Języki Formalne i Złożoność Obliczeniowa
Rozwiązanie zadania 40L

Piotr Gdowski

4 kwietnia 2020

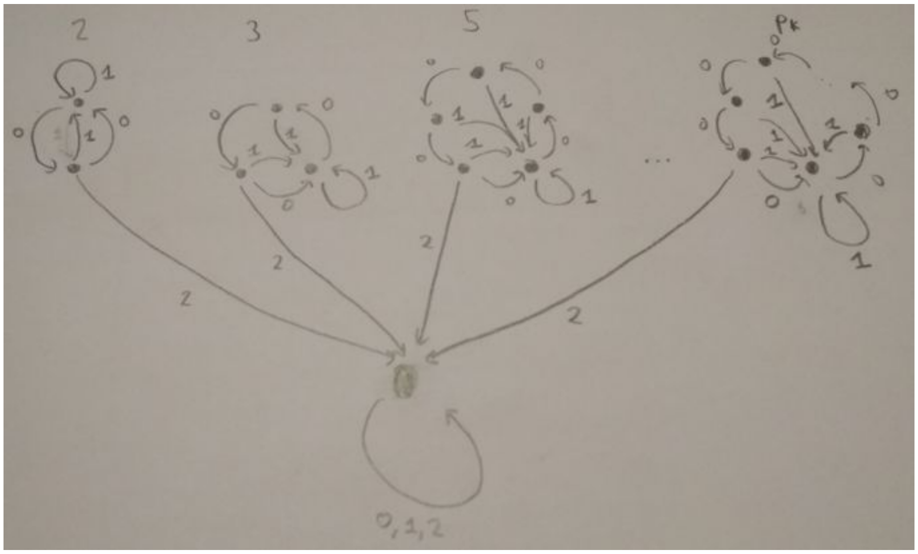
Zadanie 40L. Udowodnij, że dla każdego (dostatecznie dużego) n istnieje PDFA $\langle \Sigma, Q, q_0, F, \delta \rangle$, taki że $|Q| = n$ i że $csync(Q)$ jest niepusty, ale nie zawiera słowa krótszego od $p(n)$, gdzie p jest dowolnym, ustalonym wcześniej, wielomianem. Zakładamy, że $\Sigma = \{0, 1, 2\}$.

Definicja PDFA PDFA jedynie różni się od DFA tym, że funkcja przejścia δ może być funkcją częściową, to znaczy $\delta(q, a)$ może nie być określona dla niektórych par $\langle q, a \rangle$, gdzie $q \in Q$ i $a \in \Sigma$.

Definicja zbioru csync Dla danego PDFA $A = \langle \Sigma, Q, q_0, F, \delta \rangle$ i zbioru $S \subseteq Q$, przez $csync(S)$ oznaczmy zbiór takich słów $w \in \Sigma^*$, że dla każdego $q \in S$ wartość $\hat{\delta}(q, w)$ jest określona, oraz dla każdych dwóch stanów $q, q' \in S$ zachodzi $\hat{\delta}(q, w) = \hat{\delta}(q', w)$.

Rozwiązanie Niech p będzie wielomianem. Ustalmy $n \in \mathbb{N}$ (dostatecznie duże). Skonstruujemy PDFA $A = \langle \Sigma, Q, q_0, F, \delta \rangle$ takie, że $|Q| = n$ oraz dla każdego $w \in csync(Q)$ $|w| > p(n)$.

Idea konstrukcji PDFA Chcielibyśmy skorzystać z faktu, że suma n pierwszych liczb pierwszych jest zawsze mniejsza od $n^2 \, log \, n$, natomiast ich iloczyn zawsze jest większy od $2^{n \, log \, n}$. Stąd pomysł na zbudowanie k cykli, których długości będą kolejnymi liczbami pierwszymi, z jednym stanem, który miałby po jednej ścieżce wchodzącej od każdego cyklu. Wówczas synchronizacja wszystkich stanów przebiegałaby następująco: w pierwszych kroku (po przeczytaniu pierwszej litery) wszystkie stany z jednego cyklu zostałyby zmapowane na jeden, wybrany stan z tego cyklu. Następnie chcielibyśmy zsynchronizować cykle w takie stany, by ze nich wszystkich dało się przejść do dodatkowego stanu wspomnianego wyżej. Skoro długości cykli są różnym liczbami pierwszymi, to taka operacja będzie wymagać tylu kroków, ile wynosi iloczyn kolejnych liczb pierwszych. W ten sposób najkrótsze słowo synchronizujące będzie miało wykładniczą długość względem liczby stanów.



