Some Extra Slides on Chernoff Bounds

Chernoff Bounds (above mean)

Theorem. Suppose $X_1, \ldots, X_n$ are independent 0-1 random variables. Let $X = X_1 + \ldots + X_n$. Then for any $\mu \geq E[X]$ and for any $\delta > 0$, we have

$$\Pr[X > (1+\delta)\mu] < \left[\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right]^{\mu}$$

↑ sum of independent 0-1 random variables is tightly centered on the mean

Pf. We apply a number of simple transformations.

- For any $t > 0$,

$$\Pr[X > (1+\delta)\mu] = \Pr\left[e^{tX} > e^{t(1+\delta)\mu}\right] \leq e^{-t(1+\delta)\mu} \cdot E[e^{tX}]$$

↑ $f(x) = e^{tx}$ is monotone in x

↑ Markov's inequality: $\Pr[X > a] \leq E[X] / a$

- Now $E[e^{tX}] = E[e^{t\sum_i X_i}] = \prod_i E[e^{tX_i}]$

↑ definition of X ↑ independence

Chernoff Bounds (above mean)

Pf. (cont)

- Let $p_i = \Pr[X_i = 1]$. Then,

$$E[e^{tX_i}] = p_i e^t + (1-p_i)e^0 = 1 + p_i(e^t - 1) \le e^{p_i(e^t-1)}$$

↑ for any $\alpha \ge 0$, $1+\alpha \le e^{\alpha}$

- Combining everything:

$$\Pr[X > (1+\delta)\mu] \le e^{-t(1+\delta)\mu} \prod_i E[e^{tX_i}] \le e^{-t(1+\delta)\mu} \prod_i e^{p_i(e^t-1)} \le e^{-t(1+\delta)\mu}\, e^{\mu(e^t-1)}$$

↑ previous slide ↑ inequality above ↑ $\sum_i p_i = E[X] \le \mu$

- Finally, choose $t = \ln(1+\delta)$. ▪

Chernoff Bounds (below mean)

Theorem. Suppose $X_1, \ldots, X_n$ are independent 0-1 random variables. Let $X = X_1 + \ldots + X_n$. Then for any $\mu \leq E[X]$ and for any $0 < \delta < 1$, we have

$$\Pr[X < (1-\delta)\mu] < e^{-\delta^2 \mu / 2}$$

Pf idea. Similar.

Remark. Not quite symmetric since only makes sense to consider $\delta < 1$.

Application of Chernoff Bounds to a Load Balancing Problem

Load Balancing

Load balancing. System in which m jobs arrive in a stream and need to be processed immediately on n identical processors. Find an assignment that balances the workload across processors.

Centralized controller. Assign jobs in round-robin manner. Each processor receives at most $\lceil m/n \rceil$ jobs.

Decentralized controller. Assign jobs to processors uniformly at random. How likely is it that some processor is assigned "too many" jobs?

Load Balancing

Analysis (for m=n).

- Let X_i = number of jobs assigned to processor i.
- Let Y_{ij} = 1 if job j assigned to processor i, and 0 otherwise.
- We have $E[Y_{ij}] = 1/n$
- Thus, $X_i = \sum_j Y_{ij}$, and $\mu = E[X_i] = 1$.
- Applying Chernoff bounds with $\delta = c - 1$ yields $\Pr[X_i > c] < \dfrac{e^{c-1}}{c^c}$
- Let $\gamma(n)$ be number x such that $x^x = n$, and choose $c = e\,\gamma(n)$.

$$\Pr[X_i > c] < \frac{e^{c-1}}{c^c} < \left(\frac{e}{c}\right)^c = \left(\frac{1}{\gamma(n)}\right)^{e\gamma(n)} < \left(\frac{1}{\gamma(n)}\right)^{2\gamma(n)} = \frac{1}{n^2}$$

- Union bound $\Rightarrow$ with probability $\geq 1 - 1/n$ no processor receives more than $e\,\gamma(n) = \Theta(\log n / \log \log n)$ jobs.

Fact: this bound is asymptotically tight: with high probability, some processor receives $\Theta(\log n / \log \log n)$

Load Balancing: Many Jobs (m > n)

Theorem. Suppose the number of jobs m = 16n ln n. Then on average, each of the n processors handles μ = 16 ln n jobs. With high probability every processor will have between half and twice the average load.

Pf.

- Let X_i , Y_{ij} be as before.
- Applying Chernoff bounds with δ = 1 yields

$$\Pr[X_i > 2\mu] < \left(\frac{e}{4}\right)^{16\ln n} < \left(\frac{1}{e^2}\right)^{\ln n} = \frac{1}{n^2} \qquad \Pr[X_i < \tfrac{1}{2}\mu] < e^{-\frac{1}{2}\left(\frac{1}{2}\right)^2(16\ln n)} = \frac{1}{n^2}$$

- Union bound $\Rightarrow$ every processor has load between half and twice the average with probability $\geq$ 1 - 2/n. ▪

Kapitel 7: Online Algorithms

Kapitel 7: Online Algorithms

Inhalt:

- Deterministische Online Algorithmen
  - Einführung
  - Paging
  - Scheduling
- Randomisierte Online Algorithmen
  - Paging
  - Finanzielle Spiele

Beispiel 1

Skifahrer-Problem:

- Ein Paar Ski kann für 500 € gekauft oder für 50 € pro Tag geliehen werden. Wie viel Tage sollte man sich Skier leihen, bevor man sich für den Kauf entscheidet, wenn nicht bekannt ist, wie viel Tage man noch Ski fährt?

