Turing — Berechenbarkeit in linearer Zeit

Von K. WAGNER

Einleitung

Eine interessante Aufgabe der Kompliziertheitstheorie ist es, die Klasse der Funktionen zu untersuchen, die durch Turingmaschinen berechnet werden, an die bestimmte Kompliziertheitsforderungen gestellt sind. Das gleiche gilt für die Klasse der Mengen, die durch solche Turingmaschinen entschieden werden. Oft wird angestrebt, solche Klassen von Funktionen bzw. Mengen ohne Benutzung kompliziertheitstheoretischer Begriffe zu charakterisieren. Dies ist z. B. für die in Realzeit berechenbaren Funktionen und entscheidbaren Mengen bereits gelungen. TRACHTENBROT stellt in [1] solche Untersuchungen für die in linearer Zeit berechenbaren überall definierten Funktionen und die in linearer Zeit entscheidbaren und aufzählbaren Mengen an. Das Ziel dieser Arbeit ist es, die dort erzielten Ergebnisse für die in linearer Zeit berechenbaren partiellen Funktionen zu verallgemeinern und noch einige neue Gesichtspunkte hinzuzufügen. So sind die Sätze 1, 2 und 6 sowie die Folgerungen 1 und 2 Verallgemeinerungen der TRACHTENBROTschen Ergebnisse im eben angegebenen Sinne.

Definitionen

Mit X^* bezeichnen wir die Wortmenge über dem endlichen Alphabet X, mit e das leere Wort. Für $w \in X^*$ sei $|w|$ die Länge des Wortes w, also

$$|e| = 0$$

und

$$|wa| = |w|+1 \quad \text{für} \quad w \in X^* \quad \text{und} \quad a \in X.$$

J sei die Operation der Wortinversion, also

$$J(e) = e$$

und

$$J(wa) = aJ(w) \quad \text{für} \quad w \in X^* \quad \text{und} \quad a \in X.$$

Es seien X und Y endliche Alphabete. Eine partielle Wortfunktion f über $[X, Y]$ ist eine Funktion aus X^* in Y^*. Mit D_f und R_f bezeichnen wir den Definitionsbereich bzw. den Wertevorrat der Funktion f. Es sei f eine partielle Wortfunktion über $[Y, Z]$ und g eine partielle Wortfunktion über $[X, Y]$. Dann ist $f \circ g$ eine partielle Wortfunktion über $[X, Z]$ mit

$$(f \circ g)(w) =_{Df} f(g(w)) \quad \text{für alle} \quad w \in X^*.$$

Eine Turingmaschine $\mathfrak{M}=[X, Z, f, g, h, z_0, z_1]$ sei in dieser Arbeit stets eine *einbändrige einköpfige Turingmaschine* mit dem Arbeitsalphabet X, der Zustandsmenge Z, der Überführungsfunktion $f: Z \times X \to Z$, der Ausgabefunktion $g: Z \times X \to X$, der Transportfunktion $h: Z \times X \to \{+1, -1\}$, dem Anfangszustand z_1 und dem Endzustand z_0. Die Arbeitsweise einer solchen Turingmaschine wird in [1] beschrieben. Hier sei nur erwähnt, daß die Turingmaschine $\mathfrak{M}$ ihre Arbeit auf dem Wort $w \in (X - \{\text{⫯}\})^*$ in dem Feld beginnt, in dem der erste Buchstabe von w steht. Dabei sei $\text{⫯} \in X$ das Leerzeichen von $\mathfrak{M}$.

Eine Wortfunktion φ über $[A, B]$ mit $A \cup B \subseteq X$ wird durch die Turingmaschine $\mathfrak{M}$ berechnet $=_{Df}$.

1. Es ist $w \in D_\varphi$ genau dann, wenn $\mathfrak{M}$ die Arbeit auf dem Wort w nach endlich vielen Schritten beendet.

2. Ist $w \in D_\varphi$, so steht nach Beendigung der Arbeit von $\mathfrak{M}$ auf w auf dem Band von $\mathfrak{M}$ das nicht durch Zeichen ⫯ unterbrochene Wort $\varphi(w)$.

Es sei weiter $\mathfrak{M}$ eine Turingmaschine, die ihre Arbeit auf dem Wort w nach endlich vielen Schritten beendet. Wir bezeichnen wie in [1] mit

$t_{\mathfrak{M}}(w)$ die Anzahl der Takte, die $\mathfrak{M}$ zur Bearbeitung des Wortes w benötigt,

$r_{\mathfrak{M}}(w)$ die maximale Anzahl von Überschreitungen einer Feldgrenze bei der Arbeit von $\mathfrak{M}$ auf w und

$S_{\mathfrak{M}}(w)$ die Anzahl der Felder, die bei der Arbeit von $\mathfrak{M}$ auf w benötigt werden.

Beendet $\mathfrak{M}$ die Arbeit auf w nicht nach endlich vielen Schritten, so sind $t_{\mathfrak{M}}(w)$, $r_{\mathfrak{M}}(w)$ und $S_{\mathfrak{M}}(w)$ nicht definiert.

Die Folge von Zuständen, mit denen die Turingmaschine $\mathfrak{M}$ bei ihrer Arbeit auf dem Wort w eine bestimmte Feldgrenze überschreitet, nennen wir die Pendelfolge dieser Feldgrenze bei der Bearbeitung von w durch $\mathfrak{M}$. Dabei werden die Zustände einer Pendelfolge einer Feldgrenze so markiert, daß die jeweilige Richtung der Feldgrenzenüberschreitung ersichtlich ist.

