Michał Górniak

Zadanie 40L. 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 $csync(Q)$ jest niepusty ale nie zawiera słowa krótszego niż $p(n)$, gdzie p jest dowolnym, ustalonym wcześniej, wielomianem. Zakładamy, że $\Sigma = \{0, 1, 2\}$.

Wskazówka. Dla każdego naturalnego $k > 1$ istnieje liczba pierwsza p, taka że $k < p < 2k$.[1]

Rozwiązanie. Weźmy dowolny wielomian p. Niech $d = \deg(p) + 1$. Zauważmy, że dla dowolnej stałej $c > 0$, istnieje dostatecznie duże $n_c \in N$, takie że $(\forall n > n_c)\ \frac{n^d}{c} - 1 > p(n)$.

Niech $c = 2^{d(d+1)}$ oraz n będzie dowolną liczbą naturalną większą od $\max(n_c, 2^{d+2})$. Ze wskazówki wnioskujemy, że istnieją liczby pierwsze $p_1, p_2, \ldots, p_d$ spełniające

$$\lfloor \frac{n}{2^1} \rfloor > p_1 > \lfloor \frac{n}{2^2} \rfloor > p_2 > \lfloor \frac{n}{2^3} \rfloor > \ldots > \lfloor \frac{n}{2^d} \rfloor > p_d > \lfloor \frac{n}{2^{d+1}} \rfloor > 1.$$

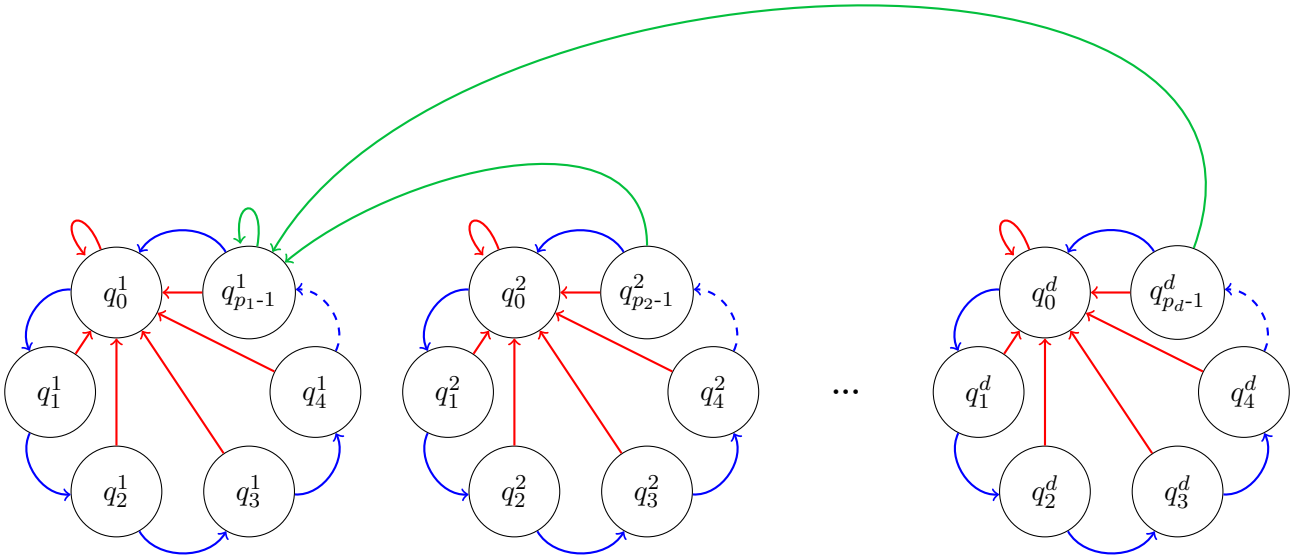
Zauważmy, że

$$\sum_{i=1}^{d} p_i < \sum_{i=1}^{d} \lfloor \frac{n}{2^i} \rfloor \leqslant \sum_{i=1}^{d} \frac{n}{2^i} < n \sum_{i=1}^{\infty} \frac{1}{2^i} = n.$$

Niech $k = p_1 p_2 \ldots p_d - 1$. Zauważmy, że ponieważ $p_i > \lfloor \frac{n}{2^{d+1}} \rfloor$, to $p_i \geqslant \frac{n}{2^{d+1}}$. Możemy stąd wywnioskować, że

$$k = \prod_{i=1}^{d} p_i - 1 \geqslant (\frac{n}{2^{d+1}})^d - 1 = \frac{n^d}{2^{d(d+1)}} - 1 = \frac{n^d}{c} - 1 > p(n).$$

Wystarczy zatem, że pokażemy dla naszego wybranego n taki automat PDFA $\mathcal{A}$, że $|Q| \leqslant n$ oraz $csync(Q)$ jest niepusty, ale nie zawiera słowa krótszego niż k.



