}\| = \bigcup_{i=1}^l \|x'_i\| = \bigcup_{x \in [\mathbf{w}^{(n)}]} \|x'_i\| \quad (\text{wenn } \hat{x} = \check{x}, \text{ so auch } \hat{x}' = \check{x}' \text{ und } \|\hat{x}'\| = \|\check{x}'\|).$$

Mit Lemma 3.2 folgt daraus: Die Kantenmengen ab der 2. Ableitung eines Wortes $\mathbf{w} \in \mathcal{A}^*$ sind entweder leer oder alternieren ab der 2. Ableitung:

Wenn $\mathbf{w}'' = \lambda$, so ist $\|\mathbf{w}^{(n)}\| = \emptyset$ für $n \geq 2$; sonst gilt $\|\mathbf{w}^{(2n)}\| = \|\mathbf{w}''\|$ und $\|\mathbf{w}^{(2n+1)}\| = \|\mathbf{w}'''\|$ für $n \geq 2$ (die Kantenmengen gerader Ableitung stimmen ab der zweiten überein, die ungerader Ableitung ab der dritten).

Nach Lemma 2.14 kann man auf die zugeordneten Bilder schließen:

Lemma 3.3 *Die Bilder ab der 2. Ableitung eines Wortes $w \in \mathcal{A}^*$ bestehen entweder nur aus dem Nullpunkt oder alternieren:*

$$w'' = \lambda \implies p(w^{(n)}) = (\{\mathfrak{o}\}) \quad (n \geq 2),$$

$$w'' \neq \lambda \implies p(w^{(2n)}) = p(w'') \wedge p(w^{(2n+1)}) = p(w''') \quad (n \geq 2).$$

Mit $w = \omega$ folgt daraus

Satz 1 *Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein sDOL-System mit $k = 0$. Dann gilt für die erzeugte Bildsprache*

$$B(G) = \{ p(\omega), p(\omega'), p(\omega''), p(\omega''') \}.$$

3.2 sDOL-Systeme mit $k > 1$

Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein sDOL-System mit $k > 1$.

Nach den Folgerungen 2.15 und 2.16 gilt

$$x^{(n)}(\mathfrak{o}) = k^n \mathfrak{v}_x \neq \mathfrak{o}, \quad n \in \mathbb{N}_0$$

für alle Buchstaben $x \in \mathcal{A}$ und Ableitungsstufen $n \in \mathbb{N}_0$.

Es sei $w = xv$ ($x \in \mathcal{A}$, $v \in \mathcal{A}^*$) ein Wort mit dem Anfangsbuchstaben x . Für die n -te Ableitung gilt $w^{(n)} = x^{(n)}v^{(n)}$, und $x^{(n)}(\mathfrak{o})$ ist ein Knoten der Graphen zu $w^{(n)}$:

$$x^{(n)}(\mathfrak{o}) \in \odot(w^{(n)}), \quad n \in \mathbb{N}_0.$$

Desweiteren seien X_w die Menge aller Punkte $x^{(n)}(\mathfrak{o})$, $n \in \mathbb{N}_0$:

$$X_w = \{ x^{(n)}(\mathfrak{o}) \mid n \in \mathbb{N}_0 \} = \{ k^n \mathfrak{v}_x \mid n \in \mathbb{N}_0 \},$$

P_w die Vereinigung der Punktfolgen $\odot(w^{(n)})$, $n \in \mathbb{N}_0$:

$$P_w = \bigcup_{n \in \mathbb{N}_0} \odot(w^{(n)})$$

und B_w die Menge aller Bilder $p(w^{(n)})$, $n \in \mathbb{N}_0$:

$$B_w = \bigcup_{n \in \mathbb{N}_0} \{ p(w^{(n)}) \}.$$

Die Menge X_w ist unendlich,

$$|X_w| = \infty,$$

denn es gilt $k > 1$ und $\mathbf{v}_x \neq \mathbf{o}$ für alle Buchstaben $x \in \mathcal{A}$. Jeder Punkt in $X_{\mathbf{w}}$ tritt auch in $P_{\mathbf{w}}$ auf:

$$X_{\mathbf{w}} \subseteq P_{\mathbf{w}};$$

daher ist auch die Menge $P_{\mathbf{w}}$ unendlich:

$$|P_{\mathbf{w}}| = \infty.$$

Jede Punktmenge $\odot(\mathbf{w}^{(n)})$, $n \in \mathbb{N}_0$, ist endlich; daher sind unter den Mengen $\odot(\mathbf{w}^{(n)})$ unendlich viele verschiedene. Sind die Knotenmengen zweier Wörter $\mathbf{u}, \mathbf{v} \in \mathcal{A}^*$ unterschiedlich

$$\odot(\mathbf{u}) \neq \odot(\mathbf{v}),$$

so sind auch die Bilder verschieden

$$p(\mathbf{u}) \neq p(\mathbf{v}).$$

Daher gibt es unter den Bildern $p(\mathbf{w}^{(n)})$, $n \in \mathbb{N}_0$, unendlich viele verschiedene, d. h. die Bildmenge ist unendlich:

$$|B_{\mathbf{w}}| = \infty.$$

Im Falle $\mathbf{w} = \omega$ entspricht die Bildmenge B_{ω} der erzeugten Bildsprache $B(G)$. Daraus folgt

Satz 2 *Zu jedem sDOL-System $G = (\mathcal{A}, h, \omega, k, m)$ mit $k > 1$ ist die erzeugte Bildsprache $B(G)$ unendlich:*

$$|B(G)| = \infty.$$

3.3 sDOL-Systeme mit $k = 1$

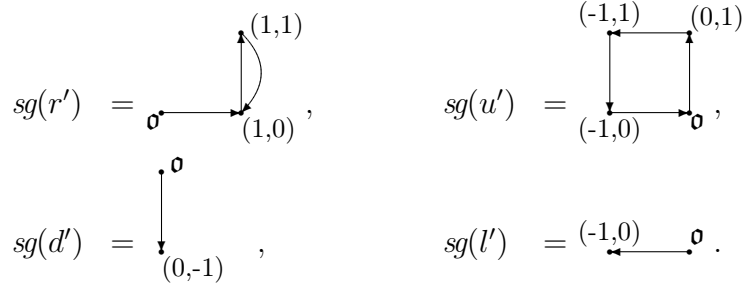
Im Gegensatz zu den vorherigen Situationen gibt es zu dem Synchronisationsparameter $k = 1$ sowohl sDOL-Systeme mit endlicher als auch solche mit unendlicher Bildmenge.

Dazu jeweils ein Beispiel:

1. Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein sDOL-System mit $\omega = r$, $k = 1$, $m = 1$ und

$$\begin{aligned} h(r) &= rud, \\ h(u) &= uldru, \\ h(d) &= d, \\ h(l) &= l. \end{aligned}$$

Die schlichten gerichteten Graphen zu den Ableitungen von $x \in \mathcal{A}$ sind:



An der Darstellung ist zu erkennen, daß jeder schlichte gerichtete Graph $sg(x)$ zu $x \in \mathcal{A}$ Untergraph von $sg(x')$ ist:

$$\|x\| \subseteq \|x'\|.$$

Damit ist auch jeder Graph $sg^a(x)$ Untergraph von $sg^a(x')$:

$$\|{}^a x\| \subseteq \|{}^a x'\|, \quad \mathbf{a} \in \mathbb{Z}^2.$$

Das *sDOL*-System G ist propagierend, d. h. bei keiner Ableitung entsteht das Leerwort:

$$x' \neq \lambda, \quad \text{für alle } x \in \mathcal{A},$$

bzw. in der Wortlänge ausgedrückt:

$$\#x' \geq 1, \quad \text{für alle } x \in \mathcal{A}.$$

In jedem abgeleiteten Wort tritt ein r auf und die Ableitung von r ist länger als r ($\#r' > \#r$). Folglich ist jedes n -stufig ($n \in \mathbb{N}$) aus $\omega = r$ abgeleitete Wort länger als jede Ableitung geringerer Stufe:

$$\forall n \in \mathbb{N} \forall i, 0 \leq i < n : \#\omega^{(i)} < \#\omega^{(n)}.$$

Daher erzeugt G eine unendliche Wortmenge mit den ersten Ableitungen

$$\begin{array}{l}
 \omega = r \\
 \omega' = rud \\
 \omega'' = ruduldrud \\
 \omega''' = ruduldruduldruldruduldrud \\
 \omega^{(4)} = ruduldruduldruldruduldruduldruldruduldruduldruduldruduldrud \\
 \vdots
 \end{array}$$

