Theoretische Informatik 2

Johannes Köbler

Institut für Informatik
Humboldt-Universität zu Berlin

WS 2009/10

Zeitkomplexität von Turingmaschinen

Die Laufzeit einer NTM M bei Eingabe x ist die maximale Anzahl an Rechenschritten, die $M(x)$ ausführt.

Definition

- Die Laufzeit einer NTM M bei Eingabe x ist definiert als

  $$time_M(x) = \max\{t \geq 0 \mid \exists K : K_x \vdash^t K\},$$

  wobei $\max \mathbb{N} = \infty$ ist.
- Sei $t : \mathbb{N} \to \mathbb{N}$ eine monoton wachsende Funktion.
- Dann ist M $t(n)$-zeitbeschränkt, falls für alle Eingaben x gilt:

  $$\mathrm{time}_M(x) \leq t(|x|).$$

Die Zeitschranke $t(n)$ beschränkt also die Laufzeit bei allen Eingaben der Länge n (worst-case Komplexität).

Zeitkomplexitätsklassen

Wir fassen alle Sprachen und Funktionen, die in einer vorgegebenen Zeitschranke $t(n)$ entscheidbar bzw. berechenbar sind, in folgenden Komplexitätsklassen zusammen.

Definition

- Die in deterministischer Zeit $t(n)$ entscheidbaren Sprachen bilden die Sprachklasse

$$\mathrm{DTIME}(t(n)) = \{L(M) \mid M \text{ ist eine } t(n)\text{-zeitbeschränkte DTM}\}.$$

- Die in nichtdeterministischer Zeit $t(n)$ entscheidbaren Sprachen bilden die Sprachklasse

$$\mathrm{NTIME}(t(n)) = \{L(M) \mid M \text{ ist eine } t(n)\text{-zeitbeschränkte NTM}\}.$$

- Die in deterministischer Zeit $t(n)$ berechenbaren Funktionen bilden die Funktionenklasse

$$\mathrm{FTIME}(t(n)) = \left\{ f \;\middle|\; \begin{array}{l} \text{es gibt eine } t(n)\text{-zeitbeschränkte} \\ \text{DTM } M \text{, die } f \text{ berechnet} \end{array} \right\}.$$

Die wichtigsten Zeitkomplexitätsklassen

- Die wichtigsten deterministischen Zeitkomplexitätsklassen sind

$$
\begin{aligned}
\mathsf{LINTIME} &= \bigcup_{c\geq 1} \mathsf{DTIME}(cn + c) && \text{„Linearzeit"} \\
\mathsf{P} &= \bigcup_{c\geq 1} \mathsf{DTIME}(n^c + c) && \text{„Polynomialzeit"} \\
\mathsf{E} &= \bigcup_{c\geq 1} \mathsf{DTIME}(2^{cn+c}) && \text{„Lineare Exponentialzeit"} \\
\mathsf{EXP} &= \bigcup_{c\geq 1} \mathsf{DTIME}(2^{n^c+c}) && \text{„Exponentialzeit"}
\end{aligned}
$$

- Die nichtdeterministischen Klassen NLINTIME, NP, NE, NEXP und die Funktionenklassen FP, FE, FEXP sind analog definiert.
- Für eine Klasse $\mathcal{F}$ von Funktionen sei $\mathsf{DTIME}(\mathcal{F}) = \bigcup_{t\in\mathcal{F}} \mathsf{DTIME}(t(n))$ (die Klassen $\mathsf{NTIME}(\mathcal{F})$ und $\mathsf{FTIME}(\mathcal{F})$ seien analog definiert).

Asymptotische Laufzeit und Landau-Notation

Definition

Seien f und g Funktionen von $\mathbb{N}$ nach $\mathbb{R}^+$.

