Department Informatik

Approximations- und Online-Algorithmen

Dr. Hans-Joachim Böckenhauer
Dr. Dennis Komm

https://courses.ite.inf.ethz.ch/appron22

Lösungsvorschläge – Blatt 11

Zürich, 19. Mai 2022

Lösung zu Aufgabe 16

(a) Der Gegenspieler bietet nacheinander m Aufträge mit Kosten von jeweils 1 an. Sollte der gegebene Online-Algorithmus Alg zwei davon derselben Maschine zuordnen, wird die Eingabe sofort beendet und die berechnete Lösung ist somit doppelt so teuer wie die optimale. Sollte Alg nach dem letzten Zeitschritt alle Aufträge verschiedenen Maschinen zugewiesen haben, wird ein Auftrag mit Kosten 2 gegeben, der zu einem Makespan von 3 führt, wohingegen die optimale Lösung Kosten von 2 hat.

(b) Wir führen einen Widerspruchsbeweis, wie im Aufgabentext vorgeschlagen. Sei $\varepsilon > 0$. Angenommen, es existiert ein $(3/2 - \varepsilon)$-kompetitiver Online-Algorithmus mit einer additiven Konstanten $\alpha > 0$. (Für $\alpha = 0$ haben wir die Aussage bereits mit Teilaufgabe (a) bewiesen.) Offensichtlich ist diese Aussage nur sinnvoll für $\varepsilon \leq 1/2$. Wir nutzen dieselbe Instanz wie in Aufgabenteil (a), bloss mit einer Skalierung aller Kosten (Auftragslängen) mit dem Faktor α/ε. Der Gegenspieler bietet also zunächst m Aufträge an, die diesmal jeweils Kosten von α/ε besitzen. Sollte Alg wieder zwei davon derselben Maschine zuteilen, endet die Eingabe. Da Alg $(3/2-\varepsilon)$-kompetitiv ist, muss gelten

$$\mathrm{cost}(\textsc{Alg}(I)) \leq \left(\frac{3}{2} - \varepsilon\right)\mathrm{cost}(\textsc{Opt}(I)) + \alpha$$

und somit

$$\begin{aligned} & 2\frac{\alpha}{\varepsilon} \leq \left(\frac{3}{2} - \varepsilon\right)\frac{\alpha}{\varepsilon} + \alpha \\ \iff \quad & 2\frac{\alpha}{\varepsilon} \leq \frac{3}{2}\frac{\alpha}{\varepsilon} - \alpha + \alpha \\ \iff \quad & \frac{\alpha}{2\varepsilon} \leq 0 \end{aligned}$$

was ein Widerspruch ist.

Sollte Alg wiederum alle Aufträge verschiedenen Maschinen zuweisen, folgt ein Auftrag mit den doppelten Kosten $2\alpha/\varepsilon$ im letzten Zeitschritt. Nun folgt, dass

$$3\frac{\alpha}{\varepsilon} \leq \left(\frac{3}{2} - \varepsilon\right)2\frac{\alpha}{\varepsilon} + \alpha$$

$$\begin{aligned} \iff \quad 0 &\leq -\varepsilon 2\frac{\alpha}{\varepsilon} + \alpha \\ \iff \quad 2\alpha &\leq \alpha \ , \end{aligned}$$

gelten muss, was erneut ein Widerspruch ist.

Lösung zu Aufgabe 17

(a) Der Gegenspieler bietet insgesamt $n = m(m-1)+1$ Aufträge an, von denen der letzte Kosten von m besitzt und alle vorherigen Kosten von jeweils 1. Im letzten Zeitschritt, in dem also der Auftrag mit Kosten m angeboten wird, hat der Greedy-Algorithmus alle bisherigen Aufträge so verteilt, dass alle Maschinen eine Arbeitszeit von $m-1$ besitzen. Der nun zuletzt gegebene Auftrag muss einer dieser Maschinen zugewiesen werden, was zu einem Makespan von $2m-1$ führt. Eine optimale Lösung weist die ersten $m(m-1)$ Aufträge hingegen gleichmässig den ersten $m-1$ Maschinen zu, was zu Arbeitszeiten von jeweils m führt. Somit kann der letzte Auftrag auf der noch freien m-ten Maschine laufen und es ergeben sich Arbeitszeiten von jeweils m und eine untere Schranke von

$$\frac{2m-1}{m} = 2 - \frac{1}{m}$$

auf den strikten kompetitiven Faktor des Greedy-Algorithmus.

Bemerkung: Die untere Schranke lässt sich mit derselben Instanz mittels einer Skalierung aller Kosten um einen Faktor α/ε auf allgemeine Kompetitivität verallgemeinern.

(b) Sei I eine beliebige Eingabe für das Online-Makespan-Scheduling-Problem mit m Maschinen. Sei M_j mit $1 \leq j \leq m$ die Maschine mit der grössten Arbeitszeit in der berechneten Greedy-Lösung. Seien k die Kosten des letzten Auftrags, der M_j zugewiesen wurde, und sei k' die Arbeitszeit von M_j unmittelbar vor dieser Zuweisung. Die Kosten der Greedy-Lösung sind somit $\text{cost}(\textsc{Greedy}(I)) = k' + k$.

Die Arbeitszeit aller Maschinen insgesamt muss mindestens $m \cdot k' + k$ sein, da M_j eine kleinste Arbeitszeit von k' hatte, bevor der Auftrag mit Kosten k zugewiesen wurde. Die Eingabe enthält somit Aufträge mit Kosten von mindestens $m \cdot k' + k$ und die optimalen Kosten $\text{cost}(\textsc{Opt}(I))$ sind mindestens $(m \cdot k' + k)/m$. Wir erhalten schliesslich

$$\begin{aligned} \text{cost}(\textsc{Greedy}(I)) &= k' + k \\ &\leq \text{cost}(\textsc{Opt}(I)) - \frac{k}{m} + k \\ &\leq \text{cost}(\textsc{Opt}(I)) + \left(1 - \frac{1}{m}\right)k \\ &\leq \left(2 - \frac{1}{m}\right)\text{cost}(\textsc{Opt}(I)) \ , \end{aligned}$$

wobei wir für die letzte Abschätzung die offensichtliche Tatsache verwendet haben, dass die optimale Lösung mindestens so gross sein muss wie jeder beliebige Auftrag in I.