Die erzeugte Bildmenge ist jedoch endlich, denn die Bilder ab der 2. Ableitung stimmen überein (die Knotenbezeichnungen wurden zur besseren Übersichtlichkeit weggelassen):

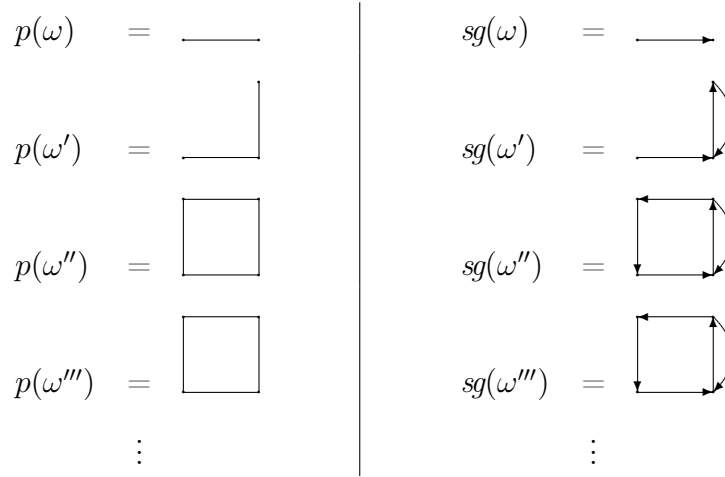


Tabelle 3: Ableitung – Beispiel 1

Zur Veranschaulichung sind die schlichten gerichteten Graphen der ersten Ableitungen von ω angegeben. Die Kantenmengen von ω'' und ω''' stimmen überein, damit stimmen auch alle Kantenmengen höherer Ableitung überein (wird in Lemma 3.9 bewiesen) und daher stimmen, nach Lemma 2.14, die Bilder ab der 2. Ableitung von ω überein.

Das bedeutet: G erzeugt genau drei unterschiedliche Bilder (also eine endliche Menge):

$$B(G) = \{ p(\omega), p(\omega'), p(\omega'') \}.$$

2. Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein $sDOL$ -System mit $\omega = r$, $k = 1$, $m = 1$ und

$$\begin{aligned} h(r) &= rud, \\ h(u) &= uldru, \\ h(d) &= d, \\ h(l) &= lrl. \end{aligned}$$

Dieses Beispiel unterscheidet sich von dem vorherigen nur in $h(l)$. Der schlichte gerichtete Graph zu l' ist

$$sg(l') = \overset{(-1,0)}{\curvearrowright} \circ.$$

Dieser Unterschied reicht aus, um aus einem $sDOL$ -System mit endlicher Bildsprache eines mit unendlicher Bildmenge zu konstruieren. Die folgende Tabelle zeigt die schlichten gerichteten Graphen und Bilder der ersten Ableitungen

vom Startwort ω :

n	0	1	2	3	4	5	6	...	
$sg(\omega^{(n)})$...
$p(\omega^{(n)})$...	

Tabelle 4: Ableitung – Beispiel 2

In der Kantenmenge von ω gibt es nur eine Kante; sie entsteht durch r :

$$\|\omega = \{ (\mathbf{o}, r) \}.$$

In der Kantenmenge von ω''' gibt es bereits zwei von r erzeugte Kanten

$$\{ (\mathbf{o}, r), ((0, 1), r) \} \subseteq \|\omega''',$$

d. h. aus der Kante (\mathbf{o}, r) entstehen in drei Ableitungsschritten die Kanten (\mathbf{o}, r) und $(\mathbf{o} + (0, 1), r)$. So entstehen aus der Kante $((0, 1), r)$ in weiteren drei Ableitungsschritten die Kanten $((0, 1), r)$ und $((0, 1) + (0, 1), r)$ usw. Für die Knotenmengen bedeutet dies

$$\begin{aligned} \{ \mathbf{o} \} &\subseteq \odot(\omega), & (0, 1) &\notin \odot(\omega), \\ \{ \mathbf{o}, (0, 1) \} &\subseteq \odot(\omega'''), & (0, 2) &\notin \odot(\omega'''), \\ \{ \mathbf{o}, (0, 1), (0, 2) \} &\subseteq \odot(\omega^{(6)}), & (0, 3) &\notin \odot(\omega^{(6)}), \dots \end{aligned}$$

d. h. die Bilder werden in jedem 3. Schritt größer (in dem Sinne, daß die Knotenanzahl zunimmt). Da diese „Ausdehnung“ unbeschränkt ist, erzeugt das $sDOL$ -System G eine unendliche Bildmenge

$$|B(G)| = \infty.$$

Die Beispiele zeigen, daß im Falle $k = 1$ nicht am Parameter k entschieden werden kann, ob ein $sDOL$ -System eine endliche Bildsprache generiert oder nicht.

Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein $sDOL$ -System mit $k = 1$.

Folgerung 3.4 *Aus der Synchronisiertheit von G folgt für alle $x \in \mathcal{A}$:*

1. $x^{(n)}(\mathbf{o}) = x(\mathbf{o})$ für alle $n \in \mathbb{N}_0$,
2. $\#_x x' = \#_{\bar{x}} x' + 1$ und $\#_{x^\perp} x' = \#_{\bar{x}^\perp} x'$.

Beweis:

1. Aus der ersten Synchronisationsbedingung (s. Abschnitt 2.3) ergibt sich mit Folgerung 2.15:

$$x^{(n)}(\mathfrak{o}) = \mathfrak{v}_x = x(\mathfrak{o}), \quad n \in \mathbb{N}_0.$$

2. Es sei $x' = x_1 \dots x_n$ ($x_i \in \mathcal{A}, i = 1, \dots, n$). Nach Folgerung 2.7 gilt

$$x'(\mathfrak{o}) = \mathfrak{v}_{x_1} + \dots + \mathfrak{v}_{x_n}.$$

Angenommen, die Behauptung gilt nicht; d. h.

$$\#_x x' - \#_{\bar{x}} x' \neq 1 \quad \text{oder} \quad \#_{x^\perp} x' - \#_{\bar{x}^\perp} x' \neq 0.$$

Dann gilt

$$\begin{aligned} x'(\mathfrak{o}) &= (\#_x x') \mathfrak{v}_x + (\#_{\bar{x}} x') \mathfrak{v}_{\bar{x}} + (\#_{x^\perp} x') \mathfrak{v}_{x^\perp} + (\#_{\bar{x}^\perp} x') \mathfrak{v}_{\bar{x}^\perp} \\ &= (\#_x x') \mathfrak{v}_x - (\#_{\bar{x}} x') \mathfrak{v}_x + (\#_{x^\perp} x') \mathfrak{v}_{x^\perp} - (\#_{\bar{x}^\perp} x') \mathfrak{v}_{x^\perp} \quad (\text{Lemma 2.8}) \\ &= (\#_x x' - \#_{\bar{x}} x') \mathfrak{v}_x + (\#_{x^\perp} x' - \#_{\bar{x}^\perp} x') \mathfrak{v}_{x^\perp} \\ &\neq \mathfrak{v}_x, \end{aligned}$$

was einen Widerspruch zu 1. darstellt. Demzufolge treten in x' die Buchstaben x einmal öfter als \bar{x} , sowie x^\perp und \bar{x}^\perp gleich oft auf. ✱

Daraus ergibt sich

Folgerung 3.5

1. Es sei \mathfrak{w} ein Wort aus \mathcal{A}_i^* . Dann gilt für eine beliebige Ableitungsstufe $n \in \mathbb{N}_0$:

$$\begin{aligned} \mathfrak{w}^{(n)}(\mathfrak{o}) &= (w_1^{(n)} \dots w_l^{(n)})(\mathfrak{o}) \\ &= w_1^{(n)}(\mathfrak{o}) + \dots + w_l^{(n)}(\mathfrak{o}) \quad (\text{Folgerung 2.7}) \\ &= w_1(\mathfrak{o}) + \dots + w_l(\mathfrak{o}) \quad (\text{Folgerung 3.4}) \\ &= (w_1 \dots w_l)(\mathfrak{o}) \quad (\text{Folgerung 2.7}) \\ &= \mathfrak{w}(\mathfrak{o}). \end{aligned}$$

2. Nach Folgerung 3.4 (2.) tritt ein Buchstabe x in seiner Ableitung x' mindestens einmal auf:

$$\#_x x' \geq 1 \quad \text{bzw.} \quad x \in [x'] \quad \text{für alle } x \in \mathcal{A}.$$

Auf Wörter erweitert, liefert der zweite Teil von Folgerung 3.5:

Folgerung 3.6 *Alle Buchstaben, die in einem Wort $w \in \mathcal{A}^*$ auftreten, kommen auch in der Ableitung von w vor:*

$$[w] \subseteq [w'].$$

Beweis: Mit Lemma 2.18 und Folgerung 3.5 gilt

$$[w] = \bigcup_{x \in [w]} \{x\} \subseteq \bigcup_{x \in [w]} [x'] = [w'].$$

✱

Daraus folgt sofort: Wenn $[w] = \mathcal{A}$, so gilt auch $[w^{(n)}] = \mathcal{A}$ mit $n \in \mathbb{N}$.

