Übung 9 – Algorithmen II

Tobias Heuer, Sebastian Lamm – *tobias.heuer@kit.edu, lamm@kit.edu*
`http://algo2.iti.kit.edu/AlgorithmenII_WS19.php`

Institut für Theoretische Informatik - Algorithmik II

Themenübersicht

- Online Algorithmen
  (Probleme mit beschränkter Information gut abschätzen)

Online Algorithmen

Grundlagen

- Information kommt nach und nach an
- Entscheidungen / Berechnungen mit beschränkter Information

  $\rightarrow$ normalerweise supoptimale Lösungen

Beispiele

- *Ski Rental* Problem
- Job-Scheduling / Memory-Paging
- *Streaming* Algorithmen
- Suche in Labyrinthen / Routing in Kommunikationsnetzen
- ...

Online Algorithmen
Gütemaß

- Ein Online Algorithmus *ALG* hat einen kompetitiven Faktor

$$c = \sup_I \frac{ALG(I)}{OPT(I)}$$

mit *OPT* optimaler Offline Algorithmus, *I* Probleminstanz
(hier für Minimierungsprobleme; "Approximationsfaktor" c)

Einige Eigenschaften

- $c = \infty$ möglich $\longrightarrow$ Online Algorithmus beliebig schlecht
- $c = const.$ (normal nicht von Problemgröße abhängig)
  aber ev. abhängig von weiteren Problemparametern

Online Algorithmen

Ski Rental Problem

Generische Problembeschreibung

- kostenpflichtige Nutzung einer Leistung (alle Kosten > 0)
  - einmalige Kosten K für unbeschränkte Nutzung oder
  - Kosten $k < K$ für jede Nutzung
- Anzahl Nutzungen n unbekannt
  $\Rightarrow$ (Wann) Soll man K für unbeschränkte Nutzung zahlen?

Nomenklatur

- Instanz I durch Anzahl Nutzungen n bestimmt
- $OPT(I)$, $ALG(I) \mathrel{\hat{=}}$ Gesamtkosten

Online Algorithmen

Ski Rental

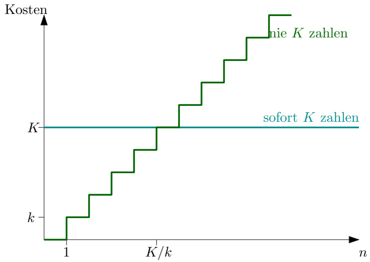
- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$

$\Rightarrow \quad c =$

Online Algorithmen

Ski Rental

- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$



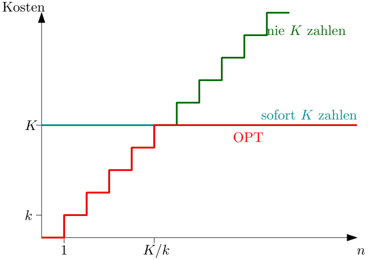
$\Rightarrow \quad c =$

Online Algorithmen

Ski Rental

- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$

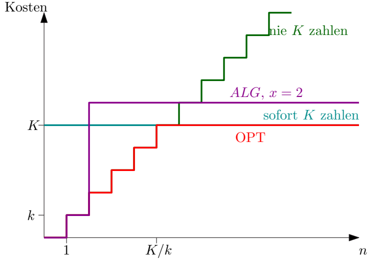
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$



$\Rightarrow \quad c =$

Online Algorithmen
Ski Rental

- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$

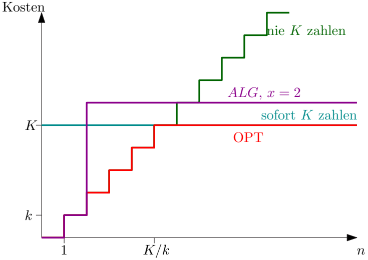


$\Rightarrow$ $c =$

Online Algorithmen

Ski Rental

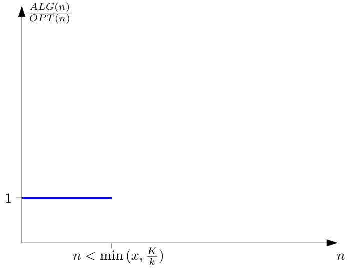
- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$



