Rolf Wanka
Alexander Raß

Erlangen, 11. Juli 2019

Übungen zur Vorlesung
Organic Computing
SS 2019
Blatt 8

AUFGABE 20:
Sei π eine Permutation der Zahlen 1 bis n. Das in der Vorlesung vorgestellte Sortiertheitsmaß *Longest Ascending Subsequence* LAS ist folgendermaßen definiert: $\mathrm{LAS}(\pi)$ ist das größtmögliche k, so daß $\pi(i_1) < \cdots < \pi(i_k)$ für $i_1 < \cdots < i_k$ gilt. Beachten Sie, daß die Positionen nicht unbedingt unmittelbar aufeinanderfolgen müssen. Z. B. ist

$$\mathrm{LAS}([1,9,5,13,3,11,7,15,2,10,6,14,4,12,8,16]) = 6$$

wegen der Teilfolgen $[1,3,7,10,14,16]$ und $[1,3,7,10,12,16]$.

In der Vorlesung wurde ein $(1+1)$-EA vorgestellt, der eine Eingabepermutation sortiert. Im folgenden sehen Sie den EA, wenn als Sortiertheitsmaß LAS benutzt wird.

ALGORITHMUS $(1+1)$-EA zum Sortieren

Input: Permutation π der Schlüssel $1,\dots,n$
Zielfunktion: LAS

$t := 0$; $\pi_0 = \pi$;
while $\mathrm{LAS}(\pi_t) < n$ **do** // *Berechne die nächste Generation* $t+1$

// *Bestimme* π' *aus* π_t *durch Mutation*
$\pi' := \pi_t$;
würfle $S \in \mathbb{N}_0$ gemäß Poisson-Verteilung mit Parameter $\lambda = 1$, d. h. $\Pr[S = i] = \dfrac{1}{\mathrm{e} \cdot i!}$;
for $i := 1$ **to** $S+1$ **do**
würfle gleichverteilt ein Index-Paar i, j mit $i \neq j$ und $i, j \in \{1,\dots,n\}$;
$\begin{cases} \text{mit Wahrscheinlichkeit } \frac{1}{2}\text{: wende exch}(i,j) \text{ auf } \pi' \text{ an} \\ \text{mit Wahrscheinlichkeit } \frac{1}{2}\text{: wende jump}(i,j) \text{ auf } \pi' \text{ an} \end{cases}$
end for;

// *Umweltselektion: Survival of the fittest*
if $\mathrm{LAS}(\pi') > \mathrm{LAS}(\pi_t)$
then $\pi_{t+1} := \pi'$
else $\pi_{t+1} := \pi_t$
end if;
$t := t+1$
end while

(Auf der Rückseite geht es weiter ➛)

Zeigen Sie: Die erwartete Anzahl an Generationen des $(1+1)$-EA ist $O(n^2 \log n)$.

Hinweis: Sei k, $k < n$, die Länge der längsten steigenden Teilfolge, d. h. die Folge ist noch nicht sortiert. Dann gibt es $n-k$ Schlüssel, die außerhalb dieser Teilfolge stehen. Ein solcher Schlüssel kann durch mindestens eine „gute“ Jump-Operation an eine richtige Stelle in der Teilfolge springen und den Wert der Zielfunktion verbessern. Berechnen Sie die Wahrscheinlichkeit, daß dies passiert, lassen Sie exch-Operationen ganz außen vor, und gehen Sie dann so vor wie in der Vorlesung für INV.

AUFGABE 21:
Sie haben n Kisten (engl.: *bins*) $K_1, \ldots, K_n$ und n Bälle $B_1, \ldots, B_n$, und Sie werfen diese Bälle so auf die Kisten, daß Sie jede der Kisten mit gleicher Wahrscheinlichkeit treffen. D. h. für alle $i, j \in \{1, \ldots, n\}$:

$$\Pr[B_i \text{ landet in } K_j] = \frac{1}{n}$$

(a) Begründen Sie, warum nach dem Werfen aller Bälle für die erste Kiste K_1 gilt:

$$\Pr[K_1 \text{ enthält mindestens } k \text{ Bälle}] \leq \binom{n}{k} \cdot \left(\frac{1}{n}\right)^k$$

Zur Begründung gehören auch Argumente für das Wort „mindestens“ und das „$\leq$“-Zeichen.

(b) Zeigen Sie nun, daß für hinreichend große n gilt:

$$\Pr\left[\text{Die vollste Kiste enthält mindestens } \frac{3 \ln n}{\ln \ln n} \text{ Bälle}\right] \leq \frac{1}{n}$$

Hinweis: Achten Sie darauf, daß jetzt über alle Kisten gesprochen wird.

Nutzen Sie zuerst, daß mit der Euler-Zahl e gilt:

$$\binom{n}{k} \cdot \left(\frac{1}{n}\right)^k \leq \left(\frac{\mathrm{e}}{k}\right)^k$$

Verständnisfrage: Wieso benötigt man gerade den Faktor 3 (und nicht z. B. 2)?

Beweisservice für obige Abschätzung: Da $\mathrm{e}^k = \sum_{i=0}^{\infty} \frac{k^i}{i!}$ ist, gilt

$$\binom{n}{k} = \frac{n!}{k! \cdot (n-k)!} \leq \frac{n^k}{k!} = \frac{n^k}{k^k} \cdot \frac{k^k}{k!} < \left(\frac{n}{k}\right)^k \cdot \sum_{i=0}^{\infty} \frac{k^i}{i!} = \left(\frac{n \cdot \mathrm{e}}{k}\right)^k \quad .$$