

Logik und Komplexität

Sommersemester 2023

Übungsblatt 1

Zu bearbeiten bis 2. Mai 2023

Aufgabe 1: (25 Punkte)

Für eine $\text{FO}[\sigma]$ -Formel $\varphi(\bar{x})$ mit k freien Variablen (für $k \geq 1$) und für jede σ -Struktur \mathcal{A} sei $\llbracket \varphi(\bar{x}) \rrbracket^{\mathcal{A}} := \{\bar{a} \in A^k : \mathcal{A} \models \varphi[\bar{a}]\}$. Zeigen Sie, dass es eine Signatur σ gibt, sodass das folgende Problem unentscheidbar ist.

QUERY CONTAINMENT PROBLEM FÜR $\text{FO}[\sigma]$

Eingabe: Eine Zahl $k \geq 1$ und $\text{FO}[\sigma]$ -Formeln φ und ψ mit je k freien Variablen \bar{x} .

Frage: Gilt für alle endlichen σ -Strukturen \mathcal{A} : $\llbracket \varphi(\bar{x}) \rrbracket^{\mathcal{A}} \subseteq \llbracket \psi(\bar{x}) \rrbracket^{\mathcal{A}}$?

Aufgabe 2: (25 Punkte)

Eine Signatur σ nennen wir **binär**, falls jedes Symbol in σ ein Relationssymbol der Stelligkeit 2 ist.

Beweisen Sie folgende Verschärfung des Satzes von Trakhtenbrot:

Es gibt eine endliche, binäre Signatur $\hat{\sigma}$, sodass das endliche Erfüllbarkeitsproblem für $\text{FO}[\hat{\sigma}]$ unentscheidbar ist.

Hinweis: Überlegen Sie sich für beliebige Signaturen σ eine geeignete Repräsentation von σ -Strukturen durch kantengefärbte Graphen, repräsentiert durch Strukturen über einer geeigneten binären Signatur $\hat{\sigma}$. Benutzen Sie die in der Vorlesung für $\sigma := \{<, \text{succ}, 0, B, K, Z\}$ bewiesene Unentscheidbarkeit des endlichen Erfüllbarkeitsproblems für $\text{FO}[\sigma]$ -Sätze, um die Unentscheidbarkeit des endlichen Erfüllbarkeitsproblems für $\text{FO}[\hat{\sigma}]$ -Sätze zu beweisen.

Achtung: Die für die Aufgaben 3 und 4 benötigten Definitionen zu Baumautomaten und regulären Baumsprachen finden Sie auf der Rückseite des Blattes.

Aufgabe 3: (16 Punkte)

Sei $\Sigma := \{a, b\}$ und sei $L \subseteq T_{\Sigma}$ die Baumsprache, die aus allen Σ -Bäumen besteht, in denen jedes Blatt gerade Höhe hat. Hierbei sei die *Höhe* eines Blattes b definiert als die Anzahl der Kanten auf einem gerichteten Pfad von der Wurzel zu b . Zeigen Sie, dass L regulär ist.

— auf der nächsten Seite geht's weiter —

Aufgabe 4:**(17 + 17 Punkte)**

Sei Σ ein endliches Alphabet. Zeigen Sie, dass Folgendes gilt:

- (a) Wenn $L_1 \subseteq T_\Sigma$ und $L_2 \subseteq T_\Sigma$ regulär sind, so ist auch $L_\cup := L_1 \cup L_2$ regulär.
- (b) Wenn $L \subseteq T_\Sigma$ regulär ist, so ist auch die Baumsprache $\bar{L} := \{t \in T_\Sigma : t \notin L\}$ regulär.