Die Buchstaben-Mengen nehmen nicht beliebig lange zu; ab der 2. Ableitung eines Wortes ändern sie sich nicht mehr. Diese Aussage wird nicht für beliebige Wörter sondern nur für die atomaren benötigt:

Lemma 3.7 *Für jeden Buchstaben $x \in \mathcal{A}$ stimmen die Buchstaben-Mengen ab der 2. Ableitung überein:*

$$[x''] = [x^{(n)}], \quad n \geq 2.$$

Beweis: Aus der Synchronisiertheit von G ergeben sich folgende Fälle:

1. $[x'] = \{x\}$, d. h. $[x] = [x']$. Mit Lemma 2.19 folgt $[x] = [x^{(n)}] = \{x\}$ für alle $n \in \mathbb{N}$.
2. $[x'] = \{x, \bar{x}\}$. Nach Lemma 2.18 gilt $[x''] = [x'] \cup [\bar{x}']$. Das bedeutet, daß entweder die Buchstaben x^\perp und \bar{x}^\perp nicht in \bar{x}' auftreten oder gemeinsam vorkommen (Folgerung 3.4). Im ersten Falle ist $[x''] = [x']$; im zweiten $[x''] = \mathcal{A}$:

$$[x''] = \begin{cases} [x'], & \text{falls } x^\perp \notin [\bar{x}'], \\ \mathcal{A}, & \text{sonst.} \end{cases}$$

Mit Lemma 2.19 und Folgerung 3.6 ergibt sich $[x^{(n)}] = [x'']$ für $n \geq 2$.

3. $[x'] = \{x, x^\perp, \bar{x}^\perp\}$. Nach Lemma 2.18 ist $[x''] = [x'] \cup [x^{\perp'}] \cup [\bar{x}^{\perp'}]$. Daher gilt

$$[x''] = \begin{cases} [x'], & \text{falls } \bar{x} \notin [x^{\perp'}] \cup [\bar{x}^{\perp'}], \\ \mathcal{A}, & \text{sonst.} \end{cases}$$

Mit Lemma 2.19 und Folgerung 3.6 ergibt sich $[x^{(n)}] = [x'']$ für $n \geq 2$.

4. $[x'] = \mathcal{A}$. Nach Folgerung 3.6 gilt $[x^{(n)}] = \mathcal{A}$ für alle $n \in \mathbb{N}$.

Insgesamt gilt damit für jeden Buchstaben $x \in \mathcal{A}$

$$[x''] = [x^{(n)}], \quad n \geq 2.$$

✱

Die Kantenmenge $\|\mathbf{w}$ eines Wortes $\mathbf{w} \in \mathcal{A}^*$ läßt sich in disjunkte x -Kantenmengen

$$\|_x \mathbf{w} = \{ (\mathbf{q}, x) \mid (\mathbf{q}, x) \in \|\mathbf{w} \}$$

der Buchstaben von \mathbf{w} zerlegen:

$$\|\mathbf{w} = \|_r \mathbf{w} \cup \|_l \mathbf{w} \cup \|_u \mathbf{w} \cup \|_d \mathbf{w}, \quad \text{und} \quad \|_x \mathbf{w} \cap \|_y \mathbf{w} = \emptyset, \quad \text{falls } x \neq y.$$

Aus Folgerung 2.12 zur Vereinigung von Kantenmengen schließt man für die Ableitung eines Wortes \mathbf{w}

Lemma 3.8 *Die Kantenmenge $\|\mathbf{w}^{(n)}$ der n -ten Ableitung eines Wortes $\mathbf{w} \in \mathcal{A}^*$ ist die Vereinigung der Kantenmengen $\|^{q,x^{(i)}}$, wobei die Kanten (\mathbf{q}, x) Elemente der Kantenmenge $\|\mathbf{w}^{(n-i)}$ der $(n-i)$ -ten Ableitung des Wortes \mathbf{w} sind:*

$$\|\mathbf{w}^{(n)} = \bigcup_{(\mathbf{q}, x) \in \|\mathbf{w}^{(n-i)}} \|^{q,x^{(i)}}, \quad \mathbf{w} \in \mathcal{A}^*, \quad n, i \in \mathbb{N}_0, \quad i \leq n.$$

Beweis: Es sei $\mathbf{w}^{(n-i)} \in \mathcal{A}_l^*$ die $(n-i)$ -te Ableitung eines Wortes $\mathbf{w} \in \mathcal{A}^*$ ($1 \leq i \leq n$). Dann lautet die n -te Ableitung von \mathbf{w}

$$\mathbf{w}^{(n)} = w_1^{(i)} \dots w_l^{(i)}.$$

Für die Kantenmengen von $\mathbf{w}^{(n-i)}$ und $\mathbf{w}^{(n)}$ gilt nach Folgerung 2.12:

$$\begin{aligned} \|\mathbf{w}^{(n-i)} &= \|w_1 \cup \|^{w_1(\mathbf{o})} w_2 \cup \dots \cup \|^{(w_1 \dots w_{l-1})(\mathbf{o})} w_l \\ &= \{ (\mathbf{o}, w_1), (w_1(\mathbf{o}), w_2), \dots, ((w_1 \dots w_{l-1})(\mathbf{o}), w_l) \}, \end{aligned}$$

$$\begin{aligned} \|\mathbf{w}^{(n)} &= \|w_1^{(i)} \cup \|^{w_1^{(i)}(\mathbf{o})} w_2^{(i)} \cup \dots \cup \|^{(w_1^{(i)} \dots w_{l-1}^{(i)})(\mathbf{o})} w_l^{(i)} \\ &= \|w_1^{(i)} \cup \|^{w_1(\mathbf{o})} w_2^{(i)} \cup \dots \cup \|^{(w_1 \dots w_{l-1})(\mathbf{o})} w_l^{(i)} \quad (\text{Folgerung 3.5 (1.)}) \\ &= \bigcup_{(\mathbf{q}, x) \in \|\mathbf{w}^{(n-i)}} \|^{q,x^{(i)}}, \end{aligned}$$

womit das Lemma bewiesen ist.

✱

Lemma 3.9 *Stimmt die Kantenmenge eines Wortes $\mathbf{w} \in \mathcal{A}^*$ mit der seiner Ableitung \mathbf{w}' überein, so stimmt sie auch mit der Kantenmenge jeder höheren Ableitung überein:*

$$\|\mathbf{w} = \|\mathbf{w}' \implies \|\mathbf{w} = \|\mathbf{w}^{(n)}, \quad \mathbf{w} \in \mathcal{A}^*, \quad n \in \mathbb{N}.$$

Beweis: Es mögen die Kantenmengen bis zu einer gewissen Ableitung übereinstimmen: $\|\mathbf{w} = \|\mathbf{w}^{(i)}$ für $1 \leq i \leq n$. Dann gilt

$$\begin{aligned} \|\mathbf{w}^{(n+1)} &= \bigcup_{(\mathbf{q},x) \in \|\mathbf{w}^{(n)}} \|\mathbf{q}x' && \text{(Lemma 3.8)} \\ &= \bigcup_{(\mathbf{q},x) \in \|\mathbf{w}} \|\mathbf{q}x' && \text{(Induktionsvoraussetzung)} \\ &= \|\mathbf{w}' && \text{(Lemma 3.8)} \\ &= \|\mathbf{w} && \text{(Induktionsvoraussetzung),} \end{aligned}$$

d. h. wenn $\|\mathbf{w}'$ mit $\|\mathbf{w}$ übereinstimmt, so stimmen auch die Kantenmengen $\|\mathbf{w}^{(n)}$ für alle $n \in \mathbb{N}$ mit $\|\mathbf{w}$ überein. ✱