- Wir schreiben $f(n) = O(g(n))$, falls es Zahlen n_0 und c gibt mit

  $$\forall n \geq n_0 : f(n) \leq c \cdot g(n).$$

  Bedeutung: „f wächst nicht wesentlich schneller als g."
- Formal bezeichnet der Term $O(g(n))$ die Klasse aller Funktionen f, die obige Bedingung erfüllen.
- Die Gleichung $f(n) = O(g(n))$ drückt also in Wahrheit eine Element-Beziehung $f \in O(g(n))$ aus.
- O-Terme können auch auf der linken Seite vorkommen.
- In diesem Fall wird eine Inklusionsbeziehung ausgedrückt.
- So steht $n^2 + O(n) = O(n^2)$ für die Aussage

  $$\{n^2 + f \mid f \in O(n)\} \subseteq O(n^2).$$

Asymptotische Laufzeit und Landau-Notation

Beispiel

- $7\log(n) + n^3 = O(n^3)$ ist richtig.
- $7\log(n)n^3 = O(n^3)$ ist falsch.
- $2^{n+O(1)} = O(2^n)$ ist richtig.
- $2^{O(n)} = O(2^n)$ ist falsch (siehe Übungen).

Mit der O-Notation lassen sich die wichtigsten deterministischen Zeitkomplexitätsklassen wie folgt charakterisieren:

$$\begin{aligned}
\mathrm{LINTIME} &= \mathrm{DTIME}(O(n)) && \text{„Linearzeit“}\\
\mathrm{P} &= \mathrm{DTIME}(n^{O(1)}) && \text{„Polynomialzeit“}\\
\mathrm{E} &= \mathrm{DTIME}(2^{O(n)}) && \text{„Lineare Exponentialzeit“}\\
\mathrm{EXP} &= \mathrm{DTIME}(2^{n^{O(1)}}) && \text{„Exponentialzeit“}
\end{aligned}$$

Platzkomplexität von Turingmaschinen

- Als nächstes definieren wir den Platzverbrauch von NTMs.
- Intuitiv ist dies die Anzahl aller während einer Rechnung benutzten Bandfelder.
- Wollen wir auch sublinearen Platz sinnvoll definieren, so dürfen wir hierbei das Eingabeband offensichtlich nicht berücksichtigen.
- Um sicherzustellen, dass eine NTM M das Eingabeband nicht als Speicher benutzt, verlangen wir, dass
  - M die Felder auf dem Eingabeband nicht verändert und
  - sich nicht mehr als ein Feld von der Eingabe entfernt.

Definition

Eine NTM M heißt offline-NTM (oder NTM mit Eingabeband), falls für jede von M bei Eingabe x erreichbare Konfiguration

$$K = (q, u_1, a_1, v_1, \ldots, u_k, a_k, v_k)$$

gilt, dass $u_1 a_1 v_1$ ein Teilwort von $\sqcup x \sqcup$ ist.

Platzkomplexität von Turingmaschinen

Definition

- Der Platzverbrauch einer offline-NTM M bei Eingabe x ist definiert als

$$\mathit{space}_M(x) = \max\left\{ s \geq 1 \;\middle|\; \begin{array}{l} \exists K = (q, u_1, a_1, v_1, \ldots, u_k, a_k, v_k) \\ \text{mit } K_x \vdash^* K \text{ und } s = \sum_{i=2}^{k} |u_i a_i v_i| \end{array} \right\}.$$

- Sei $s : \mathbb{N} \to \mathbb{N}$ eine monoton wachsende Funktion.
- M heißt $s(n)$-platzbeschränkt, falls für alle Eingaben x gilt:

$$\mathit{space}_M(x) \leq s(|x|).$$

Platzkomplexitätsklassen

Wir fassen alle Sprachen, die in einer vorgegebenen Platzschranke $s(n)$ entscheidbar sind, in folgenden Platzkomplexitätsklassen zusammen.

Definition

- Die auf deterministischem Platz $s(n)$ entscheidbaren Sprachen bilden die Klasse

  $$\text{DSPACE}(s(n)) = \{L(M) \mid M \text{ ist eine } s(n)\text{-platzb. offline-DTM}\}.$$

- Die auf nichtdeterministischem Platz $s(n)$ entscheidbaren Sprachen bilden die Klasse

  $$\text{NSPACE}(s(n)) = \{L(M) \mid M \text{ ist eine } s(n)\text{-platzb. offline-NTM}\}.$$

Die wichtigsten Platzkomplexitätsklassen

- Die wichtigsten deterministischen Platzkomplexitätsklassen sind

$$\begin{aligned} \mathsf{L} &= \mathsf{DSPACE}(O(\log n)) && \text{„Logarithmischer Platz"} \\ \mathsf{LINSPACE} &= \mathsf{DSPACE}(O(n)) && \text{„Linearer Platz"} \\ \mathsf{PSPACE} &= \mathsf{DSPACE}(n^{O(1)}) && \text{„Polynomieller Platz"} \end{aligned}$$

- Die nichtdeterministischen Klassen NL, NLINSPACE und NPSPACE sind analog definiert.
- Der Satz von Savitch besagt, dass

  $$\mathsf{NSPACE}(s(n)) \subseteq \mathsf{DSPACE}(s^2(n))$$

  ist (dies wird in der VL Komplexitätstheorie gezeigt).