$\Rightarrow \quad c = ?$

Online Algorithmen
Ski Rental

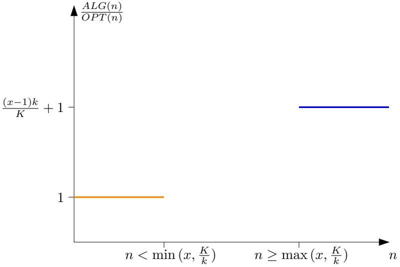
- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$



$\Rightarrow \quad c = ?$

Online Algorithmen
Ski Rental

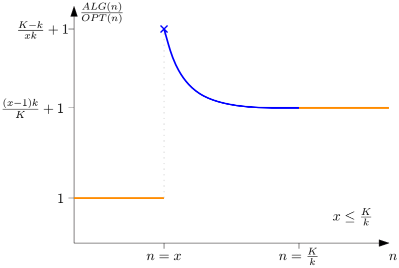
- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$



$\Rightarrow$ $c = ?$

Online Algorithmen
Ski Rental

- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$

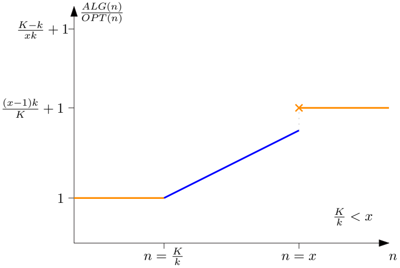


$\Rightarrow \quad c = ?$

Online Algorithmen

Ski Rental

- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$



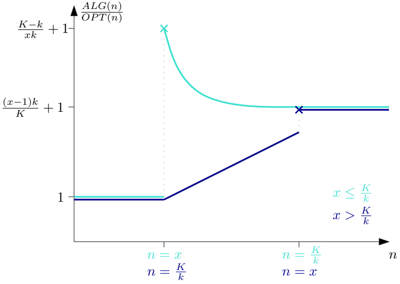
$\Rightarrow$ $c = ?$

Online Algorithmen

Ski Rental

- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$

$\Rightarrow \quad c = \begin{cases} \frac{K-k}{xk} + 1 & x \leq \frac{K}{k} \\ \frac{(x-1)k}{K} + 1 & \text{sonst} \end{cases}$

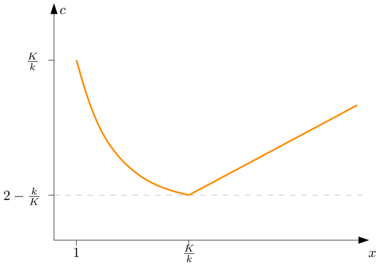


Online Algorithmen

Ski Rental

- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$

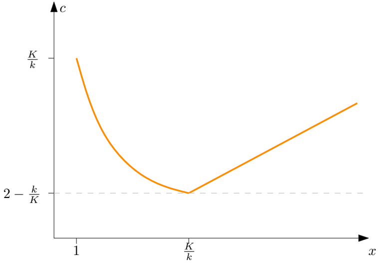
$$\Rightarrow \quad c = \begin{cases} \frac{K-k}{xk} + 1 & x \leq \frac{K}{k} \\ \frac{(x-1)k}{K} + 1 & \text{sonst} \end{cases}$$



Online Algorithmen

Ski Rental

- allgemein gilt $c = sup_n \frac{ALG(n)}{OPT(n)}$
  - $ALG(n) = \begin{cases} n \cdot k & n < x \\ (x-1) \cdot k + K & \text{sonst} \end{cases}$
    (zahle K vor x-ter Nutzung)
  - $OPT(n) = \begin{cases} n \cdot k & n < \frac{K}{k} \\ K & \text{sonst} \end{cases}$



$$\Rightarrow \quad c = \begin{cases} \frac{K-k}{xk} + 1 & x \leq \frac{K}{k} \\ \frac{(x-1)k}{K} + 1 & \text{sonst} \end{cases}$$

Sonderfall: Bezahle K, nutze nie!

$\Rightarrow \; ALG(n) = K$

$\Rightarrow \; c = \infty$

(entspricht $x = 0$)

Online Algorithmen
Ski Rental

Es gilt $c = \begin{cases} \frac{K-k}{xk} + 1 & x \leq \frac{K}{k} \\ \frac{(x-1)k}{K} + 1 & \text{sonst} \end{cases}$

Frage: Für welches x ist c minimal?