Mit $\mathbf{w} = \omega''$ und Lemma 2.14 erhält man

Satz 3 *Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein sDOL-System mit $k = 1$. Wenn die Kantenmengen der 2. und 3. Ableitung des Axioms ω übereinstimmen, so enthält die erzeugte Bildmenge höchstens drei Elemente:*

$$\|\omega'' = \|\omega''' \implies B(G) = \{ p(\omega), p(\omega'), p(\omega'') \}.$$

Lemma 3.10 *Es sei \mathbf{w} ein Wort aus \mathcal{A}^* . Wenn die Kantenmengen von \mathbf{w}'' und \mathbf{w}''' nicht übereinstimmen, so gibt es einen Buchstaben $x \in [\mathbf{w}'']$ und eine der ersten drei Ableitungskantenmengen $\|_x x'$, $\|_x x''$ oder $\|_x x'''$ enthält eine von (\mathfrak{o}, x) verschiedene Kante:*

$$\|\mathbf{w}'' \neq \|\mathbf{w}''' \implies (\exists x \in [\mathbf{w}'''] : \|_x x \neq \|_x x' \vee \|_x x \neq \|_x x'' \vee \|_x x \neq \|_x x''').$$

Beweis: Die Aussage des Lemmas ist äquivalent zur Aussage:

Es sei \mathbf{w} ein Wort aus \mathcal{A}^* . Wenn zu jedem Buchstaben $x \in [\mathbf{w}'']$ die x -Kantenmengen der ersten drei Ableitungen von x ausschließlich die Kante (\mathfrak{o}, x) enthalten, so stimmen die Kantenmengen von \mathbf{w}'' und \mathbf{w}''' überein:

$$(\forall x \in [\mathbf{w}'''] : \|_x x = \|_x x' \wedge \|_x x = \|_x x'' \wedge \|_x x = \|_x x''') \implies \|\mathbf{w}'' = \|\mathbf{w}'''.$$

Diese Aussage wird im folgenden bewiesen. Es sei $\mathbf{w} \in \mathcal{A}^*$ und es gelte für alle Buchstaben $x \in [\mathbf{w}'']$

$$\|_x x = \|_x x' = \|_x x'' = \|_x x''.$$

Per Definition gilt

$$\begin{aligned} \|x &= \|_x x, \\ \|x' &= \|_x x' \cup \|_{\bar{x}} x' \cup \|_{x^\perp} x' \cup \|_{\bar{x}^\perp} x', \\ \|x'' &= \|_x x'' \cup \|_{\bar{x}} x'' \cup \|_{x^\perp} x'' \cup \|_{\bar{x}^\perp} x'', \\ \|x''' &= \|_x x''' \cup \|_{\bar{x}} x''' \cup \|_{x^\perp} x''' \cup \|_{\bar{x}^\perp} x'''. \end{aligned}$$

Jede y -Kante in $\|_y x'$ ist auch Element von $\|_y x''$: Es sei (\mathfrak{q}, y) eine Kante aus $\|_y x'$:

$$(\mathfrak{q}, y) \in \|_y x'.$$

Da die Menge $\|_y x'$ Untermenge von $\|x'$ ist, liegt die Kante (\mathfrak{q}, y) auch in $\|x'$:

$$(\mathfrak{q}, y) \in \|x'.$$

Nach Lemma 3.8 gilt damit

$$\|_y^{\mathfrak{q}} x' \subseteq \|x'',$$

woraus

$$\|_y^{\mathfrak{q}} x' \subseteq \|_y x''$$

folgt, da alle Kanten aus $\|_y^{\mathfrak{q}} x'$ in $\|_y^{\mathfrak{q}} x'$ und damit auch in $\|x''$ auftreten; jedoch nur unter den y -Kanten. Wegen $\|_y y = \|_y y'$ gilt $\|_y^{\mathfrak{q}} y = \|_y^{\mathfrak{q}} y'$ und damit

$$\|_y^{\mathfrak{q}} y \subseteq \|_y x'',$$

was gleichbedeutend ist mit $(\mathfrak{q}, y) \in \|_y x''$. Somit gilt für alle Buchstaben y :

$$\|_y x' \subseteq \|_y x''.$$

Analog erhält man die Beziehung

$$\|_y x'' \subseteq \|_y x''',$$

womit sich

$$\|x \subseteq \|x' \subseteq \|x'' \subseteq \|x'''$$

ergibt. Nachfolgend wird gezeigt, daß die Menge $\|x'''$ in der Menge $\|x''$ eingeschlossen ist, woraus die Gleichheit $\|x'' = \|x'''$ folgt.

Die folgende Fallunterscheidung ist auf den Beweis von Lemma 3.7 zurückzuführen:

1. $[x'] = \{x\}$, d. h. $[x] = [x']$. Mit Lemma 2.19 folgt $[x^{(n)}] = [x] = \{x\}$ für alle Ableitungsstufen $n \in \mathbb{N}_0$, d. h. $x^{(n)} = x$. Insbesondere gilt $x'' = x'''$; damit stimmen auch die Kantenmengen überein:

$$\|x'' = \|x'''.$$

2. $[x'] = \{x, \bar{x}\}$, $[x''] = [x']$. Nach Lemma 2.19 gilt $[x^{(n)}] = [x''] = \{x, \bar{x}\}$ für alle Ableitungsstufen $n \geq 2$. Auf Grund der zweiten Synchronisationsbedingung wechseln sich die Buchstaben in den Wörtern ab: $x'' = x\bar{x}x\{\bar{x}x\}^*$, $x''' = x\bar{x}x\{\bar{x}x\}^*$. Damit gilt für die Kantenmengen

$$\begin{aligned} \|x'' &= \|x \cup \|x^{(0)} \bar{x} \cup \|x \cup \|x^{(0)} \bar{x} \cup \dots \cup \|x && \text{(Folgerung 2.12)} \\ &= \|x'''. \end{aligned}$$

3. $[x'] = \{ x, \bar{x} \}$, $[x''] = \mathcal{A}$. Für die Kantenmenge $\|x'$ gilt

$$\begin{aligned} \|x' &= \|x \cup \|^{x(\mathfrak{o})}\bar{x} \cup \|x \cup \|^{x(\mathfrak{o})}\bar{x} \cup \dots \cup \|x && \text{(siehe 2.)} \\ &= \{ (\mathfrak{o}, x), (x(\mathfrak{o}), \bar{x}) \}. \end{aligned}$$

In $\|x''$ tritt keine andere x -Kante auf, da dies die Bedingung $\|_x x = \|_x x''$ verletzen würde. Da es in $\|x'$ nur eine x -Kante gibt, liefert kein Buchstabe x im Wort x' beim Ableiten eine neue \bar{x} -Kante. Auch durch \bar{x} im Wort x' entstehen keine neuen \bar{x} -Kanten in $\|x''$ (wegen $\|_{\bar{x}} \bar{x} = \|_{\bar{x}} \bar{x}'$). Somit treten die x^\perp - und \bar{x}^\perp -Kanten in $\|x''$ paarweise auf:

$$(\mathfrak{q}, x^\perp) \in \|x'' \iff (x^\perp(\mathfrak{q}), \bar{x}^\perp) \in \|x''.$$

In $\|x'''$ tritt keine andere x -Kante auf als (\mathfrak{o}, x) (dies wäre eine Verletzung von $\|_x x = \|_x x'''$) und keine andere \bar{x} -Kante als $(x(\mathfrak{o}), \bar{x})$ (dies wäre eine Verletzung von $\|_{\bar{x}} \bar{x} = \|_{\bar{x}} \bar{x}'''$). Da weder aus x^\perp noch \bar{x}^\perp neue x - oder \bar{x} -Kanten entstehen, treten alle x^\perp - und \bar{x}^\perp -Kanten in $\|x'''$ paarweise auf. Aus x^\perp entstehen jedoch keine neuen x^\perp -Kanten; somit können auch keine neuen \bar{x}^\perp -Kanten in $\|x'''$ auftreten. Analoges gilt für \bar{x}^\perp . Daher gibt es keine Kante in $\|x'''$, die nicht in $\|x''$ auftritt:

$$\|x''' \subseteq \|x''.$$

4. $[x'] = \{ x, x^\perp, \bar{x}^\perp \}$, $[x''] = [x']$. In diesem Falle enthalten die Buchstaben-Mengen $[x^\perp]$ und $[\bar{x}^\perp]$ nicht den Buchstaben \bar{x} und nach Folgerung 3.4 (2.) auch nicht den Buchstaben x . In der Menge $\|x'$ treten x -, x^\perp - und \bar{x}^\perp -Kanten auf. Wegen $\|_x x = \|_x x'$ gibt es in $\|x'$ nur eine x -Kante: (\mathfrak{o}, x) . Da es keine \bar{x} -Kante in $\|x'$ gibt, treten die x^\perp - und \bar{x}^\perp -Kanten paarweise auf:

$$(\mathfrak{q}, x^\perp) \in \|x' \iff (x^\perp(\mathfrak{q}), \bar{x}^\perp) \in \|x'.$$