  Daher fallen die Klassen NPSPACE und PSPACE zusammen.

Die wichtigsten Zeit- und Platzkomplexitätsklassen

Die wichtigsten Zeitkomplexitätsklassen sind

$$\begin{aligned} \mathrm{LINTIME} &= \mathrm{DTIME}(O(n)) && \text{„Linearzeit“} \\ \mathrm{P} &= \mathrm{DTIME}(n^{O(1)}) && \text{„Polynomialzeit“} \\ \mathrm{NP} &= \mathrm{NTIME}(n^{O(1)}) && \text{„Nichtdet. Polynomialzeit“} \\ \mathrm{E} &= \mathrm{DTIME}(2^{O(n)}) && \text{„Lineare Exponentialzeit“} \\ \mathrm{EXP} &= \mathrm{DTIME}(2^{n^{O(1)}}) && \text{„Exponentialzeit“} \end{aligned}$$

Die wichtigsten Platzkomplexitätsklassen sind

$$\begin{aligned} \mathrm{L} &= \mathrm{DSPACE}(O(\log n)) && \text{„Logarithmischer Platz“} \\ \mathrm{NL} &= \mathrm{NSPACE}(O(\log n)) && \text{„Nichtdet. logarithmischer Platz“} \\ \mathrm{LINSPACE} &= \mathrm{DSPACE}(O(n)) && \text{„Linearer Platz“} \\ \mathrm{PSPACE} &= \mathrm{DSPACE}(n^{O(1)}) && \text{„Polynomieller Platz“} \\ \mathrm{EXPSPACE} &= \mathrm{DSPACE}(2^{n^{O(1)}}) && \text{„Exponentialer Platz“} \end{aligned}$$

Elementare Beziehungen zwischen Komplexitätsklassen

Frage

Welche elementaren Beziehungen gelten zwischen den verschiedenen Zeit- und Platzklassen?

Satz

- Für jede Funktion $s(n) \geq \log n$ gilt
$$\mathrm{DSPACE}(s) \subseteq \mathrm{NSPACE}(s) \subseteq \mathrm{DTIME}(2^{O(s)}).$$
- Für jede Funktion $t(n) \geq n + 2$ gilt
$$\mathrm{DTIME}(t) \subseteq \mathrm{NTIME}(t) \subseteq \mathrm{DSPACE}(t).$$

Korollar

Es gilt $\mathrm{L} \subseteq \mathrm{NL} \subseteq \mathrm{P} \subseteq \mathrm{NP} \subseteq \mathrm{PSPACE} \subseteq \mathrm{EXP} \subseteq \mathrm{EXPSPACE}$.

Komplexitätsschranken für die Stufen der Chomsky-Hierarchie

$$\begin{aligned}
\mathsf{REG} &= \mathsf{DSPACE}(O(1)) = \mathsf{NSPACE}(O(1)) \subsetneq \mathsf{L},\\
\mathsf{DCFL} &\subsetneq \mathsf{LINTIME},\\
\mathsf{CFL} &\subsetneq \mathsf{NLINTIME} \cap \mathsf{DTIME}(O(n^3)) \subsetneq \mathsf{P},\\
\mathsf{DCSL} &= \mathsf{LINSPACE} \subseteq \mathsf{CSL},\\
\mathsf{CSL} &= \mathsf{NLINSPACE} \subseteq \mathsf{PSPACE} \cap \mathsf{E},\\
\mathsf{REC} &= \textstyle\bigcup_f \mathsf{DSPACE}(f(n))\\
&= \textstyle\bigcup_f \mathsf{NSPACE}(f(n))\\
&= \textstyle\bigcup_f \mathsf{DTIME}(f(n))\\
&= \textstyle\bigcup_f \mathsf{NTIME}(f(n)),
\end{aligned}$$

wobei f alle (oder äquivalent: alle berechenbaren) Funktionen $f : \mathbb{N} \to \mathbb{N}$ durchläuft.