$\Rightarrow$ für $x = \frac{K}{k}$ ist c minimal mit $c = 2 - \frac{k}{K}$

Zahlenbeispiel: $K = 100$, $k = 10 \Rightarrow c = 1.9$

Bester randomisierter Algorithmus: $c = e/(e-1) \approx 1.58$

Online Algorithmen

Ski Rental

Es gilt $c = \begin{cases} \frac{K-k}{xk} + 1 & x \leq \frac{K}{k} \\ \frac{(x-1)k}{K} + 1 & \text{sonst} \end{cases}$

Frage: Für welches x ist c minimal?

$\Rightarrow$ für $x = \frac{K}{k}$ ist c minimal mit $c = 2 - \frac{k}{K}$

Zahlenbeispiel: $K = 100$, $k = 10 \Rightarrow c = 1.9$

Bester randomisierter Algorithmus: $c = e/(e-1) \approx 1.58$

Online Algorithmen

Ski Rental

Es gilt $c = \begin{cases} \frac{K-k}{xk} + 1 & x \leq \frac{K}{k} \\ \frac{(x-1)k}{K} + 1 & \text{sonst} \end{cases}$

Frage: Für welches x ist c minimal?

$\Rightarrow$ für $x = \frac{K}{k}$ ist c minimal mit $c = 2 - \frac{k}{K}$

Zahlenbeispiel: $K = 100, k = 10 \Rightarrow c = 1.9$

Bester randomisierter Algorithmus: $c = e/(e-1) \approx 1.58$

Online Algorithmen

Ski Rental

Es gilt $c = \begin{cases} \frac{K-k}{xk} + 1 & x \leq \frac{K}{k} \\ \frac{(x-1)k}{K} + 1 & \text{sonst} \end{cases}$

Frage: Für welches x ist c minimal?

$\Rightarrow$ für $x = \frac{K}{k}$ ist c minimal mit $c = 2 - \frac{k}{K}$

Zahlenbeispiel: $K = 100, k = 10 \Rightarrow c = 1.9$

Bester randomisierter Algorithmus: $c = e/(e-1) \approx 1.58$

Online Algorithmen

Doubling Strategie

Doubling

- Strategie für Online/Offline Approximationsalgorithmen

Idee

- schätze Lösung konservativ ab (eher zu kleine Schätzung)
- prüfe, ob Problem gelöst
- falls nicht erfolgreich, vergrößere Schätzung geometrisch
  (z.B. verdoppeln, verdreifachen, ...)

Online Algorithmen

Online Bidding

Problembeschreibung

- Unbekannter Zielwert $U > 1$
- Bieter gibt eine Reihe an Geboten (b_i) ab bis $b_k \geq U$
- Bieter muss Summe der Gebote $\sum_{i=0}^{k} b_i$ bezahlen
  $\Rightarrow$ Wie kann der Bieter möglichst geschickt vorgehen?

Nomenklatur

- Lösung durch Reihe (b_i) beschrieben
- $OPT(I)$, $ALG(I) \triangleq$ summierten Kosten

Online Algorithmen

Online Bidding

Optimaler Offline Algorithmus

- kennt den Zielwert U
- folglich bietet er direkt $b_0 = U$
  $\Rightarrow OPT = U$

Kompetitiver Faktor

$$c = \sup_{k,U} \left\{ \frac{b_0 + b_1 + \cdots + b_k}{U} : b_{k-1} < U \leq b_k \right\}$$

- Strategien sind z.B. (a) $b_i = i \cdot x$, (b) $b_i = a \cdot x^i$

Online Algorithmen

Online Bidding

Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

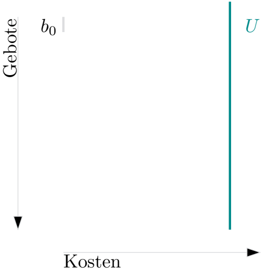
- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

  $\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$

  $\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$

  $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}(\frac{U}{x} + 2)(\frac{U}{x} + 1) \cdot x \leq \frac{U^2+2x^2}{Ux} + 3$

  $\Rightarrow$ $c = \infty$



Online Algorithmen

Online Bidding

Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

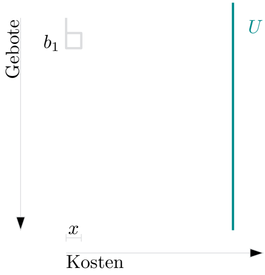
- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