Ein Buchstabe x^\perp im Wort x' liefert beim Ableiten höchstens die Kanten $(\mathfrak{q}, x^\perp), (x^\perp(\mathfrak{q}), \bar{x}^\perp)$, wobei (\mathfrak{q}, x^\perp) in $\|x'$ liegt. Analog gilt dies für \bar{x}^\perp . Damit liegen in $\|x''$ alle und nur die Kanten, die auch Elemente von $\|x'$ sind:





$$\|x' = \|x''.$$

Mit Lemma 3.9 folgt daraus

$$\|x'' = \|x'''.$$

5. $[x'] = \{ x, x^\perp, \bar{x}^\perp \}$, $[x''] = \mathcal{A}$. Das bedeutet: \bar{x} tritt in der Ableitung von x^\perp oder \bar{x}^\perp auf. Beim Ableiten von x' entsteht aus x^\perp oder \bar{x}^\perp eine \bar{x} -Kante in $\|x''$; wegen der Synchronisiertheit entsteht mit jeder \bar{x} -Kante auch eine x -Kante

(Folgerung 3.4). Da mit der 2. Ableitung von x aber keine anderen x -Kanten als (\mathfrak{o}, x) entstehen (wegen $\|_x x = \|_x x''$), kann es in der Menge $\|x'$ höchstens jeweils eine x^\perp - und \bar{x}^\perp -Kante geben. Angenommen, es gäbe zwei. Dann sind ihre Positionen bezüglich der x -Kante verschieden. Demzufolge befinden sich die beim Ableiten entstehenden x -Kanten an unterschiedlichen Positionen, was jedoch ein Widerspruch ist. Damit trifft auf die Kantenmenge von x' einer der folgenden vier Fälle zu (rechts daneben stehen zur Veranschaulichung die Graphen für $x = r$):

- (a) $\|x' = \{ (\mathfrak{o}, x^\perp), (x^\perp(\mathfrak{o}), \bar{x}^\perp), (\mathfrak{o}, x) \}$ 
- (b) $\|x' = \{ (\mathfrak{o}, x), (x(\mathfrak{o}), x^\perp), ((xx^\perp)(\mathfrak{o}), \bar{x}^\perp) \}$ 
- (c) $\|x' = \{ (\mathfrak{o}, \bar{x}^\perp), (\bar{x}^\perp(\mathfrak{o}), x^\perp), (\mathfrak{o}, x) \}$ 
- (d) $\|x' = \{ (\mathfrak{o}, x), (x(\mathfrak{o}), \bar{x}^\perp), ((x\bar{x}^\perp)(\mathfrak{o}), x^\perp) \}$ 





Da keine neue x -Kante in $\|x''$ entsteht, x^\perp keine neue x^\perp -Kante und \bar{x}^\perp keine neue \bar{x}^\perp -Kante in $\|x''$ liefern, gibt es in der Menge $\|x''$ genau zwei Möglichkeiten für eine \bar{x} -Kante:

- (*) $(x(\mathfrak{o}), \bar{x}) \in \|x''$ oder
- (*) $((xx^\perp)(\mathfrak{o}), \bar{x}) \in \|x''$ in den obigen Fällen (a) und (b) oder
- (‡) $((x\bar{x}^\perp)(\mathfrak{o}), \bar{x}) \in \|x''$ in den obigen Fällen (c) und (d).

In den beiden Fällen (*) und (‡) entsteht ein neuer Knoten \mathfrak{e} :

- (a) $\mathfrak{e} = (xx^\perp)(\mathfrak{o})$ 
- (b) $\mathfrak{e} = x^\perp(\mathfrak{o})$ 
- (c) $\mathfrak{e} = (x\bar{x}^\perp)(\mathfrak{o})$ 
- (d) $\mathfrak{e} = \bar{x}^\perp(\mathfrak{o})$ 

Da \mathfrak{e} weder der Startknoten \mathfrak{o} noch der Endknoten $x(\mathfrak{o})$ des Graphen ist, muß sowohl eine Kante zu \mathfrak{e} hinführen, als auch eine Kante von \mathfrak{e} wegführen. Somit gibt es in der Menge $\|x''$ unter den Bedingungen (*) und (‡) eine weitere Kante, die nicht in $\|x'$ auftritt:

- (a) $(x(\mathfrak{o}), x^\perp) \in \|x''$ 
- (b) $(x^\perp(\mathfrak{o}), \bar{x}^\perp) \in \|x''$ 
- (c) $(x(\mathfrak{o}), \bar{x}^\perp) \in \|x''$ 
- (d) $(\bar{x}^\perp(\mathfrak{o}), x^\perp) \in \|x''$ 

Daraus folgt, daß die \bar{x} -Kante in den Fällen (a) und (d) nur durch Ableiten von \bar{x}^\perp , in den Fällen (b) und (c) nur durch x^\perp entsteht, d. h. für (a) und (d) gilt $\bar{x} \notin [x^\perp]$, sonst ist $\bar{x} \notin [\bar{x}^\perp]$. Laut Voraussetzung entstehen beim Ableiten von x'' aus \bar{x} weder neue x - noch neue \bar{x} -Kanten. In den Fällen (a) und (d) entstehen

aus \bar{x} keine neuen \bar{x}^\perp -Kanten. Mit jeder neuen x^\perp -Kante würde auch eine neue \bar{x}^\perp -Kante entstehen, folglich entsteht aus \bar{x} auch keine neue x^\perp -Kante. Analog dazu entstehen aus \bar{x} in den Fällen (b) und (c) keine neuen x^\perp -Kanten und auch keine neuen \bar{x}^\perp -Kanten. In den Fällen (a) und (d) entstehen beim Ableiten aus x^\perp weder x - noch \bar{x} -Kanten (es gibt in $\|x''$ zwei x^\perp -Kanten; so würden unterschiedliche x - bzw. \bar{x} -Kanten entstehen) noch neue \bar{x}^\perp -Kanten (aus einer neuen \bar{x}^\perp -Kante würde beim nächsten Ableitungsschritt eine neue x^\perp -Kante entstehen, was ein Widerspruch zur Bedingung $\|_{x^\perp} x^\perp = \|_{x^\perp} x^{\perp\perp}$ wäre). Somit kommen auch keine neuen x^\perp -Kanten hinzu. Analog entstehen auch in den Fällen (b) und (c) aus \bar{x}^\perp keine neuen Kanten. Daher liegt jede Kante aus $\|x'''$ auch in $\|x''$, d. h.

