Theoretische Informatik 2

Aufgabenblatt 4, 2019-06-13

Hausaufgabe 1 [12 PUNKTE]

Vervollständigen Sie den Beweis von Lemma 2.3.11 im neuen Skript:

- [6 PUNKTE] Die Reduktion $G\#s\#t \longmapsto \Phi(G,s,t)$ kann in logarithmischem Platz berechnet werden. [Verwenden Sie die Codierungen für Graphen bzw. Formeln in 2-KNF aus dem Skript.]
- [6 PUNKTE] $\Phi(G,s,t)$ ist genau dann erfüllbar, wenn in G kein Pfad von s nach t existiert.

Lösungsvorschlag:

- Codierung von Graphen: binäre Konotenzahl gefolgt von $\#\#$ und den Adjazenzlisten der einzelnen Knoten (durch $\#$ getrennt)

  Codierung von Formeln in KNF: binäre Atomzahl, gefolgt von $\#\#\#$ und den durch $\#\#$ getrennten Klauseln in Form von Listen von Literalen; Atom p_i als $\boldsymbol{bin}(2i)$ codiert, und Literal $\neg p_i$ als $\boldsymbol{bin}(2i+1)$ (statt roter und blauer Farbe).

  Das die Graphknoten die Atome der Formel sind, braucht die erste Binärzahl nicht verändert zu werden.

  Inspektion der Adjazenzliste $\boldsymbol{ad}(i)$ liefert eine 2-Klausel $\{2i+1, 2j\}$, sofern j in $\boldsymbol{ad}(i)$ auftritt. Der Rechenaufwand beschränkt sich auf $\boldsymbol{bin}(i) \longmapsto \boldsymbol{bin}(i)1$ bzw. $\boldsymbol{bin}(j) \longmapsto \boldsymbol{bin}(j)0$, was keinen Arbeitsplatz erfordert. (Das Schreiben von $\Phi(G,s,t)$ braucht lineare Zeit.)

- ($\Rightarrow$) $\Phi(G,s,t)$ sei erfüllbar, etwa durch φ. Speziell gilt $\varphi(s) = 1$ und $\varphi(t) = 0$. Nach Konstruktion führen von mit 1 bewerteten Knoten nur Kanten zu ebenfalls mit 1 bewerteten Knoten. Damit ist t von s aus nicht erreichbar.

  ($\Leftarrow$) Ist t von s aus nicht erreichbar, so setzen wir für alle von s aus erreichbaren Knoten u (=Atome) $\varphi(u) = 1$, und alle übrigen Knoten $\varphi(v) = 0$; das schließt insbesondere t mit ein. Alle Klauseln $\{\neg p_i, p_j\}$, die einer Kante $\langle i, j\rangle$ entsprechen, die auf einem Pfad liegt, der von s irgendwohin führt, sind damit wahr, weil p_j wahr ist. Ist umgekehrt $\{\neg p_i, p_j\}$ eine von der Kante $\langle i, j\rangle$ induzierte Klausel, bei der i von s aus nicht erreichbar ist, so ist p_i falsch und die Klausel damit wahr. Daraus schließen wir für jede Kante $\langle u, v\rangle$ von G, dass $\varphi(u) \le \varphi(v)$ gilt, die entsprechende Klausel also wahr ist. Wegen $\varphi(s) = 1 = \varphi(\neg t)$ ist damit $\Phi(G,s,t)$ wahr.

Hausaufgabe 2 [15 PUNKTE]

Weisen Sie das folgende Problem als $\boldsymbol{NL}$-vollständig nach:

<u>Azyklizität von Graphen</u> (ACYCLIC)
Gegeben: gerichteter Graph $G = \langle V, E\rangle$ mit $E \subseteq V \times V$
zu entscheiden: ob G ayzklisch ist

[Hinweis: finden Sie eine Reduktion ACPATH $\le^{\log}$ $\overline{\text{ACYCLIC}}$ und wenden Sie den Satz von Immerman und Szelepcsènyi an ($\boldsymbol{NL} = \text{co}\boldsymbol{NL}$)].

