Zadanie 40 z "Około dwustu łatwych zadań z języków formalnych i złożoności obliczeniowej (i jedno czy dwa trudne) 2020 edition"

Łukasz Klasiński

4 kwietnia 2020

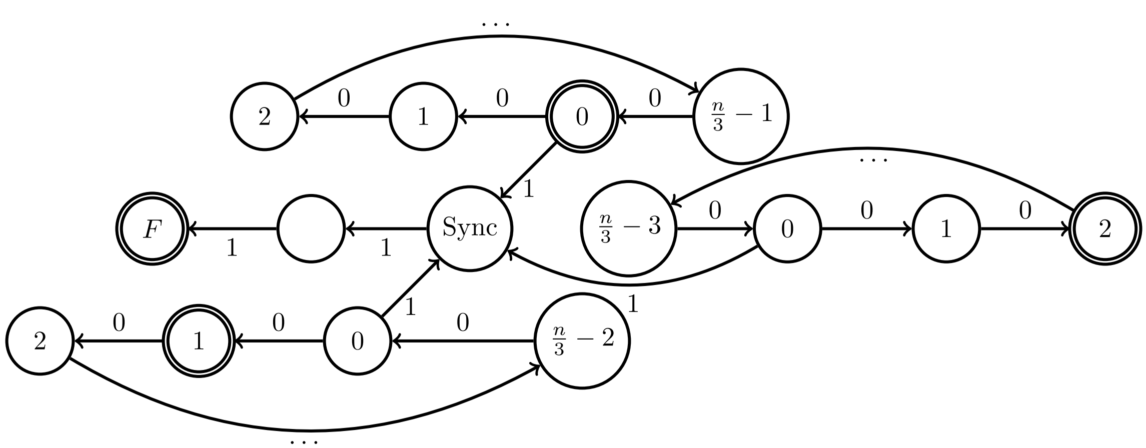
Zadanie 40M.

Udowodnij, że dla każdego (dostatecznie dużego) n istnieje PDFA $\mathcal{A} = \langle \Sigma, Q, q_0, F, \delta \rangle$, taki że $|Q| = n$ i że istnieje trzyelementowy $S \subseteq Q$ taki że zbiór $csync(S)$ jest niepusty ale nie zawiera słowa krótszego niż $n^3/10000$.

Rozwiązanie

Skonstruujemy taką maszynę, aby najkrótsze możliwe słowo synchronizujące musiało być odpowiednio długie.

Maszyna składa się z 3 cykli, wielkości kolejno $\frac{n}{3}$, $\frac{n}{3} - 1$, $\frac{n}{3} - 2$ oraz pojedyńczym wyjściem z cykli, które sprowadza się do wspólnego stanu. W cyklach przechodzimy do kolejnego stanu poprzez wczytanie litery 0, natomiast wychodzimy z cyklu po wczytaniu litery 1 (i znajdując się w odpowiednim stanie). Maszyna taka akceptuje tylko takie słowa, które składają się wpierw z k zer, gdzie k jest równe wielokrotności $\frac{n}{3}, \frac{n}{3} - 1$ lub $\frac{n}{3} - 2$, a następnie trzech jedynek. Poniżej wizualizacja - pogrubione wierzchołki w cyklach będą trzema wybranymi stanami $S \subseteq Q$.



Widać, że taki automat, dla odpowiednio dużego n, będzie miał dokładnie n stanów gdy n jest podzielne przez 3, a jeśli nie jest to możemy bardzo łatwo uzupełnić go o brakujące stany zwiększając bądź zmniejszając liczbę jedynek potrzebnych na dojście do stanu akceptującego F.

Popatrzymy teraz jak musi wyglądać najkrótsze słowo $s \in csync(S)$. Z konstrukcji widać, że aby wybrane stany mogły się zsynchronizować, muszą dojść do stanu *Sync*. Muszą to także zrobić jednocześnie, ponieważ jeśli zsynchronizujemy 2, to po wyjściu z cyklu 3 nie będzie miał już możliwości synchronizacji. Oznaczmy przez c_1, c_2, c_3 cykle od długości $\frac{n}{3}, \frac{n}{3} - 1, \frac{n}{3} - 2$. Wtedy ilość zer potrzebnych do zsynchronizowania się w punkcie 0 każdego cyklu wynosi:

$$NWW(c_1, c_2, c_3) = NWW(c_1, NWW(c_2, c_3)) = NWW(c_1, \frac{c_2 * c_3}{NWD(c_2, c_3)})$$

Ponieważ c_2 oraz c_3 są względnie pierwsze, to $NWD(c_2, c_3) = 1$. Otrzymujemy:

$$NWW(c_1, c_2 * c_3) = \frac{c_1 * c_2 * c_3}{NWD(c_1, c_2 * c_3)} = \frac{n^3 - 9n^2 + 18n}{27 * NWD(c_1, c_2 * c_3)}$$

Zauważmy, że dla odpowiednio dużego n licznik zbiega do n^3, ponadto NWD dla dużych liczb wynosi najwyżej 2, zatem otrzymamy $\frac{n^3}{54} > \frac{n^3}{10000}$, więc nasza maszyna dla odpowiedniego n spełnia założenia.