Die partielle Wortfunktion φ heißt durch die Turingmaschine $\mathfrak{M}$ in der Zeit $T(n)$ berechnet, wenn die Turingmaschine $\mathfrak{M}$ die Funktion φ berechnet und für jedes Wort $w \in D_\varphi$ die Beziehung

$$t_{\mathfrak{M}}(w) \leqq T(|w|)$$

gilt. Die partielle Wortfunktion φ heißt in linearer Zeit berechenbar, wenn es eine Turingmaschine gibt, die φ in einer Zeit $T(n)=O(n)$ berechnet.

Die partielle Wortfunktion φ heißt mit beschränktem Regime berechenbar, wenn es eine Konstante c und eine Turingmaschine $\mathfrak{M}$, die φ berechnet, gibt, so daß für jedes Wort $w \in D_\varphi$ die Beziehung $r_{\mathfrak{M}}(w) \leqq c$ gilt. Eine Menge $M \subseteq X^*$ heißt in linearer Zeit (mit beschränktem Regime) entscheidbar, wenn für ein festes $a \in X$ die Funktion χ_M^a mit

$$\chi_M^a(w) =_{Df} \begin{cases} a, & \text{falls } w \in M \\ e, & \text{falls } w \notin M \end{cases}$$

in linearer Zeit (mit beschränktem Regime) berechenbar ist.

1. In linearer Zeit berechenbare Funktionen

Wir geben zunächst ein Lemma an, das in [1] bewiesen ist und das wir beim Beweis von Satz 1 benötigen werden.

Lemma 1 (TRACHTENBROT). Ist L eine Menge von l paarweise verschiedenen Wörtern über dem Alphabet X mit k Buchstaben, so ist

$$\sum_{w \in L} |w| \geqq c \cdot l \cdot \log_k l,$$

wobei c eine von k abhängige positive Konstante ist.

Satz 1. Berechnet eine Turingmaschine $\mathfrak{M}$ die partielle Wortfunktion φ über $[X, Y]$ in einer Zeit $T(n) = o(n \cdot \log n)$, so berechnet $\mathfrak{M}$ die Funktion φ mit beschränktem Regime.

Beweis: Wir schließen indirekt und nehmen an, daß die Turingmaschine $\mathfrak{M}$ die Funktion φ zwar in einer Zeit $T(n) = o(n \cdot \log n)$ berechnet, jedoch nicht mit beschränktem Regime. Es gibt also eine Folge $\{w_\nu\}_{\nu=1,2,\ldots}$ von Wörtern über X mit

$$r_{\mathfrak{M}}(w_1) < r_{\mathfrak{M}}(w_2) < \ldots < r_{\mathfrak{M}}(w_\nu) < \ldots.$$

Wir konstruieren aus dieser Folge eine neue Folge $\{v_\nu\}_{\nu=1,2,\ldots}$ von Wörtern über X mit

$$r_{\mathfrak{M}}(v_1) < r_{\mathfrak{M}}(v_2) < \ldots < r_{\mathfrak{M}}(v_\nu) < \ldots \tag{a}$$

und

$$t_{\mathfrak{M}}(v_\nu) \geqq c \cdot |v_\nu| \cdot \log |v_\nu| \tag{b}$$

mit geeigneter Konstante $c > 0$.

Da wegen (a) die Glieder der Folge $\{v_\nu\}_{\nu=1,2,\ldots}$ paarweise verschieden sind, wächst die Folge $\{|v_\nu|\}_{\nu=1,2,\ldots}$ über alle Schranken und somit ist $T(n) \neq o(n \cdot \log n)$. Nun zur Konstruktion der Folge $\{v_\nu\}_{\nu=1,2,\ldots}$.

Steht ein Wort w auf dem Band der Maschine $\mathfrak{M}$, so bezeichnen wir mit $\Gamma(w)$ die Menge der Grenzen derjenigen Felder, in denen ein Buchstabe von w steht, mit $\lambda(w)$ die linke Grenze des Feldes, in dem der erste Buchstabe von w steht und mit $\varrho(w)$ die rechte Grenze des Feldes, in dem der letzte Buchstabe von w steht.

Es sei nun $w_\nu = a_\nu \cdot w'_\nu$, wobei $a_\nu \in X$ ist.

Fall 1. Bei der Bearbeitung von w_ν durch $\mathfrak{M}$ wird das Maximum $r_{\mathfrak{M}}(w_\nu)$ von Grenzüberschreitungen auf den Feldgrenzen aus $\Gamma(w'_\nu)$ nicht angenommen. Gibt es ein Teilwort u von w'_ν mit $s(\lambda(u)) = s(\varrho(u))$ ($s(i)$ ist dabei die an der Grenze i bei dieser Berechnung entstehende Pendelfolge), so wird dieses u in w'_ν durch das leere Wort e ersetzt. Mit dem so erhaltenen Wort verfahren wir wie mit w'_ν usw. Nach endlich vielen Schritten gelangt man zu einem Wort v'_ν, das kein Teilwort u besitzt mit $s(\lambda(u)) = s(\varrho(u))$. Dann setzen wir $v_\nu =_{\mathrm{Df}} a_\nu \cdot v'_\nu$*).

Fall 2. Es gibt Wörter p_ν und q_ν mit $w'_\nu = p_\nu \cdot q_\nu$ und auf der Feldgrenze $\varrho(p_\nu) = \lambda(q_\nu)$ wird das Maximum $r_{\mathfrak{M}}(w_\nu)$ von Grenzüberschreitungen bei der Bearbeitung von w_ν durch $\mathfrak{M}$ angenommen. Mit p_ν und q_ν wird genauso verfahren wie im Fall 1 mit w'_ν, und wir erhalten die Wörter p'_ν und q'_ν. Nun setzen wir $v_\nu =_{\mathrm{Df}} a_\nu \cdot p'_\nu \cdot q'_\nu$.