Hausaufgabe 3 [24 PUNKTE]

Untersuchen Sie $\boldsymbol{P}$ hinsichtlich folgender Abschlußeigenschaften:

(a) [9 PUNKTE] Durchschnitt, Komplement und Konkatenation;

(b) [4 PUNKTE] Homomorphe Bilder;

(c) [5 PUNKTE] Kleene-Stern;

(d) [6 PUNKTE] Shuffle.

Lösungsvorschlag:

(a) Durchschnitt: Die Sprachen L_i werden von den Maschinen M_i jeweils in polynomialer Zeit akzeptiert. Wir kopieren die Eingabe auf ein zweites Band B_1 und simulieren dann die Maschinen M_i auf Band B_i ; da es sich um Entscheider handelt, ist kein Interleaving nötig. Damit addiert sich im Wesentlichen die Rechenzeit der Maschinen und bleibt somit polynimial.

Komplement: Bei dTMs, die immer halten, können wir einfach q_{acc} mit q_{rej} vertauschen, die Zeitschranke bleibt dabei unverändert.

Konkatenation: die Eingabe $x = x_0x_1 \dots x_{n-1}$ ist systematisch in zwei Teile $u_k = x_0 \dots x_{k-1}$ und $v_k = x_k \dots x_{n-1}$, zu zerlegen für $k < n$. Dann sind u_k mit M_0 und v_k mit M_1 zu bearbeiten, wobei sich die Rechenzeiten addieren. Das ist für $n+1$ Fälle zu wiederholen, also bleibt die Rechenzeit polynomial.

(b) Homomorphe Bilder: Man überzeugt sich leicht (na ja...), dass das Problem CVPVI Boole'scher Schaltkreise mit variablem Input ein homomorphes Bild des Circuit-Value Problems CVP ist; der Homomorphismus bildet einfach (die Codierungen von) 0 und 1 auf (die Codierung von) ? ab, während alles Übrige invariant bleibt. Insbesondere ist dieser Homomorphismus sogar ε-frei. Während CVP in $\boldsymbol{P}$ liegt, ist CVPVI mindestens so schwer wie SAT , also $\boldsymbol{NP}$-hart. In der Tat ist $\boldsymbol{P}$ genau dann unter homomorphen Bildern abgeschlossen, wenn $\boldsymbol{P} = \boldsymbol{NP}$ gilt.

(c) Kleene-Stern; Wir nehmen an, $L \in \Sigma^*$ wird von einer dTM M mit polynomialer Zeitkomplexität $t(n)$ akzeptiert.

Behauptung: Die Maximalzahl der *nichtleeren* Teilwörter eines Wortes der Länge n ist gegeben durch $\sum_{i \le n} i = n(n+1)/2$.

Beweis: Der Summand i liefert die Anzahl der Teilwörter der Länge $n+1-i$. □

Die zu konstruierende Maschine M^* wird nachfolgend beschreiben. Sie verwendet eine als *dynamisches Programmieren* bekannte Technik:

Falls $x = \varepsilon$ akzeptiert M^* sofort.

Andernfalls testet M^* alle nichtleeren Teilwörter der Eingabe x auf Zugehörigkeit zu L , indem sie M anwendet. Wir denken us die Ergebnisse im oberen Dreieck einer binären $(|x|+1) \times (|x|+1)$ -Matrix Z angeordnet:

$$Z_{i,j} = \begin{cases} 1 & \text{falls das Teilwort von Position } i \text{ bis Position } j \text{ zu } L - \{\varepsilon\} \text{ gehört} \\ 0 & \text{sonst} \end{cases}$$

für $i < j < |x|+1$. Welche Bedeutung hat die reflexive transitive Hülle $Z^* = \sum_{i \in \mathbb{N}} Z^i$ von Z ? Man überlegt sich leicht, dass

$$(Z^k)_{i,j} = \begin{cases} 1 & \text{falls das Teilwort von Position } i \text{ bis Position } j \text{ zu } (L - \{\varepsilon\})^k \text{ gehört} \\ 0 & \text{sonst} \end{cases}$$

für $i < j < |x|+1$, und folglich

$$Z^+_{i,j} = \begin{cases} 1 & \text{falls das Teilwort von Position } i \text{ bis Position } j \text{ zu } (L - \{\varepsilon\})^+ \text{ gehört} \\ 0 & \text{sonst} \end{cases}$$

für $i \leq j < |x|+1$, bzw.,

$$Z^*_{i,j} = \begin{cases} 1 & \text{falls das Teilwort von Position } i \text{ bis Position } j \text{ zu } (L-\{\varepsilon\})^* = L^* \text{ gehört} \\ 0 & \text{sonst} \end{cases}$$

für $i \leq j < |x|$, Damit entscheidet die Komponente $Z^*_{0,|x|}$ die Frage, ob $w \in L^*$ gilt oder nicht.