- Optimale Strategie (falls Zukunft nicht bekannt): So lange Ski leihen, bis die Leihkosten gleich den Kaufkosten sind.

Beispiel 2

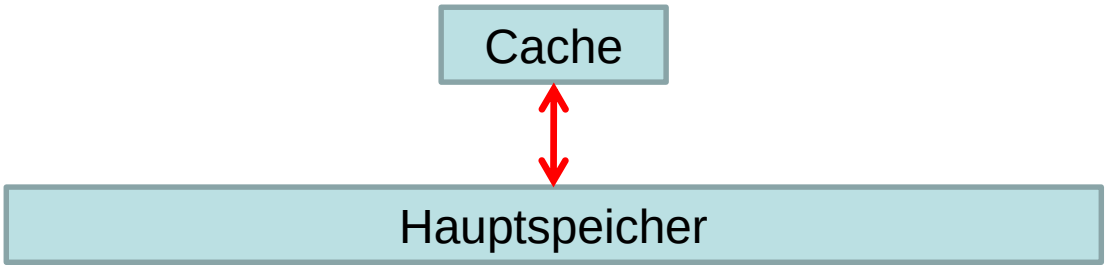
Geldanlageproblem:

- Ein Geldbetrag (z.B. 10.000 €) soll in eine andere Währung (z.B. $) gewechselt werden. Zu welchem Zeitpunkt soll getauscht werden, wenn nicht bekannt ist, wie sich der Wechselkurs entwickelt?

Beispiel 3

Paging/Caching:

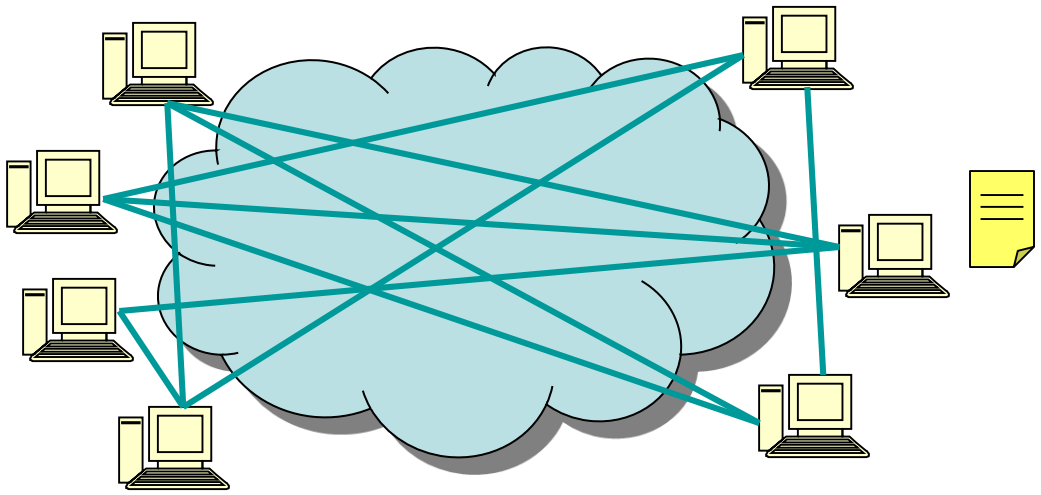
- Es soll ein zweistufiges Speichersystem verwaltet werden, das aus einem schnellen Speicher mit kleiner Kapazität und einem langsamen Speicher mit großer Kapazität besteht. Dabei müssen Anfragen auf Speicherseiten bedient werden. Welche Seiten hält man im Cache, um die Anzahl der Cache-Misses zu minimieren?



Beispiel 4

Verteilte Systeme – „Akamai´s Business“:

- Daten sollen in einem Netzwerk dynamisch so platziert werden, dass möglichst wenige Daten übertragen werden müssen, um die Anfragen zu bedienen.



Beispiel 5

Scheduling / Load Balancing:

- Jobs mit im Voraus bekannter Bearbeitungsdauer treffen hintereinander ein und sollen von m Maschinen bearbeitet werden. Die Jobs müssen dabei unmittelbar einer bestimmten Maschine zugeordnet werden. Optimierungsziel sei hierbei beispielsweise wiederum die Minimierung der Gesamtbearbeitungszeit.

Notation

Online Problem:

- Statt einer vollständig gegebenen Eingabe I haben wir jetzt eine Eingabesequenz $\sigma=(\sigma(1), \sigma(2),\ldots, \sigma(t))$, in der die Eingabeteile $\sigma(i)$ erst nach und nach an den Online Algorithmus übergeben werden.
- Nach jedem $\sigma(i)$ muss der Online Algorithmus eine Entscheidung treffen, bevor er $\sigma(i+1)\ldots\sigma(t)$ gesehen hat. Diese Entscheidung kann (im Standard-Online-Modell) nicht wieder zurück-genommen werden.

Notation

Definition: Sei Π ein Minimierungsproblem und A ein Online Algorithmus für Π. Für eine beliebige Eingabesequenz $\sigma=(\sigma(1), \sigma(2),\ldots, \sigma(t))$ seien $A(\sigma)$ die Kosten von A für σ.

A heißt c-kompetitiv, wenn es einen von t unabhängigen Parameter a gibt, so dass für alle Eingabesequenzen σ gilt:

$$A(\sigma) \leq c \cdot OPT(\sigma) + a$$

Fragen?