$$\|x''' \subseteq \|x''.$$

6. $[x'] = \mathcal{A}$. Laut Voraussetzung entsteht beim Ableiten von x keine andere x -Kante als (\circ, x) . Es seien v, w, y verschiedene Buchstaben aus $\mathcal{A} \setminus \{x\}$. Beim Ableiten von x' entstehen aus v keine neuen x - oder v -Kanten in $\|x''$. Möglicherweise entsteht eine neue w -Kante. Das w in x'' erzeugt beim Ableiten keine neuen x -, v - oder w -Kanten. Ähnlich zu Fall 5 sind unter diesen Bedingungen aber auch keine neuen y -Kanten möglich. Das bedeutet, daß in $\|x'''$ keine Kanten liegen, die nicht auch Elemente von $\|x''$ sind:

$$\|x''' \subseteq \|x''.$$

Andere Fälle gibt es nicht (s. Lemma 3.7). Damit ist gezeigt, daß

$$\|x''' \subseteq \|x''$$

gilt. Mit der eingangs gezeigten Beziehung $\|x'' \subseteq \|x'''$ folgt die Übereinstimmung

$$\|x'' = \|x'''.$$

Damit stimmen auch die Kantenmengen bezüglich eines beliebigen Punktes $\mathbf{a} \in \mathbb{Z}^2$ überein:

$$\|^\mathbf{a} x'' = \|^\mathbf{a} x'''$$

Es habe \mathbf{w} die Länge n : $\mathbf{w} \in \mathcal{A}_n^*$. Für die Kantenmenge der zweiten Ableitung von \mathbf{w} ergibt sich aus den obigen Überlegungen

$$\begin{aligned} \|\mathbf{w}'' &= \|w_1'' \cup \|^{w_1''(\circ)} w_2'' \cup \dots \cup \|^{(w_1'' \dots w_{n-1}'')(\circ)} w_n'' && \text{(Folgerung 2.12)} \\ &= \|w_1'' \cup \|^{w_1'''(\circ)} w_2'' \cup \dots \cup \|^{(w_1''' \dots w_{n-1}''')(\circ)} w_n'' && \text{(Folgerung 3.5)} \\ &= \|w_1''' \cup \|^{w_1'''(\circ)} w_2''' \cup \dots \cup \|^{(w_1''' \dots w_{n-1}''')(\circ)} w_n''' && (\|^\mathbf{a} x'' = \|^\mathbf{a} x''') \\ &= \|\mathbf{w}''' && \text{(Folgerung 2.12),} \end{aligned}$$

woraus das Lemma folgt. ✱

Im folgenden sei $w \in \mathcal{A}^*$ ein nichtleeres Wort, so daß

$$\|_x x \neq \|_x x^{(l)}$$

für einen Buchstaben $x \in [w']$ und eine Ableitungsstufe $l \in \{1, 2, 3\}$ gilt. Die Kantenmenge $\|x$ besteht aus der Kante von \mathfrak{o} nach $x(\mathfrak{o})$:

$$\|x = \{ (\mathfrak{o}, x) \}.$$

Auf Grund der gestellten Bedingung gibt es im Graphen der l -ten Ableitung von x eine andere x -Kante:

$$(\mathfrak{q}, x) \in \|x^{(l)} \quad \text{mit } \mathfrak{q} \neq \mathfrak{o}.$$

Nach Lemma 3.8 gilt

$$\|x^{(2l)} = \bigcup_{(\mathfrak{r}, y) \in \|x^{(l)}} \|\mathfrak{r}y^{(l)}.$$

Daraus folgt

$$\|\mathfrak{q}x^{(l)} \subseteq \|x^{(2l)},$$

was bedeutet, daß alle Kanten zur l -ten Ableitung um den Punkt \mathfrak{q} verschoben in der $2l$ -ten Ableitung auftreten. Damit gilt insbesondere

$$(\mathfrak{q} + \mathfrak{q}, x) \in \|x^{(2l)}.$$

Durch Induktionsschluß erhält man allgemein

$$(n\mathfrak{q}, x) \in \|x^{(nl)}.$$

Die Behauptung gelte für $i \leq n$: $(i\mathfrak{q}, x) \in \|x^{(il)}$. Dann gilt für die $((n+1)l)$ -te Ableitung nach Lemma 3.8

$$\|x^{((n+1)l)} = \bigcup_{(\mathfrak{r}, y) \in \|x^{(nl)}} \|\mathfrak{r}y^{(l)}.$$

Auf Grund der Induktionsvoraussetzung ist $(n\mathfrak{q}, x) \in \|x^{(nl)}$. Damit gilt

$$\|n\mathfrak{q}x^{(l)} \subseteq \|x^{((n+1)l)}.$$

Es liegt die Kante (\mathfrak{q}, x) in der Menge $\|x^{(l)}$; daher liegt die Kante $(\mathfrak{q} + n\mathfrak{q}, x)$ in der Menge $\|n\mathfrak{q}x^{(l)}$ und damit auch in der Menge $\|x^{((n+1)l)}$, d. h.

$$((n+1)\mathfrak{q}, x) \in \|x^{((n+1)l)}.$$

Das bedeutet: Es gilt

$$(n\mathbf{q}, x) \in \|x^{(nl)}\|$$

für alle natürlichen Zahlen $n \in \mathbb{N}_0$.

Mit $\mathbf{w}'' = vx\tilde{\mathbf{v}}$ gilt für die Kantenmengen jeder l -ten Ableitung von \mathbf{w}''

$$\begin{aligned} \|\mathbf{w}^{(nl+2)}\| &= \|\mathbf{v}^{(nl)}\| \cup \|\mathbf{v}^{(nl)(\mathfrak{o})}x^{(nl)}\| \cup \|(\mathbf{vx})^{(nl)(\mathfrak{o})}\tilde{\mathbf{v}}^{(nl)}\| \\ &= \|\mathbf{v}^{(nl)}\| \cup \|\mathbf{v}^{(\mathfrak{o})}x^{(nl)}\| \cup \|(\mathbf{vx})^{(nl)(\mathfrak{o})}\tilde{\mathbf{v}}^{(nl)}\| \quad (\text{Folgerung 3.5}). \end{aligned}$$

Es gibt einen von \mathfrak{o} verschiedenen Punkt $\mathbf{q} \in \mathbb{Z}^2$, so daß in jeder Menge $\|\mathbf{v}^{(\mathfrak{o})}x^{(nl)}\|$, $n \in \mathbb{N}_0$, die Kante $(n\mathbf{q} + \mathbf{v}(\mathfrak{o}), x)$ auftritt. Damit ist für jedes $n \in \mathbb{N}_0$ die Kante $(n\mathbf{q} + \mathbf{v}(\mathfrak{o}), x)$ auch Element der Menge $\|\mathbf{w}^{(nl+2)}\|$ und der Knoten $n\mathbf{q} + \mathbf{v}(\mathfrak{o})$ Element der Knotenmenge $\odot(\mathbf{w}^{(nl+2)})$:

$$\exists \mathbf{q} \in \mathbb{Z}^2 \forall n \in \mathbb{N}_0 : n\mathbf{q} + \mathbf{v}(\mathfrak{o}) \in \odot(\mathbf{w}^{(nl+2)}).$$

Solch ein Punkt \mathbf{q} sei \mathbf{q}^* . Es seien $Q_{\mathbf{w}}$ die Menge aller Knoten $n\mathbf{q}^* + \mathbf{v}(\mathfrak{o})$:

$$Q_{\mathbf{w}} = \{ n\mathbf{q}^* + \mathbf{v}(\mathfrak{o}) \mid n \in \mathbb{N}_0 \},$$

$P_{\mathbf{w}}$ die Vereinigung der Punktmenge $\odot(\mathbf{w}^{(nl+2)})$ über alle $n \in \mathbb{N}_0$:

$$P_{\mathbf{w}} = \bigcup_{n \in \mathbb{N}_0} \odot(\mathbf{w}^{(nl+2)})$$

und $B_{\mathbf{w}}$ die Menge aller Bilder $p(\mathbf{w}^{(nl+2)})$ über alle $n \in \mathbb{N}_0$:

$$B_{\mathbf{w}} = \bigcup_{n \in \mathbb{N}_0} \{p(\mathbf{w}^{(nl+2)})\}.$$

Die Menge $Q_{\mathbf{w}}$ ist unendlich, da $\mathbf{q}^* \neq \mathfrak{o}$:

$$|Q_{\mathbf{w}}| = \infty.$$

Jeder Punkt aus $Q_{\mathbf{w}}$ tritt auch in $P_{\mathbf{w}}$ auf:

$$Q_{\mathbf{w}} \subseteq P_{\mathbf{w}};$$

damit ist auch $P_{\mathbf{w}}$ unendlich

$$|P_{\mathbf{w}}| = \infty.$$

Jede Punktmenge $\odot(\mathbf{w}^{(nl+2)})$, $n \in \mathbb{N}_0$, ist endlich; daher sind unter den Mengen $\odot(\mathbf{w}^{(nl+2)})$ unendlich viele verschiedene. Sind die Knotenmengen zweier Wörter $\mathbf{x}, \mathbf{y} \in \mathcal{A}^*$ unterschiedlich:

$$\odot(\mathbf{x}) \neq \odot(\mathbf{y}),$$

so sind auch die Bilder verschieden:

$$p(x) \neq p(y).$$

Daher gibt es unter den Bildern $p(\mathbf{w}^{(nl+2)})$, $n \in \mathbb{N}_0$, unendlich viele verschiedene, d. h.