* Dieses Verfahren ist im allgemeinen nicht eindeutig bestimmt, denn es kann zu einem Wort möglicherweise mehrere Teilwörter mit der oben angegebenen Bedingung geben. Es genügt, jeweils irgendeine der möglichen Ersetzungen durchzuführen.

Die Folge $\{v_\nu\}_{\nu=1,2,\ldots}$ besitzt nun die folgenden Eigenschaften.

$$\text{Für jedes } \nu = 1, 2, \ldots \text{ ist } \varphi(v_\nu) \text{ definiert.} \tag{1}$$

Da die Anfangsbuchstaben von w_ν und v_ν gleich sind, wird zu Beginn der Arbeit von $\mathfrak{M}$ auf v_ν der gleiche Befehl angewendet wie bei w_ν, also gleicher Zustand und gleiche Bewegungsrichtung. Die Arbeit von $\mathfrak{M}$ verläuft auf v_ν ebenso, wie sie bei der Bearbeitung von w_ν durch $\mathfrak{M}$ auf den Teilwörtern von w_ν verläuft, aus denen v_ν zusammengesetzt ist. Dies sichert die Bedingung $s(\lambda(u))=s(\varrho(u))$ für ein Teilwort u, das beim Übergang von w_ν zu v_ν eliminiert wird. Andererseits besitzen für ein Teilwort u von w_ν die Folgen $\lambda(u)$ und $\varrho(u)$ verschiedene Längen, falls bei der Bearbeitung von w_ν die Turingmaschine $\mathfrak{M}$ ihre Arbeit zwischen $\lambda(u)$ und $\varrho(u)$ beendet und ihre Arbeit dort nicht beginnt. Somit kann ein solches Teilwort u von w_ν beim Übergang von w_ν zu v_ν nicht ersetzt werden, und die Turingmaschine $\mathfrak{M}$, die φ berechnet, beendet ihre Arbeit auf v_ν ebenfalls nach endlich vielen Takten.

$$\text{Für jedes} \quad \nu = 1, 2, \ldots \quad \text{gilt} \quad r_{\mathfrak{M}}(v_\nu) = r_{\mathfrak{M}}(w_\nu). \tag{2}$$

Diese Folgerung ist evident, denn jede Pendelfolge, die bei der Bearbeitung von v_ν durch $\mathfrak{M}$ entsteht, entsteht auch bei der Bearbeitung von w_ν durch $\mathfrak{M}$. Mindestens eine Feldgrenze, auf der das Maximum von Grenzüberschreitungen bei der Bearbeitung von w_ν durch $\mathfrak{M}$ angenommen wird, bleibt jedoch beim Übergang von w_ν zu v_ν erhalten.

Trivialerweise gilt nun auch

$$r_{\mathfrak{M}}(v_1) < r_{\mathfrak{M}}(v_2) < \ldots < r_{\mathfrak{M}}(v_\nu) < \ldots. \tag{3}$$

Schlißlich zeigen wir: Es gibt eine Konstante $c > 0$ mit

$$t_{\mathfrak{M}}(v_\nu) \geqq c \cdot |v_\nu| \cdot \log|v_\nu| \quad \text{für alle} \quad \nu = 1, 2, \ldots. \tag{4}$$

Im oben betrachteten Fall 1 ist $|v'_\nu| \geqq \frac{1}{2} |v_\nu|$. Die Anzahl der Takte, die $\mathfrak{M}$ in dem zwischen $\lambda(v'_\nu)$ und $\varrho(v'_\nu)$ liegenden Bereich arbeitet, ist gleich

$$\sum_{\gamma \in \Gamma(v'_\nu)} |s(\gamma)|.$$

Da $s(\Gamma(v'_\nu))$ aus $|v'_\nu|+1$ verschiedenen Pendelfolgen besteht, gilt nach Lemma 1

$$t_{\mathfrak{M}}(v_\nu) \geqq \sum_{\gamma \in \Gamma(v'_\nu)} |s(\gamma)| \geqq c'|v'_\nu| \cdot \log|v'_\nu| \quad \text{mit} \quad c' > 0.$$

Wegen $|v'_\nu| \geqq \frac{1}{2}|v_\nu|$ existiert eine Konstante $c > 0$ mit

$$t_{\mathfrak{M}}(v_\nu) \geqq c \cdot |v_\nu| \cdot \log|v_\nu|.$$

Im oben betrachten Fall 2 gilt entweder $|p'_\nu| \geqq \frac{1}{3} |v_\nu|$ oder $|q'_\nu| \geqq \frac{1}{3} |v_\nu|$. Hier ergibt sich die gleiche Rechnung wie im Fall 1. Damit ist Satz 1 bewiesen.

Satz 2. Berechnet die Turingmaschine $\mathfrak{M}$ die partielle Wortfunktion φ über $[X, Y]$ mit beschränktem Regime, so berechnet $\mathfrak{M}$ die Funktion φ in linearer Zeit.

