# Platzkomplexität

Mareike Ziese

18.05.2010

## 1 Einführung in die Komplexitätstheorie

### 1.1 Komplexitätsschranken

Unter einer **Komplexitätsschranke** versteht man eine Funktion $T : \mathbb{N} \to \mathbb{N}$. Von besonderem Interesse sind hierbei zB. die folgende:

$$\begin{array}{rll}
\textbf{CON} & := O(1) & \text{konstant,} \\
\textbf{Lin(T)} & := \Theta(T) & \text{linear,} \\
, \textbf{Pol(T)} & := \bigcup_{k \in \mathbb{N}} O(T^k) & \text{polynomial,} \\
\textbf{Log(T)} & := \Theta(log(T)) & \text{logarithmisch,} \\
\textbf{LLog(T)} & := \Theta(log(T)) & \text{zweifach logarithmisch,} \\
\textbf{PLog(T)} & := Pol(log(T)) & \text{polynomial logarithmisch,} \\
\textbf{ExL(T)} & := exp(\Theta(T)) & \text{exponentiell linear,} \\
\textbf{ExP(T)} & := exp(Pol(T)) & \text{exponentiell polynomial,} \\
\textbf{EExL(T)} & := eexp(\Theta(T)) & \text{doppelt exponentiell linear.}
\end{array}$$

### 1.2 Komplexitätsklassen

#### 1.2.1 Allgemeine Einführung

$\Sigma = \Sigma_E$ sei ein festes endliches Alphabet, G ein Speicher. $\mathcal{M}_\mathcal{G}$ bezeichne die Menge aller TM-Akzeptoren mit Speicher G. Dann definieren Komplexitätsschranken T und S die folgenden deterministischen **Komplexitätsklassen** :

$$\begin{array}{rcl}
DTime_G(T) & := & \{L \subseteq \Sigma^* \mid \exists \text{ T-zeitb. DTM } M \in \mathcal{M}_\mathcal{G} \text{ mit } L(M) = L\} \\
DSpace_G(S) & := & \{L \subseteq \Sigma^* \mid \exists \text{ S-platzb. DTM } M \in \mathcal{M}_\mathcal{G} \text{ mit } L(M) = L\} \\
DTimeSpace_G(T, S) & := & \{L \subseteq \Sigma^* \mid \exists \text{ T-zeit- und S-platzb. DTM } M \in \mathcal{M}_\mathcal{G} \text{ mit } L(M) = L\}
\end{array}$$

Die entsprechenden nichtdeterministischen Klassen definiert mit Hilfe von NTM, werden mit $NTime_G(T)$ , $NSpace_G(S)$ und $NTimeSpace_G(T)$ bezeichnet.

### 1.2.2 Übersicht der wichtigsten Klomplexitätsklassen

Um die Schreibweise zu verkürzen, führen wir spezielle Bezeichnungen für die wichtigsten Komplexitätsklassen ein. Diese Bezeichnungen sind allgemein aus Vorlesungen wie GTI und Höhere Algorithmik bekannt :

$$
\begin{array}{rcl}
\mathcal{L} & := & \text{DSpace(LOG)} \\
\mathcal{NL} & := & \text{NSpace(LOG)} \\
\mathcal{P} & := & \text{DTime(POL)} \\
\mathcal{NP} & := & \text{DTime(POL)} \\
\mathcal{PSPACE} & := & \text{DSpace(POL)} \\
\mathcal{DEXP} & := & \text{DTime(EXP)} \\
\mathcal{NEXP} & := & \text{NTime(EXP)} \\
\mathcal{EXPSPACE} & := & \text{DSpace(EXP)}
\end{array}
$$

## 1.3 Reduktion, Vollständigkeit

### 1.3.1 Polynomial-Zeit Reduktion

Es seien $L_1 \subseteq \Sigma_1^*, L_2 \subseteq \Sigma_2^*$ Sprachen. $L_1$ heißt **polynomialzeit-reduzierbar** auf $L_2$, geschrieben $L_1 \leq_{pol} L_2$ , falls es eine in polynomialer Zeit berechenbare Funktion $f$ : $\Sigma_1^* \to \Sigma_2^*$ gibt, sodass für alle $X \in \Sigma_1^*$ gilt:

$X \in L_1 \iff f(X) \in L_2$

### 1.3.2 Logspace Reduktion

Es seien $L_1 \subseteq \Sigma_1^*, L_2 \subseteq \Sigma_2^*$ Sprachen. $L_1$ heißt **logspace-reduzierbar** auf $L_2$, geschrieben $L_1 \leq_{log} L_2$ , falls die Reduktion $f$ durch eine Maschine in $\mathcal{L} = DSpace(LOG)$ berechnet werden kann.

**Lemma:** Logspace-Reduzierbarkeit impliziert Polynomialzeit-Reduzierbarkeit.
Die Reduzierbarkeitsrelationen sind *transitiv*

### 1.3.3 Vollständigkeit

$\mathcal{C}$ sei eine Menge von Sprachen. Eine Sprache L heißt **hart** für $\mathcal{C}$ bzgl. einer Reudzierbarkeit, falls sich jede Sprache $L' \in \mathcal{C}$ auf $L$ reduzieren lässt.