$$|B_{\mathbf{w}}| = \infty.$$

Mit $\mathbf{w} = \omega$ ist die Menge B_{ω} eine Teilmenge der erzeugten Bildsprache $B(G)$ und es folgt

Lemma 3.11 *Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein sDOL-System mit $k = 1$. Wenn es einen Buchstaben $x \in [\omega'']$ und eine Ableitungsstufe $l \in \{1, 2, 3\}$ gibt, so daß*

$$\|_x x \neq \|_x x^{(l)}$$

gilt, dann ist die erzeugte Bildmenge unendlich

$$|B(G)| = \infty.$$

Mit $\mathbf{w} = \omega$ liefern die Lemmata 3.10 und 3.11

Satz 4 *Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein sDOL-System mit $k = 1$. Wenn die Kantenmengen der 2. und 3. Ableitung des Axioms ω nicht übereinstimmen, so ist die erzeugte Bildmenge unendlich:*

$$\|\omega'' \neq \|\omega''' \implies |B(G)| = \infty.$$

Zusammengefaßt liefern die Sätze 3 und 4

Satz 5 *Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein sDOL-System mit $k = 1$. Die erzeugte Bildmenge $B(G)$ ist genau dann unendlich, wenn die Kantenmengen der zweiten und dritten Ableitung des Axiom ω nicht übereinstimmen:*

$$\|\omega'' \neq \|\omega''' \iff |B(G)| = \infty.$$

Aus algorithmischer Sicht ist der Fall $m = 0$ hervorzuheben.

Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein sDOL-System mit $k = 1$ und $m = 0$. Aus der Synchronisiertheit folgt

Folgerung 3.12 *Für alle Buchstaben $x \in \mathcal{A}$ und Bezugspunkte $\mathbf{a} \in \mathbb{Z}^2$ gilt*

$$p^{\mathbf{a}}(x') = (\{\mathbf{a}, x(\mathbf{a})\}, \{(\mathbf{a}, x(\mathbf{a})), (x(\mathbf{a}), \mathbf{a})\}) = p^{\mathbf{a}}(x) \quad (\mathbf{a} \in \mathbb{Z}^2).$$

Beweis: Auf Grund der zweiten Synchronisationsbedingung besteht die Knotenmenge $\odot(x')$ aus den Knoten \mathfrak{o} und \mathfrak{v}_x , d. h.

$$\odot^{\mathfrak{a}}(x') = \{ \mathfrak{a}, \mathfrak{v}_x + \mathfrak{a} \} = \{ \mathfrak{a}, x(\mathfrak{a}) \}.$$

Da die ungerichteten Graphen $p^{\mathfrak{a}}(x')$, $p^{\mathfrak{a}}(x)$ zusammenhängend und schlicht sind, gibt es zwischen den Knoten \mathfrak{a} und $x(\mathfrak{a})$ jeweils genau eine Kante. \spadesuit

Es sei \mathbf{w} ein Wort aus \mathcal{A}_l^* . Mit Folgerung 2.13 ergibt sich

$$\begin{aligned} p(\mathbf{w}') &= p(w'_1) \cup p^{w'_1(\mathfrak{o})}(w'_2) \cup \dots \cup p^{(w'_1 \dots w'_{l-1})(\mathfrak{o})}(x'_l) \\ &= p(w'_1) \cup p^{w_1(\mathfrak{o})}(w'_2) \cup \dots \cup p^{(w_1 \dots w_{l-1})(\mathfrak{o})}(w'_l) \quad (\text{Folgerung 3.5}) \\ &= p(w_1) \cup p^{w_1(\mathfrak{o})}(w_2) \cup \dots \cup p^{(w_1 \dots w_{l-1})(\mathfrak{o})}(w_l) \quad (\text{Folgerung 3.12}) \\ &= p(\mathbf{w}). \end{aligned}$$

Es gelte $p(\mathbf{w}) = p(\mathbf{w}^{(i)})$ für alle $i \leq n$. Es sei $\mathbf{w}^{(n)} \in \mathcal{A}_l^*$. Dann gilt

$$\begin{aligned} p(\mathbf{w}^{(n+1)}) &= p(w'_1) \cup p^{w'_1(\mathfrak{o})}(w'_2) \cup \dots \cup p^{(w'_1 \dots w'_{l-1})(\mathfrak{o})}(w'_l) \\ &= p(w_1) \cup p^{w_1(\mathfrak{o})}(w_2) \cup \dots \cup p^{(w_1 \dots w_{l-1})(\mathfrak{o})}(w_l) \quad (\text{Folg. 3.5 und 3.12}) \\ &= p(\mathbf{w}^{(n)}) \\ &= p(\mathbf{w}) \quad (\text{Induktionsvoraussetzung}). \end{aligned}$$

Das bedeutet, daß die Bilder aller Ableitungen eines Wortes $\mathbf{w} \in \mathcal{A}^*$ mit dem Bild von \mathbf{w} übereinstimmen:

$$p(\mathbf{w}) = p(\mathbf{w}^{(n)}), \quad n \in \mathbb{N}.$$

Mit $\mathbf{w} = \omega$ folgt daraus

Satz 6 *Es sei $G = (\mathcal{A}, h, \omega, k, m)$ ein sDOL-System mit $k = 1$ und $m = 0$. Dann ist die erzeugte Bildsprache $B(G)$ einelementig, und es gilt*

$$B(G) = \{ p(\omega) \}.$$

Das bedeutet, daß im Falle $m = 0$ die Endlichkeit sofort (ohne weitere Untersuchung des Systems) entschieden werden kann.

4 Zusammenfassung und Ausblick

Ketten-Code-Bild-Systeme sind LINDENMAYER-Systeme über einem speziellen Alphabet. Die erzeugten Wörter werden als Bilder interpretiert. Dies führt zu Ketten-Code-Bild-Sprachen. Die vorliegende Arbeit untersucht synchrone, deterministische Ketten-Code-Bild-Systeme über dem Alphabet $\{r, u, l, d\}$ hinsichtlich der Endlichkeit der von ihnen erzeugten Bildsprachen.

Zunächst wird eine Abstrahierungshierarchie entwickelt, in der die Interpretation eines Wortes als Bild einen mehrstufigen Prozeß durchläuft. Die unterste Schicht bilden Wörter einer Sprache. Den Wörtern sind gerichtete Graphen (mit Mehrfachkanten) auf der nächsthöheren Ebene zugeordnet. Durch Abstrahieren von den Kantenwiederholungen (dritte Stufe) und Kantenrichtungen gelangt man schließlich auf die Ebene der Bilder. Anhand dieses Modells wird nachgewiesen, daß es entscheidbar ist, ob ein *sDOL*-System eine endliche oder unendliche Bildmenge erzeugt. Die Menge aller *sDOL*-Systeme läßt sich nach dem Synchronisationsparameter k in folgende Teilmengen zerlegen:

- $\underline{\mathcal{S}} = \{ (\mathcal{A}, h, \omega, k, m) \mid k < 1 \},$
- $\mathcal{S} = \{ (\mathcal{A}, h, \omega, k, m) \mid k = 1 \},$
- $\overline{\mathcal{S}} = \{ (\mathcal{A}, h, \omega, k, m) \mid k > 1 \}.$

Im Falle $k < 1$ generiert jedes *sDOL*-System $G \in \underline{\mathcal{S}}$ eine endliche Bildmenge mit höchstens vier Elementen; im Falle $k > 1$ ist die erzeugte Bildsprache jedes Systems $G \in \overline{\mathcal{S}}$ unendlich. Für $k = 1$ gibt es sowohl *sDOL*-Systeme, die eine endliche Menge von Bildern liefern, als auch solche, die eine unendliche Bildmenge erzeugen. Mittels einer notwendigen und hinreichenden Bedingung kann jedoch nach dreimaligem Ableiten des Startworts die Endlichkeit bzw. Unendlichkeit entschieden werden. Falls die Bildmenge endlich ist, so enthält sie höchstens drei Elemente. Gilt für die Synchronisationsparameter $k = 1$ und $m = 0$, so ist die Bildsprache stets einelementig; ein dreimaliges Ableiten des Axioms ist daher in diesem Spezialfall nicht nötig. Die folgende Tabelle faßt die Ergebnisse zusammen:

$G = (\mathcal{A}, h, \omega, k, m)$ mit $k < 1$:	$B(G) = \{ p(\omega), p(\omega'), p(\omega''), p(\omega''') \}$
$G = (\mathcal{A}, h, \omega, k, m)$ mit $k > 1$:	$ B(G) = \infty$
$G = (\mathcal{A}, h, \omega, k, m)$ mit $k = 1$:	
allgemein:	$sg(\omega'') = sg(\omega''') \implies B(G) = \{ p(\omega), p(\omega'), p(\omega'') \}$
	$sg(\omega'') \neq sg(\omega''') \implies B(G) = \infty$
mit $m = 0$:	$B(G) = \{ p(\omega) \}$

Tabelle 5: Zusammenfassung

Aus dieser Aufspaltung folgt: Wenn die von einem *sDOL*-System erzeugte Bildsprache endlich ist, so enthält sie höchstens vier Elemente. Außerdem ist damit ein Algorithmus gegeben, der zu einem beliebigen *sDOL*-System G entscheidet, ob die erzeugte Bildsprache $B(G)$ endlich ist oder nicht.