Erfreulicherweise können wir Z^* effizient berechnen. Offenbar genügt es, bis $|x|+1$ zu summieren, denn x kann nicht aus mehr als $|x|$ nichtleeren Teilwörtern zusammengesetzt sein.. Statt Z betrachten wir die Matrix $Z + \Delta_{|x|+1}$, deren (i,j)-Komponente mit $i \leq j$ angibt, ob das Teilwort von Position i bis Position j zu $L \cup \{\varepsilon\}$ gehört. Entsprechend drückt $(Z+\Delta_{|x|+1})^k$ aus, ob die entsprechednen Teilwörter aus *höchstens* k Wörtern von $L = \{\varepsilon\}$ zusammengesetzt sind. Und um $(Z+\Delta_{|x|+1})^{|x|+1}$ zu bestimmen, genügt es, $Z+\Delta_{|x|+1}$ hinreichend oft zu *quadrieren*, nämlich $\lceil \log(|x|+1) \rceil$-mal. Weitere Multiplikation mit $Z+\Delta_{|x|+1}$ liefern dann keine Veränderungen mehr.

Jede Matrixmultiplikationen kostet $O((|x|+1)^3)$ Operationen, bis Z^* bekannt ist. Aber dies ist polynomial in $|x|$!

Dieser Matrix-Trick kann auch bei anderen Problemen von Nutzen sein.

(d) Shuffle: Die Abgeschlossenheit einer KLasse von Sprachen unter Shuffle folgt aus der Abgeschlossenheit unter Durchschnitten, ε-freien(!) homomorphen Bildern und homomorphen Urbildern. Zwar ist Letzteres sehr einfach zu zeigen (vergl. Aufgabe 2, Blatt 1), aber wie wir in (b) gesehen haben, ist $\boldsymbol{P}$ nur dann unter ε-freien homomorphen Bildern abgeschlossen, wenn $\boldsymbol{P} = \boldsymbol{NP}$ gilt.

Hausaufgabe 4 [15 PUNKTE]

Eine *Clique* in einem ungerichteten Graphen $\langle V, E \rangle$ ist eine Teilmenge $C \subseteq V$ mit der Eigenschaft, dass $\{u,v\} \in E$ gilt für alle $u, v \in C$ mit $u \neq v$. (Also induziert C einen *vollständigen Untergraphen*, d.h., je zwei verschiedene Knoten sind durch eine Kante verbunden.)

Zeigen Sie, dass sich die drei folgenden Entscheidungsprobleme wechselseitig aufeinander $\leq^{\text{poly}}_{P}$-reduzieren lassen:

- **Clique** (CLIQUE)
  Gegeben: ungerichteter Graph $G = \langle V, E \rangle$ mit $E \subseteq P_2(V)$ (Menge der 2-elementigen Teilmengen), $k \in \mathbb{N}$;
  zu entscheiden: ob in G eine Clique mit mindestens k Elementen existiert

- **Unabhängige Menge** (UM)
  Gegeben: ungerichteter Graph $G = \langle V, E \rangle$ mit $E \subseteq P_2(V)$, $k \in \mathbb{N}$;
  zu entscheiden: ob in G eine Knotenmenge mit mindestens k Elementen existiert, so dass keine zwei dieser Knoten miteinander verbunden sind.

- **Knotenüberdeckung** (KÜ)
  Gegeben: ungerichteter Graph $G = \langle V, E \rangle$ mit $E \subseteq P_2(V)$, $k \in \mathbb{N}$;
  zu entscheiden: ob in G eine Knotenmenge mit mindestens k Elementen existiert, so dass jede Kante einen Endpunkt in dieser Menge hat.

Abgabe bis Dienstag, 2019-06-25, bis 14:45 Uhr, im Kasten neben IZ 343