$L$ heißt **vollständig** für $\mathcal{C}$ bzgl. einer Reduzierbarkeit, wenn sie hart für $\mathcal{C}$ ist und zusätzlich $L \in \mathcal{C}$ gilt.

$\mathcal{NL}$-**vollständig** und $\mathcal{P}$-**vollständig** bedeute vollständig für $\mathcal{NL}$ bzw. $\mathcal{P}$ bezgl. logspace-Reduzierbarkeit.

$\mathcal{NP}$-**vollständig** und $\mathcal{PSPACE}$-**vollständig** bedeute vollständig für $\mathcal{NP}$ bzw. $\mathcal{PSPACE}$ bezgl. Polynomialzeit Reduktion.

# 2 Platzkomplexitätsklassen - von klein nach groß

Für alle Platzklassen mit Platzschranke $S(n), S \geq log(n)$ gilt:
$NSPACE(S) = Co - NSPACE(S)$

## 2.1 $\mathcal{L}$ und $\mathcal{NL}$

Die Klasse $\mathcal{L}$ ( bzw. $\mathcal{NL}$) beschreibt alle Sprachen, die auf einer DTM (NTM) mit logarithmisch viel Platz akzeptiert werden können. Dazu muss gesagt werden, dass bei Komplexitätsklassen mit weniger als linearem Speicherplatz das Arbeitsband unabhängig von der Größe des Eingabebandes betrachtet wird (anders wäre logarithmischer Speicherplatz nicht möglich, weil schon die Eingabe linear ist.)

Es gilt $\mathcal{L} \subseteq \mathcal{NL}$

## 2.2 $\mathcal{PSPACE}$

**Satz von Savitch**(1970):
Sei $L \in NSPACE(S(n)), S(n) \geq logn\ S(n)$ in $(S(n))^2$ berechenbar. Dann ist
$L \in DSPACE((S(N))^2)$

Durch den Satz von Savitch konnte insbesondere bewiesen werden, dass
$\mathcal{PSPACE} = \mathcal{NPSPACE}$

Es gilt
$\mathcal{P} \subseteq \mathcal{PSPACE}$
$\mathcal{NP} \subseteq \mathcal{PSPACE}$

## 2.3 $\mathcal{EXPSPACE}$

Auch auf $\mathcal{EXPSPACE}$ herrscht Vollstänigkeit bzgl. Polynomialzeit-Reduktion. Ein $\mathcal{EXPSPACE}$-vollständiges Problem ist zB folgendes:

**Eingabe:** Zwei reguläre Ausdrücke mit Iterationen(neben Vereinigung, Konkatenation, Kleene-Stern)
(Wenn $E$ ein regulärer Ausdruck ist, dann auch $E^k$ $(k \in \mathbb{N})$)

**Frage:** Sind beide Ausdrücke äquivalent?

Es gilt
$\mathcal{L} \subset \mathcal{PSPACE} \subset \mathcal{EXPSPACE}$

# 3 Hierarchien zwischen den Klassen

$\mathcal{L} \subseteq \mathcal{NL} \subseteq \mathcal{P} \subseteq \mathcal{NP}/\mathcal{CO} - \mathcal{NP} \subseteq \mathcal{PSPACE} \subseteq \mathcal{DEXP} \subseteq \mathcal{NEXP} \subseteq \mathcal{EXPSPACE}$

Es gilt
$\mathcal{P} \subset \mathcal{DEXP}$
$\mathcal{L} \subset \mathcal{PSPACE} \subset \mathcal{EXPSPACE}$

Man vermutet, dass alle Komplexitätsklassen voneinander verschieden sind; aber bislang konnte dies noch für kein Paar gezeigt werden. Auch nicht es bis heute noch nicht gelungen, obere Schranken entscheidend zu verbessern. Die exponentielle Vergrößerung der Ressourcenschranke beim Übergang von nichtdeteministischer auf deterministischer Zeit sowie beim Übergang von Platz auf Zeit

$DSPACE(S) \subseteq DTIME(ExL(S))$

kann durch die bisher zur Verfügung stehenden Simulationstechniken bei den Beziehungen

$\mathcal{P} \subseteq \mathcal{NP} \subseteq \mathcal{DEXP}$ bzw. $\mathcal{P} \subseteq \mathcal{PSPACE} \subseteq \mathcal{DEXP}$

nicht vermieden werden.Nur in der Sequenz

$\mathcal{L} \subseteq \mathcal{P} \subseteq \mathcal{PSPACE}$

konnte man mit erheblichen Beweisaufwand die trivialen Inklusionen $DTIME(T) \subseteq DSPACE(T)$ um einen logarithmischen Faktor verbessern zu:

$DTIME(T) \subseteq DSPACE(T/logT)$

Quellen:
Komplexitätstheorie Band 1: Grundlagen, K.Rüdiger Reischuk, B.G.Teubner Stuttgart/Leipzig
Theoretische Informatik 2, B.Beckert
Theoretische Informatik, Ernst W. Mayr