

9. Übung Komplexitätstheorie

Abgabe: bis Dienstag, den 13.6. um 12:00 Uhr am Lehrstuhl oder in der Vorlesung

Aufgabe 1

- (a) Zeigen Sie dass das Erreichbarkeitsproblem für gerichtete Graphen sowohl NC^2 als auch in AC^1 ist.

Hinweis: Betrachten Sie die Repräsentation eines Graphen $G = (\{1, \dots, n\}, E)$ durch seine Adjazenzmatrix A mit $a_{ij} = 1$ falls $i = j$ oder $(i, j) \in E$ und $a_{ij} = 0$ sonst, und verwenden Sie die Technik des wiederholten Quadrierens.

- (b) Zeigen Sie, dass für alle $i \geq 2$ die Klassen NC^i (und für alle $i \geq 1$ die Klassen AC^i) unter logspace-Reduktionen abgeschlossen sind.

Hinweis: Zeigen Sie dazu, wie ein (NC^2) -Schaltkreis die bitweise Berechnung der Reduktionsfunktion simulieren kann.

- (c) Zeigen Sie $NLOGSPACE \subseteq AC^1 \subseteq NC^2$.

Aufgabe 2

Für eine Funktion $g: X \times Y \rightarrow Z$ sei die *Kommunikationskomplexität* $C(g)$ die minimale Anzahl von Bits, die Alice und Bob im schlimmsten Fall austauschen müssen, um gemeinsam den Funktionswert $g(x, y) \in Z$ zu berechnen, wobei Alice nur den Teil-Input x und Bob nur den Teil-Input y erhält.

Analog ist die Kommunikationskomplexität für eine Relation $R \subseteq X \times Y \times Z$ definiert, wobei Alice und Bob für gegebene x und y gemeinsam einen Wert $z \in Z$ mit $(x, y, z) \in R$ bestimmen müssen.

Sei $f: \{0, 1\}^n \rightarrow \{0, 1\}$ eine Boolesche Funktion. Wir definieren die Relation $R_f \subseteq f^{-1}(0) \times f^{-1}(1) \times \{1, \dots, n\}$ wie folgt: $(x, y, i) \in R_f$ gdw. $x_i \neq y_i$. Das heißt, Alice bekommt einen Input x mit $f(x) = 0$, Bob bekommt einen Input y mit $f(y) = 1$ ($|x| = |y| = n$). Da $f(x) \neq f(y)$, muss es (mindestens) ein Bit i geben, an dem sich x und y unterscheiden. Die Aufgabe für Alice und Bob besteht nun darin, ein solches i zu finden.

- (a) Zeigen Sie, dass $C(R_{\oplus_n}) \leq 2 \log n$.
- (b) Zeigen Sie, dass für jede Boolesche Funktion f die Kommunikationskomplexität $C(R_f)$ der minimalen Tiefe eines Schaltkreises über $\Omega = \{\wedge, \vee\}$ entspricht, der die Funktion f berechnet und für jeden Input x_i auch auf den negierten Input \bar{x}_i zugreifen kann.

Aufgabe 3

Das Problem MAX-3-SAT besteht darin, zu einer gegebenen Menge von Klauseln mit jeweils drei verschiedenen Literalen eine Belegung zu finden, die möglichst viele Klauseln erfüllt. Zeigen Sie, dass dieses Problem in APX liegt, und geben Sie einen möglichst guten Approximationsalgorithmus dafür an.

Aufgabe 4

Zeigen Sie, dass aus $\text{MAX-CLIQUE} \in \text{APX}$ folgen würde, dass $\text{MAX-CLIQUE} \in \text{PTAS}$.

Hinweis: Zu jedem Graph G lässt sich in polynomialer Zeit ein Graph G^2 konstruieren, so dass G genau dann eine Clique der Größe k hat, wenn G^2 eine Clique der Größe k^2 hat.