Ilustracja szukanego automatu. Czerwone krawędzie odpowiadają literze 0*, niebieskie* 1*, a zielone* 2*.*

Automat $\mathcal{A}$ wygląda tak, jak na powyższym rysunku.[2] Składa się z d cykli, i-ty z nich ma długość p_i, czyli $|Q| = \sum_{i=1}^{d} p_i \leqslant n$. Zgodnie z intuicją przekazaną na rysunku, poniżej opisana jest formalna definicja funkcji przejścia:

$$\begin{cases} \delta(q_j^i, 0) = q_0^i & \text{dla } 1 \leqslant i \leqslant d,\ 0 \leqslant j < p_i, \\ \delta(q_j^i, 1) = q_{(j+1) \bmod p_i}^i & \text{dla } 1 \leqslant i \leqslant d,\ 0 \leqslant j < p_i, \\ \delta(q_{p_i-1}^i, 2) = q_{p_1-1}^1 & \text{dla } 1 \leqslant i \leqslant d. \end{cases}$$

[1] `https://en.wikipedia.org/wiki/Bertrand%27s_postulate`

[2] Próbowałem uniknąć podwójnego indeksowania, ale nie wyszło.

Pozostało jeszcze uzasadnić, że (1) $csync(Q)$ jest niepusty oraz że (2) najkrótsze słowo do niego należące ma długość przynajmniej k. Zacznijmy od (1). Weźmy w tym celu słowo $w = 01^k2$. Weźmy dowolny stan automatu $\mathcal{A}$. Zobaczmy gdzie znajdziemy się po przeczytaniu słowa w. Po przeczytaniu 0 znajdziemy się w wierzchołku q_0^i, gdzie i jest numerem cyklu wierzchołka od którego zaczęliśmy. Następnie po kolejnych k jedynkach znajdziemy się w wierzchołku $q_{p_i-1}^i$, bo

$$k = p_1p_2 \dots p_d - 1 = (p_1p_2 \dots p_{i-1}p_{i+1} \dots p_d)p_i - 1 \equiv p_i - 1 \pmod{p_i}.$$

Stąd oczywiście wynika, że po przeczytaniu dwójki trafimy do $q_{p_1-1}^1$, co pokazuje, że niezależnie od wyboru stanu początkowego, po przeczytaniu słowa w jesteśmy w tym samym stanie, czyli $csync(Q)$ jest niepusty.

Pokażmy zatem (2). Niech w' będzie najkrótszym słowem należącym do $csync(Q)$. Żuczkowy dowód: w każdym stanie automatu niech stanie żuczek. Po otrzymaniu kolejnej literki z w' każdy z żuczków się przesuwa zgodnie z poleceniem, a jeśli któryś nie może tego zrobić, to wszystkie żuczki płaczą do końca świata. Po przeczytaniu całego słowa żuczki się cieszą tylko wtedy, gdy wszystkie są razem w tym samym stanie. Najpierw przeanalizujmy literę 2. Zauważmy, że po przeczytaniu litery 2, niezależnie od stanu, albo nie możemy już nigdzie pójść, albo trafiamy do $q_{p_1-1}^1$. Z drugiej strony $q_{p_1-1}^1$ jest jedynym wierzchołkiem osiągalnym ze wszystkich innych, zatem musimy użyć co najmniej jednej dwójki. To oznacza, że najkrótsze słowo w $csync(Q)$ ma dokładnie jedną dwójkę i znajduje się ona na końcu słowa, co z kolei implikuje, że przed jej użyciem każdy z żuczków znajduje się cały czas na swoim cyklu. Rozpatrzmy teraz literę 0. Zauważmy, że w słowie w' musimy mieć chociaż jedno 0, bo inaczej w każdym kroku za wyjątkiem ostatniego, na każdym cyklu każdy żuczek byłby cały czas samotny w swoim obecnym stanie, każdy z nich zawsze przesuwałby się do kolejnego wierzchołka swojego cyklu. Musimy zatem używać 0, ale każde jego użycie sprawia, że dzieje się dokładnie to samo: wszystkie żuczki z danego cyklu trafiają do stanu q_0^i, gdzie i jest numerem ich cyklu. To oznacza, że skoro w' jest najkrótsze, to użyjemy zera na samym początku i dokładnie raz. Zatem $w' = 01^x2$, gdzie x jest pewną liczbą naturalną. Ta liczba musi mieć taką własność, że jeśli na każdym cyklu wykonamy dokładnie x kroków zaczynając w q_0^i, to skończymy w $q_{p_i-1}^i$. Zatem musi spełniać układ równań:

$$\begin{cases} x \equiv p_1 - 1 \pmod{p_1}, \\ x \equiv p_2 - 1 \pmod{p_2}, \\ \dots \\ x \equiv p_d - 1 \pmod{p_d}. \end{cases}$$

Ale jako, że $x \in \mathbb{N}$ oraz $\{p_i \mid 1 \leqslant i \leqslant d\}$ jest zbiorem parami różnych liczb pierwszych, to (np. z Chińskiego Twierdzenia o Resztach) wiemy że najmniejszą liczbą naturalną spełniającą ten układ kongruencji jest $k = p_1p_2 \dots p_d - 1$, co implikuje, że w' ma długość przynajmniej k, co kończy dowód.