Beweis: Wir wählen eine Konstante R, so daß für jedes $w \in D_\varphi$ die Ungleichung $r_{\mathfrak{M}}(w) \leqq R$ gilt. Ist $w \in D_\varphi$, so erhält man

$$t_{\mathfrak{M}}(w) \leqq R \cdot (S(w)+1).$$

Da es nur endlich viele Pendelfolgen mit der maximalen Länge R gibt und wegen $w \in D_\varphi$ sowie links von $\lambda(w)$ als auch rechts von $\varrho(w)$ jede dieser Pendelfolgen nur einmal auftreten kann, gibt es eine Konstante c' mit $S(w) \leqq |w| + c'$. Mithin gibt es eine Konstante c mit $t_{\mathfrak{M}}(w) \leqq c \cdot |w|$. Damit ist Satz 2 bewiesen.

Folgerung 1. Berechnet eine Turingmaschine $\mathfrak{M}$ die partielle Wortfunktion φ in einer Zeit $T(n) = o(n \cdot \log n)$, so berechnet $\mathfrak{M}$ die Funktion φ bereits in linearer Zeit.

Für eine zweistellige partielle Wortfunktion ψ über $[X, Y]$ definieren wir für alle $w \in X^*$ die Funktionen $\psi_{1,w}$ und $\psi_{2,w}$ mit

$$\psi_{1,w}(w') =_{\mathrm{Df}} \psi(w, w')$$

und

$$\psi_{2,w}(w') =_{\mathrm{Df}} \psi(w', w).$$

Weiter sei $G_1(\psi) =_{\mathrm{Df}} kz\{\psi_{1,w},\ w \in X^*\}$ das Gewicht von ψ in der 1. Stelle und $G_2(\psi) =_{\mathrm{Df}} kz\{\psi_{2,w},\ w \in X^*\}$ das Gewicht von ψ in der 2. Stelle.

Satz 3. Eine partielle Wortfunktion φ über $[X, Y]$ ist genau dann mit beschränktem Regime berechenbar, wenn es zwei zweistellige partielle Wortfunktionen φ_l und φ_r gibt mit

1) Die Werte $\varphi(w, w')$, $\varphi_l(w, w')$ und $\varphi_r(w, w')$ sind alle gleichzeitig definiert oder nicht definiert. Im Falle ihrer Existenz gilt

$$\varphi(w \cdot w') = \varphi_l(w, w') \cdot \varphi_r(w, w').$$

2) Ist $\varphi(waw')$ mit $a \in X$ definiert, so gilt $|\varphi_l(wa, w')| = |\varphi_l(w, aw')| + 1$, falls $\varphi_l(wa, w') \neq e$ und $\varphi_r(w, aw') \neq e$ und es gilt $|\varphi_l(wa, w')| = |\varphi_l(w, aw')|$, sonst.

3) Es sind $G_2(\varphi_l)$ und $G_1(\varphi_r)$ endlich und aus $(\varphi_l)_{2,w} = (\varphi_l)_{2,w'}$ $\left((\varphi_r)_{1,w} = (\varphi_r)_{1,w'}\right)$ folgt $(\varphi_l)_{2,aw} = (\varphi_l)_{2,aw'}$ $\left((\varphi_r)_{1,wa} = (\varphi_r)_{1,w'a}\right)$ für alle $a \in X$.

Bemerkung: Man sieht wegen 1) leicht ein, daß 2) durch folgende Bedingung äquivalent ersetzt werden kann.

2′) Ist $\varphi(waw')$ mit $a \in X$ definiert, so gilt $|\varphi_r(w, aw')| = |\varphi_r(wa, w')| + 1$, falls $\varphi_l(wa, w') \neq e$ und $\varphi_r(w, aw') \neq e$ und es gilt $|\varphi_r(w, aw')| = |\varphi_r(wa, w')|$, sonst.

Beweis von Satz 3: I. Die Turingmaschine $\mathfrak{M}$ berechne φ mit beschränktem Regime. Ist $\varphi(w \cdot w')$ für $w, w' \in X^*$ definiert, so setzen wir $\varphi_l(w, w')$ gleich dem Wort, welches nach der Bearbeitung von $w \cdot w'$ durch $\mathfrak{M}$ links von $\lambda(w') = \varrho(w)$ steht und $\varphi_r(w, w')$ gleich dem Wort, welches nach der Bearbeitung von $w \cdot w'$ durch $\mathfrak{M}$ rechts von $\lambda(w') = \varrho(w)$ steht. Ist $\varphi(w \cdot w')$ nicht definiert, so seien $\varphi_l(w, w')$ und $\varphi_r(w, w')$ ebenfalls nicht definiert.

Die Bedingungen 1) und 2) sind trivialerweise erfüllt. Es ist die Bedingung 3) nachzuweisen. Wir zeigen zunächst, daß $G_2(\varphi_l)$ endlich ist. Da die Turingmaschine $\mathfrak{M}$ mit beschränktem Regime arbeitet, gibt es eine natürliche Zahl R, so daß $r_{\mathfrak{M}}(v) \leqq 2R+1$ für alle $v \in D_\varphi$ gilt. Außerdem besitze $\mathfrak{M}$ genau k Zustände.

Die Frage nach $G_2(\varphi_l)$ ist die Frage danach, auf wieviel verschiedene Arten das am Beginn einer Berechnung durch $\mathfrak{M}$ rechts von einer gewissen Feldgrenze stehende Wort auf den Berechnungsvorgang links von dieser Feldgrenze Einfluß nehmen kann. Diese Einflußnahme ist jedoch nur möglich durch die Zustände, in denen der Lese- und Schreibekopf von $\mathfrak{M}$ diese Feldgrenze von rechts nach links überschreitet.