$\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$

$\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$

$\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}(\frac{U}{x} + 2)(\frac{U}{x} + 1) \cdot x \leq \frac{U^2+2x^2}{Ux} + 3$

$\Rightarrow$ $c = \infty$



Online Algorithmen

Online Bidding

Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

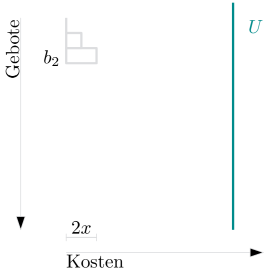
- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

  $\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$

  $\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$

  $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}\left(\frac{U}{x} + 2\right)\left(\frac{U}{x} + 1\right) \cdot x \leq \frac{U^2 + 2x^2}{Ux} + 3$

  $\Rightarrow$ $c = \infty$



Online Algorithmen

Online Bidding

Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

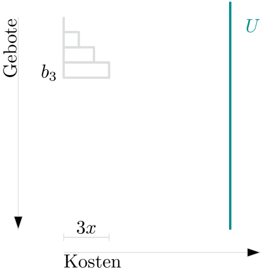
- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

  $\Rightarrow$ Anzahl Gebote: $k+1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$

  $\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$

  $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}\left(\frac{U}{x} + 2\right)\left(\frac{U}{x} + 1\right) \cdot x \leq \frac{U^2 + 2x^2}{Ux} + 3$

  $\Rightarrow$ $c = \infty$



Online Algorithmen

Online Bidding

Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

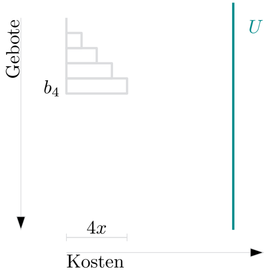
- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

  $\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$

  $\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$

  $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}(\frac{U}{x} + 2)(\frac{U}{x} + 1) \cdot x \leq \frac{U^2 + 2x^2}{Ux} + 3$

  $\Rightarrow$ $c = \infty$



Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

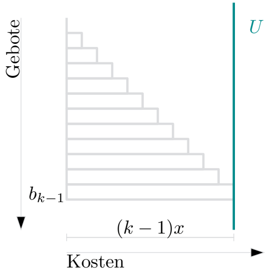
- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

  $\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$

  $\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$

  $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}\left(\frac{U}{x} + 2\right)\left(\frac{U}{x} + 1\right) \cdot x \leq \frac{U^2 + 2x^2}{Ux} + 3$

  $\Rightarrow$ $c = \infty$



Online Algorithmen

Online Bidding

Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

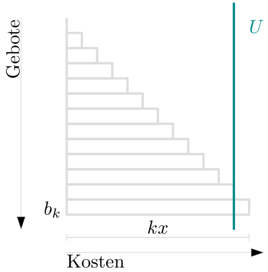
- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

$\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$

$\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$

$\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}(\frac{U}{x} + 2)(\frac{U}{x} + 1) \cdot x \leq \frac{U^2 + 2x^2}{Ux} + 3$

$\Rightarrow$ $c = \infty$



Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

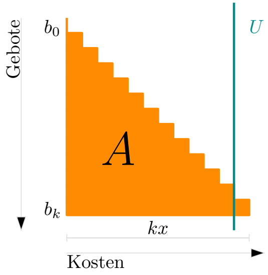
- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

$\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$

$\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$

$\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}(\frac{U}{x} + 2)(\frac{U}{x} + 1) \cdot x \leq \frac{U^2+2x^2}{Ux} + 3$

$\Rightarrow$ $c = \infty$



Online Algorithmen

Online Bidding

Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

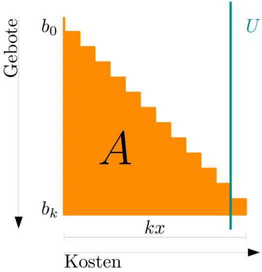
- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

  $\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$

  $\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$

  $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}\left(\frac{U}{x} + 2\right)\left(\frac{U}{x} + 1\right) \cdot x \leq \frac{U^2 + 2x^2}{Ux} + 3$

  $\Rightarrow$ $c = \infty$



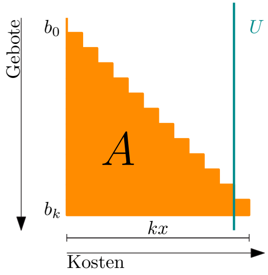
Fläche $A \approx \frac{1}{2}kx \cdot (k+1)$