Rysunek 1: Skonstruowany PDFA spełniający warunki zadania

Konstrukcja PDFA Niech (p_i) będzie ciągiem kolejnych liczb pierwszych. Niech k będzie największą liczbą naturalną taką, że $\sum_{i=1}^{k} p_i \leq n$. Rozważmy PDFA A przedstawiony na rysunku 1. Automat składa się z k cykli, każdy o długości kolejnej liczby pierwszej. Po cyklu można chodzić w kółko poruszając się krawędziami z literą 0. Każdy cykl posiada stan, do którego można przejść z dowolnego innego stanu wewnątrz cyklu krawędzią z literą 1. Dodatkowo, na każdym cyklu znajduje się jeden wierzchołek (nazwijmy go "wyjściowym") pozwalający przejść z cyklu przy pomocy krawędzi z literą 2 do stanu "skupiającego"wszystkie cykle.

Lemat 1 Niech A PDFA jak zdefiniowano wyżej, $w = 10^{(\prod_{i=1}^{k} p_i)-1}2$. Wówczas w jest najkrótszym słowem synchronizującym wszystkie stany A.

Dowód lematu 1 Skonstruujemy najkrótsze słowo w synchronizujące wszystkie stany. Zauważmy, że nie może ono zaczynać się od 2, gdyż nie dla wszystkich stanów istnieje przejście z tą literą, natomiast jeśli $w = 0^*w'$, to również w' będzie słowem synchronizującym, zatem w nie będzie najkrótsze. Stąd wniosek, że w zaczyna się od litery 1. W ten sposób w każdym cyklu wszystkie stany zbiegły do jednego wierzchołka, znajdującego się jeden krok za stanem "wyjściowym".

Aby zsynchronizować stany tak, by znalazły się w wierzchołkach "wyjściowych" musimy wczytać słowo $0^{\prod_{i=1}^{k} p_i - 1}$. Wynika to z faktu, że znajdujemy się na drugim w kolejności wierzchołku w cyklu (licząc, że "wyjściowy" jest pierwszym") oraz że długości dwóch dowolnych (różnych) cykli są względnie pierwsze. Zauważmy, jeśli wczytamy literę 1 w trakcie wczytywania podanego ciągu zer, to przemieścimy się do stanu, w którym już wcześniej byliśmy (a zatem otrzymane słowo nie byłoby najkrótsze). Na koniec, kiedy wszystkie osiągnięte stany to "wyjściowe" i "skupiający", możemy użyć krawędzi z literą 2, przechodząc do stanu "skupiającego", w którym zsynchronizowaliśmy wszystkie stany. Zatem w jest najkrótszym słowem synchronizującym Q.

Lemat 2 $|w| > p(n)$ dla dostatecznie dużego n.

Dowód lematu 2 Bez strat ogólności załóżmy, że wielomian p jest niemalejący na przedziale $\langle n, \infty)$. Zauważmy, że wybraliśmy takie k, że $\sum_{i=1}^{k} p_i \leq n$ oraz że $\sum_{i=1}^{k+1} p_i > n$. Zatem skoro

$$\sum_{i=1}^{k} p_i \leq n < \sum_{i=1}^{k+1} p_i \leq (k+1)^2 log\,(k+1) \leq k^3 \tag{1}$$

to

$$p(n) \leq p(k^3) = p(q(k)) = p'(k) \tag{2}$$

gdzie $q(x) = x^3$, a skoro p i q są wielomianami, to p' też nim jest. Z drugiej strony

$$|w| = \prod_{i=1}^{k} p_i + 1 \geq 2^{k log\, k} \geq p'(k) \tag{3}$$

gdyż funkcja wykładnicza rośnie szybciej od dowolnego wielomianu. Czyli

$$|w| \geq 2^{k log\, k} \geq p'(k) > p(n) \tag{4}$$

Zatem skonstruowaliśmy PDFA, dla którego zbiór $csync(Q)$ jest niepusty i nie zawiera słowa krótszego od dowolnego wielomianu p, stąd taki automat istnieje, czego należało dowieść. ■