Da die erste Überschreitung dieser Grenze von links nach rechts geschieht, interessieren nur die ersten R Überschreitungen dieser Grenze von rechts nach links. Gibt es mehr als R solche Überschreitungen, so ist φ_l auf den zugrundeliegenden Argumenten nicht definiert. Von links nach rechts kann diese Grenze $(R+1)$-mal überschritten werden. Wird diese Feldgrenze in einem bestimmten Zustand von links nach rechts überschritten, so sind $k+2$ „Reaktionen" möglich:

1. Rückkehr über die Grenze mit einem der k möglichen Zustände,
2. $\mathfrak{M}$ beendet die Arbeit rechts von dieser Grenze oder
3. $\mathfrak{M}$ arbeitet rechts von dieser Grenze ad infinitum weiter (also φ_l ist auf diesen Argumenten nicht definiert).

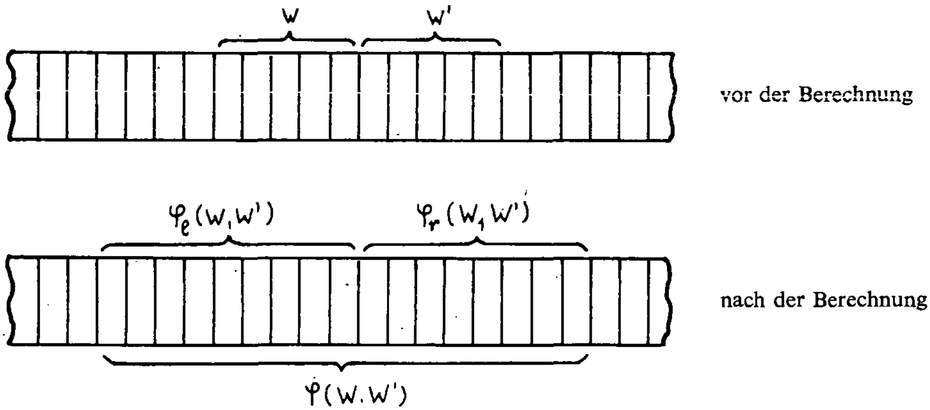

Abb. 1

Insgesamt sind dies $(k+2)^k$ mögliche „Reaktionen" bei jeder Überschreitung der Grenze von links nach rechts. Ausgenommen ist dabei die $(R+1)$-te Überschreitung dieser Grenze von links nach rechts, denn dort folgt aus der Rückkehr über die Grenze, daß φ_l auf diesen Argumenten nicht definiert ist. In diesem Fall gibt es also als „Reaktion" auf einen bestimmten Zustand nur zwei Möglichkeiten, also insgesamt 2^k Möglichkeiten. Damit ergibt sich schließlich $G_2(\varphi_l) \leqq ((k+2)^k)^R \cdot 2^k$. Außerdem ist klar, daß die „Reaktionen" bei einer Überschreitung einer Feldgrenze von links nach rechts nur von den „Reaktionen" beim Überschreiten der nächsten rechts gelegenen Feldgrenze und dem Inhalt der zwischen diesen beiden Feldgrenzen liegenden Feldes abhängt. Also haben wir mit $(\varphi_l)_{2,w} = (\varphi_l)_{2,w'}$ auch $(\varphi_l)_{2,aw} = (\varphi_l)_{2,aw'}$ für alle $a \in X$.

Analog zeigt man, daß $G_1(\varphi_r)$ endlich ist und daß mit $(\varphi_r)_{1,w} = (\varphi_r)_{1,w'}$ auch $(\varphi_r)_{1,wa} = (\varphi_r)_{1,w'a}$ gilt für alle $a \in X$.

II. Es seien φ, φ_l und φ_r Wortfunktionen, die den Bedingungen 1)—3) genügen.

Die Wörter $w, w' \in X^*$ stehen genau dann in der Relation $\sim_l$ $(\sim_r)$, wenn $(\varphi_l)_{2,w} = (\varphi_l)_{2,w'}$ $\big((\varphi_r)_{1,w} = (\varphi_r)_{1,w'}\big)$ ist. Es ist klar, daß $\sim_l$ und $\sim_r$ Äquivalenzrelationen auf X^* sind. Mit $w^{(l)}$ bzw. $w^{(r)}$ bezeichnen wir diejenige Äquivalenzklasse bzgl. $\sim_l$ bzw. $\sim_r$, in der w liegt. Aus 3) folgt nun *a)* Die Äquivalenzrelation $\sim_l$ ist linksstabil bzgl. der Operation der Wortaneinandersetzung.

b) Die Äquivalenzrelation $\sim_r$ ist rechtsstabil bzgl. der Operation der Wortaneinandersetzung.

Ist $\varphi(u \cdot v \cdot w)$ für $u, v, w \in X^*$ definiert, so bezeichnen wir mit $\varphi_m(u, v, w)$ dasjenige Wort, für das gilt

$$\varphi_l(uv, w) = \varphi_l(u, vw) \cdot \varphi_m(u, v, w). \tag{α}$$

Ist $\varphi(u \cdot v \cdot w)$ nicht definiert, so ist $\varphi_m(u, v, w)$ ebenfalls nicht definiert.

Im Falle der Existenz von $\varphi(u \cdot v \cdot w)$ haben wir sofort auch

$$\varphi_r(u, vw) = \varphi_m(u, v, w) \cdot \varphi_l(uv, w). \tag{β}$$

Die Paare $(v_1, w_1), (v_2, w_2) \in X^* \times X^*$ stehen genau dann in der Relation $\sim_m$, wenn $\varphi_m(v_1, u, w_1) = \varphi_m(v_2, u, w_2)$ für jedes $u \in X^*$ gilt. Die Relation $\sim_m$ ist eine Äquivalenzrelation und wir zeigen

c) Aus $u_1 \sim_r u_2$ und $w_1 \sim_l w_2$ folgt $(u_1, w_1) \sim_m (u_2, w_2)$.