Online Algorithmen

Online Bidding

Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$
  - $\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$
  - $\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$
  - $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}(\frac{U}{x} + 2)(\frac{U}{x} + 1) \cdot x \leq \frac{U^2+2x^2}{Ux} + 3$
  - $\Rightarrow$ $c = \infty$



Fläche $A \approx \frac{1}{2}kx \cdot (k+1)$

Online Algorithmen

Online Bidding

Analyse: (a) $b_i = i \cdot x$ $(x > 0)$

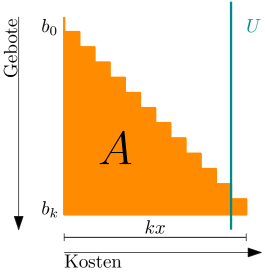
- *worst case*: $b_{k-1} = (k-1) \cdot x = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

  $\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\frac{U-\varepsilon}{x} + 1\right) + 1 \leq \frac{U}{x} + 2$

  $\Rightarrow$ $ALG = \sum_{i=0}^{k} b_i = \sum_{i=0}^{k} i \cdot x = \frac{1}{2}(k+1)k \cdot x$

  $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{1}{2U}(\frac{U}{x} + 2)(\frac{U}{x} + 1) \cdot x \leq \frac{U^2 + 2x^2}{Ux} + 3$

  $\Rightarrow$ $c = \infty$ (unabhängig von x schlecht)



Fläche $A \approx \frac{1}{2}kx \cdot (k+1)$

Online Algorithmen

Online Bidding

Analyse: (b) $b_i = a \cdot x^i$ $(x > 1)$

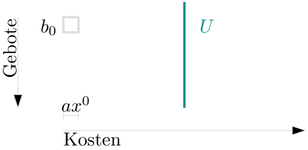
- *worst case*: $b_{k-1} = a \cdot x^{k-1} = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

$\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\log_x(\frac{U-\varepsilon}{a}) + 1\right) + 1 \leq \log_x \frac{U}{a} + 2$

$\Rightarrow$ $ALG = \sum_{i=0}^{k} a \cdot x^i = \frac{a \cdot (x^{k+1}-1)}{x-1} \leq \frac{a \cdot (x^{\log_x \frac{U}{a}+2}-1)}{x-1} = \frac{a \cdot (\frac{U}{a}x^2-1)}{x-1}$

$\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{a \cdot (\frac{U}{a}x^2-1)}{U(x-1)} = \frac{x^2}{x-1} - \frac{a}{(x-1)U}$

$\Rightarrow$ $c \leq \frac{x^2}{x-1}$



Online Algorithmen

Online Bidding

Analyse: (b) $b_i = a \cdot x^i$ $(x > 1)$

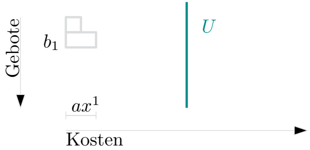
- *worst case*: $b_{k-1} = a \cdot x^{k-1} = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

$\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\log_x\left(\frac{U-\varepsilon}{a}\right) + 1\right) + 1 \leq \log_x \frac{U}{a} + 2$

$\Rightarrow$ $ALG = \sum_{i=0}^{k} a \cdot x^i = \frac{a \cdot (x^{k+1} - 1)}{x - 1} \leq \frac{a \cdot (x^{\log_x \frac{U}{a} + 2} - 1)}{x - 1} = \frac{a \cdot (\frac{U}{a} x^2 - 1)}{x - 1}$

$\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{a \cdot (\frac{U}{a} x^2 - 1)}{U(x - 1)} = \frac{x^2}{x - 1} - \frac{a}{(x - 1)U}$

$\Rightarrow$ $c \leq \frac{x^2}{x - 1}$



Online Algorithmen

Online Bidding

Analyse: (b) $b_i = a \cdot x^i$ $(x > 1)$

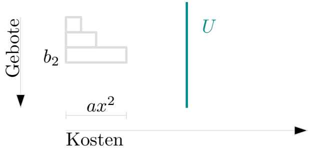
- *worst case*: $b_{k-1} = a \cdot x^{k-1} = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

  $\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\log_x\left(\frac{U-\varepsilon}{a}\right) + 1\right) + 1 \leq \log_x \frac{U}{a} + 2$