Definitionen

Binärbäume. Ein *gewurzelter Baum* G ist ein endlicher gerichteter Graph, der als ungerichteter Graph ein Baum ist (d.h. der ungerichtete Graph, der aus dem gerichteten Graphen G entsteht, indem jede Kante durch eine ungerichtete Kante ersetzt wird, ist ein Baum) und der einen *Wurzelknoten* enthält, von dem aus jeder andere Knoten über einen gerichteten Pfad erreichbar ist. Wenn jeder Knoten von G entweder ein Blatt ist oder genau zwei Kinder besitzt, so heißt G *voll*. Ein voller Baum, dessen Kantenmenge E in $E_1 \dot{\cup} E_2$ partitioniert ist, sodass jeder Knoten, der kein Blatt ist, genau ein Kind in E_1 (sein *erstes Kind*) und ein Kind in E_2 (sein *zweites Kind*) besitzt, heißt *geordneter Binärbaum*.

Σ -Bäume. Sei Σ ein endliches nicht-leeres Alphabet. Ein Σ -Baum $t = (B, \lambda)$ besteht aus einem geordneten Binärbaum B und einer Abbildung λ , die jedem Knoten v von B eine *Beschriftung* $\lambda(v) \in \Sigma$ zuordnet. Die Menge aller Σ -Bäume bezeichnen wir mit T_Σ . Eine *Baumsprache* ist eine Teilmenge L von T_Σ .

Baumautomaten und reguläre Baumsprachen. Analog zu den endlichen Automaten aus der Theorie der formalen Sprachen, definieren wir Automaten, die Σ -Bäume verarbeiten. Ein (bottom-up) *Baumautomat* ist ein Tupel $\mathfrak{A} := (Q, \Sigma, \Delta, F)$, wobei Q eine endliche Menge, Σ ein endliches Alphabet, $F \subseteq Q$ und $\Delta \subseteq (Q^2 \cup \{\perp\}) \times \Sigma \times Q$ eine Relation ist. Q heißt *Zustandsmenge*, F heißt *Menge der akzeptierenden Zustände* und Δ heißt *Überführungsrelation* von \mathfrak{A} . Falls Δ der Graph einer Abbildung ist, die auf ganz $(Q^2 \cup \{\perp\}) \times \Sigma$ definiert ist, so heißt \mathfrak{A} *deterministisch*, ansonsten heißt \mathfrak{A} *nichtdeterministisch*. Intuitiv können wir uns vorstellen, dass \mathfrak{A} einen Σ -Baum t von den Blättern ausgehend verarbeitet und sich dabei auf die Wurzel zubewegt. Sei $V(t)$ die Knotenmenge des Binärbaums von t . Dabei baut \mathfrak{A} eine Funktion $q : V(t) \rightarrow Q$ auf, die als *Lauf von \mathfrak{A} auf t* bezeichnet wird. Wenn \mathfrak{A} deterministisch ist, ist q eindeutig bestimmt; ansonsten kann \mathfrak{A} viele Läufe (oder auch keinen Lauf) auf t haben. Zunächst wird jedem Blatt v mit Beschriftung $a \in \Sigma$ ein Zustand $q(v) \in Q$ mit $(\perp, a, q(v)) \in \Delta$ zugewiesen. Nun bestimmt \mathfrak{A} rekursiv den Zustand $q(v)$ eines mit $a \in \Sigma$ gefärbten Knotens v , der kein Blatt ist, aus den Zuständen seines ersten und zweiten Kindes u_1 und u_2 . Dazu weist \mathfrak{A} dem Knoten v einen Zustand $q(v) \in Q$ mit $(q(u_1), q(u_2), a, q(v)) \in \Delta$ zu. Wenn schließlich alle Knoten von t durch q mit Zuständen markiert sind, prüft \mathfrak{A} , ob der Zustand $q(w)$ an der Wurzel w von t zur Menge F gehört. Falls ja, so ist q ein *akzeptierender Lauf*, ansonsten ist q ein *verwerfender Lauf*. Ein Σ -Baum t wird von \mathfrak{A} genau dann *akzeptiert*, wenn es mindestens einen akzeptierenden Lauf von \mathfrak{A} auf t gibt. Die vom Baumautomat \mathfrak{A} *erkannte* Baumsprache $L(\mathfrak{A})$ ist die Menge aller Σ -Bäume t , die von \mathfrak{A} akzeptiert werden. Eine Baumsprache L heißt *regulär*, wenn es einen Baumautomaten \mathfrak{A} gibt, der L erkennt.