Ist $\varphi(u_1 v w_1)$ definiert, so gilt

$$\begin{aligned}
\varphi_m(u_1, v, w_1) \cdot \varphi_r(u_2 v, w_1) &= \varphi_m(u_1, v, w_1) \cdot \varphi_r(u_1 v, w_1) \quad \text{wegen } u_1 \sim_r u_2 \text{ und } b) \\
&= \varphi_r(u_1, v w_1) \quad \text{wegen } (\beta) \\
&= \varphi_r(u_2, v w_1) \quad \text{wegen } u_1 \sim_r u_2 \\
&= \varphi_m(u_2, v, w_1) \cdot \varphi_r(u_2 v, w_1).
\end{aligned}$$

Mithin gilt $\varphi_m(u_1, v, w_1) = \varphi_m(u_2, v, w_1)$. Analog folgt aus $w_1 \sim_l w_2$ die Gleichung $\varphi_m(u_2, v, w_1) = \varphi_m(u_2, v, w_2)$.

Ist $\varphi(u_1 v w_1)$ nicht definiert, so sind auch $\varphi_m(u_1, v, w_1)$ und $\varphi_m(u_2, v, w_2)$ nicht definiert. Damit ist *c)* bewiesen.

Mit *c)* wurde gezeigt, daß $\varphi_m(u, v, w)$ lediglich von v und der $\sim_r$ — bzw. $\sim_l$ — Klasse abhängt, in der u bzw. w liegt. Mit *c)* wurde also die Reräsentantenunabhängigkeit der Definition

$$\varphi'_m(u^{(r)}, v, w^{(l)}) =_{\mathrm{Df}} \varphi_m(u, v, w)$$

gezeigt. Damit haben wir auch

d) Ist $\varphi(a_1 \cdot \ldots \cdot a_n)$ für $a_1, \ldots, a_n \in X$ definiert, so gilt

$$\begin{aligned}
\varphi(a_1 \cdot \ldots \cdot a_n) &= \varphi_l(e, a_1 \cdot \ldots \cdot a_n) \cdot \varphi'_m(e^{(r)}, a_1, (a_2 \cdot \ldots \cdot a_n)^{(l)}) \cdot \ldots \cdot \\
&\quad \ldots \cdot \varphi'_m((a_1 \cdot \ldots \cdot a_{i-1})^{(r)}, a_i, (a_{i+1} \cdot \ldots \cdot a_n)^{(l)}) \cdot \ldots \cdot \\
&\quad \ldots \cdot \varphi'_m((a_1 \cdot \ldots \cdot a_{n-1})^{(r)}, a_n, e^{(l)}) \cdot \varphi_r(a_1 \cdot \ldots \cdot a_n, e),
\end{aligned}$$

wobei $\varphi'_m((a_1 \cdot \ldots \cdot a_{i-1})^{(r)}, a_i, (a_{i+1} \cdot \ldots \cdot a_n)^{(l)}) \in Y \cup \{e\}$ für $i = 1, \ldots, n$.

Nun können wir eine Turingmaschine $\mathfrak{M}$ konstruieren, die die Funktion φ mit beschränktem Regime berechnet. Diese Turingmaschine $\mathfrak{M}$ arbeitet wie folgt: Zunächst ersetzt $\mathfrak{M}$ in dem auf dem Band stehenden Wort $a_1 \cdot \ldots \cdot a_n$ $(a_1, \ldots, a_n \in X)$ von links nach rechts jeden Buchstaben a_i durch das Zeichen $((a_1 \cdot \ldots \cdot a_{i-1})^{(r)}, a_i)$. Dabei hat $\mathfrak{M}$ nach Abarbeitung des Anfangswortes $a_1 \cdot \ldots \cdot a_n$ den Zustand $(a_1 \cdot \ldots \cdot a_i)^{(r)}$ inne.

Nach *a)* ist das ausreichend, um im nächsten Schritt den Zustand $(a_1 \cdot \ldots \cdot a_{i+1})^{(r)}$ annehmen zu können. Am Wortende angelangt, wird $\varphi_r(a_1 \cdot \ldots \cdot a_n, e)$ auf das Band geschrieben, falls $\varphi_r(a_1 \cdot \ldots \cdot a_n, e)$ definiert ist. Nun wird das Wort von rechts nach links durchlaufen, wobei das Feld mit dem Inhalt $((a_1 \cdot \ldots \cdot a_{i-1})^{(r)}, a_i)$ mit dem Zustand $(a_{i+1} \cdot \ldots \cdot a_n)^{(l)}$ erreicht (dies ist nach *b)* möglich) und mit dem Buchstaben $\varphi_m'((a_1 \cdot \ldots \cdot a_{i-1})^{(r)}, a_i, (a_{i+1} \cdot \ldots \cdot a_n)^{(l)})$ beschrieben wird, falls $\varphi_m'((a_1 \cdot \ldots \cdot a_{i-1})^{(r)}, a_i, (a_{i+1} \cdot \ldots \cdot a_n)^{(l)})$ ein Buchstabe ist bzw. mit ♀ beschrieben wird, falls $\varphi_m'((a_1 \cdot \ldots \cdot a_{i-1})^{(r)}, a_i, (a_{i+1} \cdot \ldots \cdot a_n)^{(l)}) = e$. Ist der Wortanfang erreicht, so wird, falls definiert, $\varphi_l(e, a_1 \cdot \ldots \cdot a_n)$ auf das Band geschrieben. Ist irgendeiner der benötigten Funktionswerte nicht definiert, so läuft $\mathfrak{M}$ ad infinitum nach links.