  $\Rightarrow$ $ALG = \sum_{i=0}^{k} a \cdot x^i = \frac{a \cdot (x^{k+1} - 1)}{x-1} \leq \frac{a \cdot (x^{\log_x \frac{U}{a} + 2} - 1)}{x-1} = \frac{a \cdot (\frac{U}{a} x^2 - 1)}{x-1}$

  $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{a \cdot (\frac{U}{a} x^2 - 1)}{U(x-1)} = \frac{x^2}{x-1} - \frac{a}{(x-1)U}$

  $\Rightarrow$ $c \leq \frac{x^2}{x-1}$



Online Algorithmen

Online Bidding

Analyse: (b) $b_i = a \cdot x^i$ $(x > 1)$

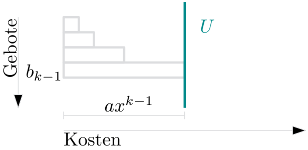
- *worst case*: $b_{k-1} = a \cdot x^{k-1} = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

  $\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\log_x\left(\frac{U-\varepsilon}{a}\right) + 1\right) + 1 \leq \log_x \frac{U}{a} + 2$

  $\Rightarrow$ $ALG = \sum_{i=0}^{k} a \cdot x^i = \frac{a \cdot (x^{k+1} - 1)}{x - 1} \leq \frac{a \cdot (x^{\log_x \frac{U}{a} + 2} - 1)}{x - 1} = \frac{a \cdot (\frac{U}{a} x^2 - 1)}{x - 1}$

  $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{a \cdot (\frac{U}{a} x^2 - 1)}{U(x-1)} = \frac{x^2}{x-1} - \frac{a}{(x-1)U}$

  $\Rightarrow$ $c \leq \frac{x^2}{x-1}$



Online Algorithmen

Online Bidding

Analyse: (b) $b_i = a \cdot x^i$ $(x > 1)$

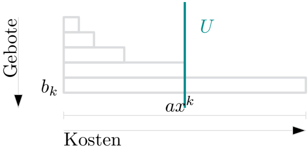
- *worst case*: $b_{k-1} = a \cdot x^{k-1} = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

$\Rightarrow$ Anzahl Gebote: $k+1 = \left(\log_x\left(\frac{U-\varepsilon}{a}\right)+1\right)+1 \leq \log_x \frac{U}{a} + 2$

$\Rightarrow$ $ALG = \sum_{i=0}^{k} a \cdot x^i = \frac{a\cdot(x^{k+1}-1)}{x-1} \leq \frac{a\cdot(x^{\log_x \frac{U}{a}+2}-1)}{x-1} = \frac{a\cdot(\frac{U}{a}x^2-1)}{x-1}$

$\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{a\cdot(\frac{U}{a}x^2-1)}{U(x-1)} = \frac{x^2}{x-1} - \frac{a}{(x-1)U}$

$\Rightarrow$ $c \leq \frac{x^2}{x-1}$



Online Algorithmen

Online Bidding

Analyse: (b) $b_i = a \cdot x^i$ $(x > 1)$

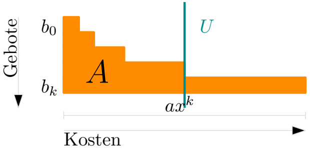
- *worst case*: $b_{k-1} = a \cdot x^{k-1} = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

  $\Rightarrow$ Anzahl Gebote: $k+1 = \left(\log_x\left(\frac{U-\varepsilon}{a}\right)+1\right)+1 \le \log_x \frac{U}{a}+2$

  $\Rightarrow$ $ALG = \sum_{i=0}^{k} a \cdot x^i = \frac{a\cdot(x^{k+1}-1)}{x-1} \le \frac{a\cdot(x^{\log_x \frac{U}{a}+2}-1)}{x-1} = \frac{a\cdot(\frac{U}{a}x^2-1)}{x-1}$

  $\Rightarrow$ $\frac{ALG}{OPT} \le \frac{a\cdot(\frac{U}{a}x^2-1)}{U(x-1)} = \frac{x^2}{x-1} - \frac{a}{(x-1)U}$

  $\Rightarrow$ $c \le \frac{x^2}{x-1}$



Online Algorithmen

Online Bidding

Analyse: (b) $b_i = a \cdot x^i$ $(x > 1)$

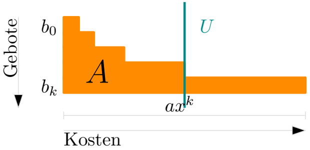
- *worst case*: $b_{k-1} = a \cdot x^{k-1} = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

