Algorytmy online

Lista 4

Theorem 1. *Współczynnik konkurencyjności algorytmu Rand przeciwko adwersarzowi nieświadomemu wynosi co najmniej k.*

Dowód. Oznaczmy strony, które są na początku w pamięci podręcznej przez $a_1, a_2, \ldots, a_k$. Rozważmy następującą sekwencję:

$$\sigma = \underbrace{(b_1, a_2, a_3, \ldots, a_k)^\ell}_{\text{blok 1}}, \underbrace{(b_2, a_2, a_3, \ldots, a_k)^\ell}_{\text{blok 2}}, \ldots, \underbrace{(b_\gamma, a_2, a_3, \ldots, a_k)^\ell}_{\text{blok } \gamma} \ ,$$

gdzie ℓ zostanie ustalone za chwilę, a wszystkie elementy b_i są różne od elementów a_j (dla dowolnych i, j mamy $b_i \neq a_j$). Koszt optymalnego algorytmu w każdym bloku to co najwyżej 1. Pokażemy, że dla dowolnego bloku zachodzi

$$\mathbf{E}[Rand(\mathsf{blok})] \geqslant k \cdot \left(1 - \left(\frac{k-1}{k}\right)^\ell\right) \tag{1}$$

Czynnik $(\frac{k-1}{k})^\ell$ zbiega do zera jeśli weźmiemy odpowiednio duże ℓ; wtedy $\mathbf{E}[Rand(\mathsf{blok})]$ jest dowolnie bliskie k.

Rozważmy zatem dowolny i-ty blok, złożony z ℓ segmentów $(b_i, a_2, a_3, \ldots, a_k)$. Na początku tego bloku *Rand* nie ma w swojej pamięci elementu b_i, więc pierwsze odwołanie w bloku to chybienie. Mówimy, że chybienie algorytmu *kończy się sukcesem*, jeśli *Rand* wybiera do wyrzucenia taką stronę, że po danym kroku w jego pamięci podręcznej są wszystkie ze stron $\{b_i, a_2, a_3, \ldots, a_k\}$ (od takiej chwili *Rand* nie płaci już nic w danym bloku). Zauważmy, że chybienie kończy się sukcesem z prawdopodobieństwem nie większym niż $1/k$ (jeśli w momencie chybienia $k-1$ stron jest „dobrych", to jest to dokładnie prawdopodobieństwo, że żadnej z nich nie wyrzucimy; w przeciwnym przypadku nie mamy szans na sukces).

Mamy zatem do czynienia z następującym procesem: jeśli *Rand* chybia to płaci 1 i z prawdopodobieństwem co najwyżej $1/k$ osiąga sukces (i nie płaci już więcej). Z pozostałym prawdopodobieństwem za jakiś czas będzie musiał znowu chybić (najpóźniej wydarzy się to w kolejnym segmencie). Jeśli zatem mielibyśmy nieskończenie wiele segmentów ($\ell = \infty$), to liczba chybień byłaby liczbą kroków do wystąpienia pierwszego sukcesu w procesie niezależnych prób Bernouliego z prawdopodobieństwem sukcesu $1/k$. Czyli w wartości oczekiwanej *Rand* chybiałby k razy. Jednak liczba segmentów w bloku jest jednak skończona i wynosi ℓ. Jeśli liczba chybień w bloku wyniesie powyżej ℓ to „wybaczamy" algorytmowi koszt za odwołania powyżej ℓ-tego chybienia. Mamy jednak gwarancję, że jeśli algorytmowi przyjdzie chybiać ℓ razy, to te razy zmieszczą się w bloku. Rozważmy zatem proces Bernouliego „obcięty" po losowaniu n; niech W_n będzie zmienną losową oznaczającą liczbę kroków w takim procesie. Z powyższych rozważań wynika, że $\mathbf{E}[Rand(\mathsf{blok})] \geqslant \mathbf{E}[W_\ell]$. Możemy zatem napisać

$$\begin{aligned}\mathbf{E}[W_\ell] &= \ell \cdot \Pr[W_\ell \geqslant \ell] + \sum_{i=1}^{\ell-1} i \cdot \Pr[W_\ell = i] \\ &= \ell \cdot (1-p)^{\ell-1} + \sum_{i=1}^{\ell-1} i \cdot p \cdot (1-p)^{i-1} \ ,\end{aligned}$$

gdzie $p = 1/k$ i po długich przekształceniach dojść do (**??**) lub też przyjrzeć się czym jest $\mathbf{E}[W_n]$. Dla $n = 1$, $W_1 = 1$. Natomiast dla $n \geqslant 2$ mogą zajść dwa przypadki Albo — z prawdopodobieństwem p — dostaniemy sukces i wtedy $W_n = 1$, albo też sukcesu nie będzie i wtedy czas oczekiwania jest równy $1 + \mathbf{E}[W_{n-1}]$. Zatem

$$\mathbf{E}[W_n] = p + (1-p) \cdot (1 + \mathbf{E}[W_{n-1}]) = 1 + (1-p) \cdot \mathbf{E}[W_{n-1}] \ .$$

Rozwijając tę zależność $\ell - 1$ razy i podstawiając $\mathbf{E}[W_1] = 1$ otrzymujemy

$$\mathbf{E}[W_\ell] = 1 + (1-p) + (1-p)^2 + \ldots + (1-p)^{\ell-1} = \frac{1-(1-p)^\ell}{1-(1-p)} = k \cdot \left(1 - (1-1/k)^\ell\right) \ . \qquad \square$$