Es ist klar, daß $\mathfrak{M}$ die Funktion φ mit beschränktem Regime berechnet. Damit ist Satz 3 bewiesen.

Aus dem Beweis des Satzes 3 ergibt sich die

Folgerung 2. Ist die partielle Wortfunktion φ über $[X, Y]$ mit beschränktem Regime berechenbar, so gibt es eine Turingmaschine, die φ berechnet und deren Lese- und Schreibekopf bei der Bearbeitung eines jeden Wortes aus X^* nur einmal die Bewegungsrichtung ändert (Bewegung des Kopfes zuerst von links nach rechts).

Aus Folgerung 2 ergibt sich eine andere Charakterisierung der in linearer Zeit berechenbaren partiellen Wortfunktionen. Zunächst eine Definition.

Eine partielle Wortfunktion φ heißt genau dann einseitig berechenbar, wenn es eine Turingmaschine gibt, die φ berechnet und deren Lese- und Schreibekopf sich in jedem Arbeitstakt um ein Feld nach rechts bewegt. Solche einseitig berechenbaren Funktionen wurden von Modrow in [2] untersucht. Setzt man noch voraus, daß diese Funktionen überall definiert sind, so sind sie fastsequentiell im Sinne von Wechsung ([3]).

Satz 4. Eine partielle Wortfunktion φ ist genau dann mit beschränktem Regime berechenbar, wenn es zwei einseitig berechenbare Wortfunktionen φ_1 und φ_2 gibt mit $\varphi = J \circ \varphi_2 \circ J \circ \varphi_1$.

Dieser Satz ergibt sich aus Folgerung 2.

Den Inhalt der Sätze 1—4 fassen wir zusammen.

Satz 5. Für eine partielle Wortfunktion φ über $[X, Y]$ sind folgende Aussagen äquivalent:

a) φ ist in einer Zeit $T(n) = o(n \cdot \log n)$ berechenbar.
b) φ ist in linearer Zeit berechenbar.
c) φ ist mit beschränktem Regime berechenbar.
d) Es existieren zwei einseitig berechenbare Funktionen φ_1 und φ_2 mit $\varphi = J \circ \varphi_2 \circ J \circ \varphi_1$.
e) Es existieren zwei zweistellige partielle Wortfunktionen φ_l und φ_r mit

1) Die Werte $\varphi(w \cdot w')$, $\varphi_l(w, w')$ und $\varphi_r(w, w')$ sind alle gleichzeitig oder nicht definiert. Im Falle ihrer Existenz gilt

$$\varphi(w \cdot w') = \varphi_l(w, w') \cdot \varphi_r(w, w').$$

2) Ist $\varphi(waw')$ mit $a \in X$ definiert, so gilt $|\varphi_l(wa, w')| = |\varphi_l(w, aw')| + 1$, falls $\varphi_l(wa, w') \neq e$ und $\varphi_r(w, aw') \neq e$ und es gilt $|\varphi_l(wa, w')| = |\varphi_l(w, aw')|$, sonst.

3) Es sind $G_2(\varphi_l)$ und $G_1(\varphi_r)$ endlich und aus $(\varphi_l)_{2,w} = (\varphi_l)_{2,w'}((\varphi_r)_{1,w} = (\varphi_r)_{1,w'})$ folgt $(\varphi_l)_{2,aw} = (\varphi_l)_{2,aw'}((\varphi_r)_{1,wa} = (\varphi_r)_{1,w'a})$ für alle $a \in X$.

2. In linearer Zeit entscheidbare und aufzählbare Mengen

Wir wollen hier einen Satz von TRACHTENBROT ([1]) verallgemeinern, der besagt, daß die Klasse der mit beschränktem Regime (d.h. in linearer Zeit) entscheidbaren Wortmengen zusammenfällt mit der Klasse der durch überall definierte, in linearer Zeit berechenbare Funktionen aufgezählte Mengen und der Klasse der regulären Mengen.

Satz 6. Folgende Aussagen sind für eine Wortmenge M äquivalent:

a) M ist in linearer Zeit entscheidbar.
b) M ist Wertebereich einer in linearer Zeit berechenbaren partiellen Wortfunktion.
c) M ist Definitionsbereich einer in linearer Zeit berechenbaren partiellen Wortfunktion.
d) M ist regulär.

Beweis: I. Aus *a)* folgt *c)*. Es sei $\mathfrak{M}$ eine Turingmaschine, die M mit beschränktem Regime entscheidet. Nach Beendigung der Arbeit von $\mathfrak{M}$ auf einem Wort $w \in M$ steht auf dem Band ein ausgezeichneter Buchstabe a, in dem Fall $w \notin M$ ist das Band nach der Beendigung der Arbeit von $\mathfrak{M}$ leer. Wir setzen o.B.d.A. voraus, daß im ersten Fall $\mathfrak{M}$ links von a anhält. Wir betrachten eine Turingmaschine $\mathfrak{M}'$, die wie $\mathfrak{M}$ arbeitet, aber im Endzustand von $\mathfrak{M}$ nach rechts läuft, bis ein von $\text{\textdollar}$ verschiedener Buchstabe erreicht wird. Wird ein solcher Buchstabe erreicht, so hält $\mathfrak{M}'$ an, im anderen Falle läuft $\mathfrak{M}'$ ad infinitum nach rechts. Somit ist ein Wort w genau dann in M, wenn die durch $\mathfrak{M}$ (mit beschränktem Regime) berechnete Funktion auf w definiert ist.