$\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\log_x\left(\frac{U-\varepsilon}{a}\right) + 1\right) + 1 \leq \log_x \frac{U}{a} + 2$

$\Rightarrow$ $ALG = \sum_{i=0}^{k} a \cdot x^i = \frac{a\cdot(x^{k+1}-1)}{x-1} \leq \frac{a\cdot(x^{\log_x \frac{U}{a}+2}-1)}{x-1} = \frac{a\cdot(\frac{U}{a}x^2-1)}{x-1}$

$\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{a\cdot(\frac{U}{a}x^2-1)}{U(x-1)} = \frac{x^2}{x-1} - \frac{a}{(x-1)U}$

$\Rightarrow$ $c \leq \frac{x^2}{x-1}$



Fläche $A \approx \frac{a\cdot(x^{k+1}-1)}{x-1}$

Online Algorithmen

Online Bidding

Analyse: (b) $b_i = a \cdot x^i$ $(x > 1)$

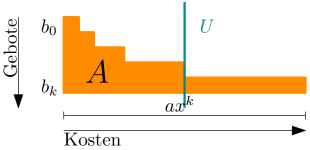
- *worst case*: $b_{k-1} = a \cdot x^{k-1} = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$

$\Rightarrow$ Anzahl Gebote: $k+1 = \left(\log_x(\frac{U-\varepsilon}{a}) + 1\right) + 1 \leq \log_x \frac{U}{a} + 2$

$\Rightarrow$ $ALG = \sum_{i=0}^{k} a \cdot x^i = \frac{a \cdot (x^{k+1}-1)}{x-1} \leq \frac{a \cdot (x^{\log_x \frac{U}{a}+2}-1)}{x-1} = \frac{a \cdot (\frac{U}{a}x^2-1)}{x-1}$

$\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{a \cdot (\frac{U}{a}x^2-1)}{U(x-1)} = \frac{x^2}{x-1} - \frac{a}{(x-1)U}$

$\Rightarrow$ $c \leq \frac{x^2}{x-1}$



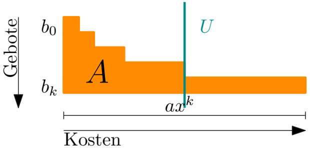
Fläche $A \approx \frac{a \cdot (x^{k+1}-1)}{x-1}$

Online Algorithmen

Online Bidding

Analyse: (b) $b_i = a \cdot x^i$ $(x > 1)$

- *worst case*: $b_{k-1} = a \cdot x^{k-1} = U - \varepsilon$ $(k > 0, \varepsilon \to 0)$
  - $\Rightarrow$ Anzahl Gebote: $k + 1 = \left(\log_x\left(\frac{U-\varepsilon}{a}\right) + 1\right) + 1 \leq \log_x \frac{U}{a} + 2$
  - $\Rightarrow$ $ALG = \sum_{i=0}^{k} a \cdot x^i = \frac{a \cdot (x^{k+1} - 1)}{x-1} \leq \frac{a \cdot (x^{\log_x \frac{U}{a} + 2} - 1)}{x-1} = \frac{a \cdot (\frac{U}{a} x^2 - 1)}{x-1}$
  - $\Rightarrow$ $\frac{ALG}{OPT} \leq \frac{a \cdot (\frac{U}{a} x^2 - 1)}{U(x-1)} = \frac{x^2}{x-1} - \frac{a}{(x-1)U}$
  - $\Rightarrow$ $c \leq \frac{x^2}{x-1}$ (minimal für $x = 2 \Rightarrow c \leq 4$)



Fläche $A \approx \frac{a \cdot (x^{k+1} - 1)}{x-1}$

Online Algorithmen

Zusammenfassung

Wissenswert (nicht erschöpfend!)

- kompetitiver Faktor c
- *Doubling* Technik
- *Ski Rental* Problem

Typische Fragestellungen

- Gegeben sei Algorithmus X. Wie groß ist sein kompetitiver Faktor?
- Geben Sie einen Algorithmus mit kompetitivem Faktor c an.
- Bestimme untere Schranke für kompetitiven Faktor c von Problem P.

Ende!

Feierabend!