Gilt bei einem gegebenen *sDOL*-System $G = (\mathcal{A}, h, \omega, k, m)$ für die Parameter $k < 1$, $k > 1$ oder $k = 1, m = 0$, so kann die Entscheidung, ob die erzeugte Bildmenge endlich ist oder nicht, sofort (ohne eine weitere Untersuchung des Systems), d. h. mit konstantem Zeitaufwand getroffen werden. Sonst ist das Startwort dreimal abzuleiten, und es sind die Kantenmengen der zweiten und dritten Ableitung auf Übereinstimmung zu prüfen. Damit ergibt sich ein Zeitaufwand, der kubisch in der Länge der ersetzenden Wörter ist.

Das bedeutet insgesamt: Für synchrone, deterministische, kontextfreie Ketten-Code-Bild-Systeme $G = (\mathcal{A}, h, \omega, k, m)$ sind Endlich- und Unendlichkeit in der Zeit $\mathcal{O}(pn^3)$ entscheidbar, wobei p die Länge des Startwortes und n die maximale Länge der ersetzenden Wörter sind: $p = \#\omega$ und $n = \max \{ \#h(x) \mid x \in \mathcal{A} \}$.

An diese Arbeit können allgemeinere Untersuchungen angeknüpft werden, indem z. B. (nichtdeterministische) *sOL*-Systeme oder tabellierte Systeme (*sTOL*, *sDTOL*) als Grundlage genommen werden oder die Synchronisation aufgegeben wird. Dies ist dadurch motiviert, daß *sDOL*-Systeme im endlichen Falle höchstens vier Bilder erzeugen. Für Anwendungen sind jedoch Systeme wünschenswert, die eine große aber dennoch endliche Menge von Bildern bereitstellen. Eine andere Möglichkeit ist der Übergang zu einem Alphabet, das um die „Befehle“ *Stift anheben*, *Stift absenken* angereichert wird. Damit können auch nichtzusammenhängende Bilder generiert werden.

Eine Reihe anderer Probleme – z. B. die Äquivalenz zweier Ketten-Code-Bild-Systeme betreffend – sind weiterhin ungelöst.

Literatur

- [Cho56] CHOMSKY, N.: *Three models for the description of language*. IRE Trans. on Inform. Theory, 2(3):113–124, 1956.
- [Das88] DASSOW, J.: *Convexity and Simplicity of Chain Code Picture Languages*. Rostocker Math. Kolloq., 34:53–60, 1988.
- [DH89] DASSOW, J.; HINZ, F.: *Kettenkode-Bildsprachen. Theorie und Anwendungen*. Wiss. Zeitschrift d. Techn. Univ. Magdeburg, 33, 1989.
- [DHr92] DASSOW, J.; HROMKOVIC, J.: *On Synchronized Lindenmayer Picture Languages*. In: *Lindenmayer Systems*, Seiten 253–261. Springer-Verlag, 1992.
- [DH93] DASSOW, J.; HINZ, F.: *Decision problems and regular chain code picture languages*. Discrete Applied Mathematics, 45:29–49, 1993.
- [Fre61] FREEMAN, H.: *On the encoding of arbitrary geometric configurations*. IRE Trans. EC, 10:260–268, 1961.
- [Fre74] FREEMAN, H.: *Computer processing of line-drawing images*. Computer Surveys, 6:57–97, 1974.
- [Hol99] HOLLATZ, H.: *Mathematische Grundlagen der Informatik*. <http://horst.hollatz.de>, Otto-von-Guericke-Universität Magdeburg, Fakultät für Mathematik, 1999.
- [LW87] LENGAUER, T.; WAGNER, K. W.: *The correlation between the complexities of the non-hierarchical and hierarchical version of graph problems*. In: *Proc. STACS'87*, Band 247 der Reihe LNCS, Seiten 100–113. Springer-Verlag, 1987.
- [Lin68] LINDENMAYER, A.: *Mathematical models for cellular interaction in development*. Journal of Theoretical Biology, 18:280–315, 1968.
- [M⁺82] MAURER, H.; ROZENBERG, G.; WELZL, E.: *Using string languages to describe picture languages*. Information and Control, 54:155–185, 1982.
- [M⁺94] MELNIKOV, O.; TYSHKEVICH, R.; YEMELICHEV, V.; SARVANOV, V.: *Lectures on Graph Theory*. BI-Wissenschaftsverlag, 1994.
- [PL90] PRUZINKIEWICZ, P.; LINDENMAYER, A.: *The Algorithmic Beauty of Plants*. Springer Verlag, New York, 1990.
- [RS80] ROZENBERG, G.; SALOMAA, A.: *The Mathematical Theory of L Systems*. Academic Press, 1980.

- [SW85] SUDBOROUGH, I. H.; WELZL, E.: *Complexity and decidability for chain code picture languages*. Theoretical Computer Science, 36:173–202, 1985.
- [Wel83] WELZL, E.: *Chain code picture languages*. Dissertation, Techn. Univ. Graz, 1983. IIG-Report F 124.

Erklärung

Ich versichere, die vorliegende Arbeit selbständig angefertigt und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet zu haben.

Magdeburg, den 15. April 1999

.....

Bianca Truthe

Thesen

1. Wörter lassen sich als Bilder interpretieren und damit graphisch darstellen. CHOMSKY-Grammatiken und LINDENMAYER-Systeme erzeugen Wortmengen; diesen werden über eine Interpretation Bildmengen zugeordnet.
2. Spezielle bildgenerierende LINDENMAYER-Systeme sind synchrone, deterministische, kontextfreie L -Systeme (genannt *Ketten-Code-Bild-Systeme*) über dem Alphabet $\{r, u, l, d\}$. Die entstehenden Bilder sind achsenparallele Strichgraphiken.
3. Ein $sDOL$ -System ist ein 5-Tupel

$$G = (\{r, u, l, d\}, h, \omega, k, m),$$

wobei $(\{r, u, l, d\}, h, \omega)$ ein LINDENMAYER-System darstellt, das durch die Parameter k und m synchronisiert wird.

4. Die Menge der $sDOL$ -Systeme $G = (\{r, u, l, d\}, h, \omega, k, m)$ läßt sich nach dem Synchronisationsparameter k zerlegen in Systeme mit $k < 1$, $k = 1$, $k > 1$.
5. Ein $sDOL$ -System $G = (\{r, u, l, d\}, h, \omega, k, m)$ mit $k < 1$ erzeugt höchstens vier verschiedene Bilder.
6. Ein $sDOL$ -System $G = (\{r, u, l, d\}, h, \omega, k, m)$ mit $k = 1$ erzeugt höchstens drei verschiedene Bilder, wenn die Bilder der 2. und 3. Ableitung von ω übereinstimmen und jede Bildkante des einen Bildes auf die gleiche Art wie die entsprechende Bildkante des anderen Bildes entstanden ist, sonst entstehen unendlich viele Bilder.
7. Ein $sDOL$ -System $G = (\{r, u, l, d\}, h, \omega, k, m)$ mit $k > 1$ erzeugt unendlich viele Bilder.
8. Es ist in kubischer Zeit entscheidbar, ob ein $sDOL$ -System eine endliche Bildmenge erzeugt oder nicht. Falls eine Bildmenge endlich ist, so enthält sie höchstens vier Elemente.