II. Aus *c)* folgt *b)*. Es sei $\mathfrak{M}$ eine Turingmaschine, die die Funktion φ mit beschränktem Regime berechnet. Wir konstruieren eine Turingmaschine $\mathfrak{M}'$, die eine Funktion φ' mit beschränktem Regime berechnet, so daß $D_\varphi = R_{\varphi'}$ gilt.

Die Turingmaschine $\mathfrak{M}'$ arbeitet wie $\mathfrak{M}$, nur daß sie anstelle des Buchstaben y das Zeichen (x, y) druckt, wobei x der urspüngliche Inhalt des betreffenden Feldes ist. Wird der Endzustand von $\mathfrak{M}$ erreicht, so wird überall das Zeichen (x, y) durch x ersetzt. Hier die Befehlsliste von $\mathfrak{M}'$:

Ist $zx \to z'y\sigma$ ($\sigma \in \{+1, -1\}$) ein Befehl von $\mathfrak{M}$, so sind

$zx \to z'(x, y)\sigma$ und

$z(x', x) \to z'(x', y)\sigma$ (x' ist aus dem Arbeitsalphabet von $\mathfrak{M}$).

Befehle von $\mathfrak{M}'$.

Ist z_0 der Endzustand von $\mathfrak{M}$, so sind

$$z_0(x, y) \rightarrow z_0(x, y) - 1 \quad (x \text{ und } y \text{ sind aus dem Arbeitsalphabet von } \mathfrak{M}),$$

$$z_0 x \rightarrow z_0' x + 1,$$

$$z_0'(x, y) \rightarrow z_0' x + 1 \qquad \text{und}$$

$$z_0' x \rightarrow s_0 x + 1$$

Befehle von $\mathfrak{M}'$. Dabei sind z_0' und s_0 neue Zustände, s_0 ist der Endzustand von $\mathfrak{M}'$. Es ist evident, das $\mathfrak{M}'$ das Gewünschte leistet.

III. Aus *b)* folgt *d)*. Es sei M der Definitionsbereich einer in linearer Zeit berechenbaren partiellen Wortfunktion φ über $[X, Y]$. Nach Satz 4 gibt es zwei zweistellige einseitig berechenbare partielle Wortfunktionen φ_1 und φ_2 mit $\varphi = J \circ \varphi_2 \circ J \circ \varphi_1$. Dann gibt es zwei partielle sequentielle Funktionen ψ_1 und ψ_2 mit endlichem Gewicht und $\varphi_1(w) = \psi_1(w*)$ sowie $\varphi_2(w) = \psi_2(w*)$ für alle $w \in X^*$ und einem geeigneten Buchstaben $* \notin X$. Die Menge $X^* \cdot \{*\} =_{\mathrm{Df}} =_{\mathrm{Df}} \{w*\,;\, w \in X^*\}$ ist regulär. Dann ist auch die Menge $\psi_1(X^* \cdot \{*\}) = \varphi_1(X^*)$ regulär. Die Menge $M_1 =_{\mathrm{Df}} J(\varphi_1(X^*))$ ist regulär und somit auch die Menge $M_1 \cdot \{*\} =_{\mathrm{Df}} \{w*\,;\, w \in M_1\}$. Dann ist aber die Menge $\psi_2(M_1 \cdot \{*\}) = \varphi_2(M_1)$ ebenfalls regulär. Mithin ist $M = J(\varphi_2(J(\varphi_1(X^*)))) = J(\varphi_2(M_1))$ regulär.

IV. Die Aussage *a)* folgt trivialerweise aus *d)*. Damit ist Satz 6 bewiesen.

Zum Schluß sei Herrn Doz. Dr. Wechsung, Jena, für die wertvollen Gespräche und Hinweise zu diesen Problemen gedankt.

Kurzfassung

Das Trachtenbrotsche Ergebnis, daß jede Turingmaschine, die eine überall definierte Wortfunktion in einer Zeit $T(n) = o(n \cdot \log n)$ berechnet, diese Funktion bereits in linearer Zeit und mit beschränktem Regime berechnet, wird auf den Fall partieller Wortfunktionen verallgemeinert, und es wird eine „maschinenfreie" Charakterisierung der in linearer Zeit berechenbaren partiellen Wortfunktionen angegeben. Wie im Fall der überall definierten Wortfunktionen gilt auch für die in linearer Zeit berechenbaren partiellen Wortfunktionen, daß die Klasse der Wertebereiche dieser Funktionen mit der Klasse der regulären Mengen zusammenfällt, und außerdem ist diese identisch mit der Klasse der Definitionsbereiche dieser Funktionen.

SEKTION MATHEMATIK DER
FRIEDRICH-SCHILLER-UNIVERSITÄT
69 JENA — DDR
UNIVERSITÄTSHOCHHAUS

Literatur

[1] Трахтенброт, Б. А., Сложность алгоритмов и вычислений, Новосибирск 1967.
[2] Modrow, H. D., Eine Erweiterung der Theorie der determinierten abstrakten Automaten auf nicht-sequentielle Wortfunktionen., Diss. A, Humboldt-Universität Berlin, 1969.
[3] Wechsung, G., Quasisequentielle Funktionen, *Acta Cybernet.*, Bd. 2, Heft 1, 1973, S. 24—33.

(Eingegangen am 30. Juli 1973)