7.3 Komplexitätsmaße für Schaltnetze

Es ist intuitiv klar und auch leicht zu zeigen, dass sich jeder Algorithmus durch ein Schaltnetz beschreiben lässt; genau nach diesem Prinzip der bitweisen Operationen arbeiten ja Computer auf der untersten Ebene – Das Schaltnetz ist also ein universelles Maschinenmodell. Klar ist auch, dass man mit einem solchen simplen Modell nicht direkt genaue Aufwandsberechnungen für reale Computer durchführen kann; wohl aber ist die Unterscheidung, ob ein Algorithmus effizient, also mit polynomialem Zeitaufwand, arbeitet oder nicht, damit sehr gut möglich. Dazu betrachtet man die beiden Werte $\#C$, die **Größe** des Schaltnetzes, also die Anzahl der Knoten, und $t(C)$, die **Tiefe**, also die größte Weglänge innerhalb des Schaltnetzes. Die Größe ist ein Komplexitätsmaß für die serielle Abarbeitung, die Tiefe eines für die unbeschränkt parallele Abarbeitung des Algorithmus. **Beispiele**

1. Das Schaltnetz für die Addition zweier Bits, Abbildung 1 hat die Größe 4 und die Tiefe 1.

2. Das Schaltnetz für die Addition dreier Bits, Abbildung 2 hat die Größe 10 und die Tiefe 3.

3. Aus dem obigen Beispiel 3 erhält man ein einfaches (längst nicht optimales) Schaltnetz zur Addition von s Einbit-Zahlen mit $A(s)$ Knoten, darunter $a(s)$ Ausgängen. Nach Beispiel 1 ist $A(2) = 4$, $a(2) = 2$, nach Beispiel 3 gilt die Rekursion $a(s) = a(s-1) + 1$ und $A(s) = A(s-1) + 2a(s-1) + 1$. Daraus folgt $a(s) = s$ und $A(s) = s^2$ durch Induktion.

4. Das Schaltnetz für die Multiplikation zweier Zweibitzahlen, Abbildung 3 hat die Größe 12 und die Tiefe 3.

5. Beim Schaltnetz für die Verzweigung, Abbildung 4 gelten die Formeln

$$\begin{aligned} \#C &= \#C_1 + \#C_2 + \#E - 2r + 3s + 2, \\ t(C) &= \max\{t(C_0) + 2, t(C_1) + 2, t(E) + 3\}. \end{aligned}$$

6. Für die Größe $V(n)$ des Schaltnetzes zum Vergleich in Beispiel 6 gilt die Rekursionsformel $V(n) = V(n-1) + 11$ mit V(2) = 6, also $V(n) = 11n - 5$.

7. Die Aufwandsabschätzungen für die Ganzzahl-Operationen mit der Basis $B = 2$ des Zahlensystems besagen, dass Paare von n-Bitzahlen mit Schaltnetzen der Größe $O(n)$ addiert und subtrahiert und mit Schaltnetzen der Größe $O(n^2)$ multipliziert und dividiert werden können.

Anmerkungen

Bei der Definition von Schaltnetzen wird manchmal der Innengrad nicht durch 2 beschränkt. Diese Beschränkung ist aber durchaus sinnvoll, da reale Maschinen in jedem Schritt nur eine bestimmte Anzahl von Bits verarbeiten können. Ob man die Schranke auf 2 festsetzt oder höher, spielt für die folgenden Komplexitätsüberlegungen allerdings keine Rolle.

Zwischenergebnisse können in einem Schaltnetz wegen des unbeschränkten Außengrades beliebig oft verwendet werden; das entspricht einem unbeschränktem Speicher auf einem realen Computer. Ein engerer Begriff, bei dem dies vermieden wird, ist die BOOLEsche Formel: Eine solche ist ein Schaltnetz, dessen Graph ein Baum ist. Das bedeutet, dass der Außengrad jedes Knotens 1 ist (außer den Ausgängen mit Außengrad 0). Ein solches Schaltnetz entspricht genau dem Aufbau einer ausgeschriebenen Formel aus zweistelligen Operationen, wo ja auch jedes Zwischenergebnis nur einmal verwendet werden kann. Insbesondere muss jede Eingabe so oft wiederholt werden, wie sie gebraucht wird.

Bei einer weiteren allgemeineren Definition von Schaltnetzen werden auch beliebige zweistellige Bitoperationen statt nur $\oplus$ (XOR) und $\otimes$ (AND) zugelassen. Tabelle 1 zeigt auch, wie sich alle solchen durch $\oplus$ und $\otimes$ ausdrücken lassen. Diese allgemeinere Definition gestattet in der Regel etwas kürzere Schaltnetze, hat aber keine lohnenden Auswirkungen auf die Komplexitätsüberlegungen.

Schaltnetze (oder Algorithmen) dienen nicht nur zur Berechnung von Funktionen, sondern auch zum Finden von Lösungen, formalisiert durch das Erfüllen einer Relation. Seien $X \subseteq \mathbb{F}_2^r$ und $Y \subseteq \mathbb{F}_2^s$ zwei Mengen und E eine Relation auf $X \times Y$, beschrieben durch eine Funktion

$$E : X \times Y \longrightarrow \mathbb{F}_2.$$

Gefragt sind Algorithmen, die zu gegebenem $x \in X$ ein $y \in Y$ finden mit $E(x, y) = 1$. Den bisher behandelten Spezialfall einer auszuwertenden Funktion $f : X \longrightarrow Y$ findet man als rechtseindeutige Relation wieder – $E(x, y) = 1 \Longleftrightarrow y = f(x)$. Man erfasst aber auch das Lösen von Gleichungen, etwa von linearen Gleichungssystemen: $X = \mathbf{M}_{m,n}(\mathbb{F}_2)$ = die Menge der $m \times n$-Matrizen, $Y = \mathbb{F}_2^m \times \mathbb{F}_2^n$, $E(x, y) = 1 \Longleftrightarrow xy_1 = y_2$ (als Produkt Matrix $\times$ Vektor).

Ein Schaltnetz $C : \mathbb{F}_2^r \longrightarrow \mathbb{F}_2^s$ erfüllt E, wenn $C(X) \subseteq Y$ und $E(x, C(x)) = 1$ für alle $x \in X$.