Misc

Ting vi ikke gik igennem, eller ikke er en del af spørgsmålene

Kevin Vinther

January 4, 2024

Table of Contents i

Table of Contents ii

Misc 1

- Det her er bare et simpelt kombinatorisk bevis af Jørgen.

Theorem

If S is a finite set with at least one element, then the number of even subsets of S is the same as the number of odd subsets of S.

- Der kommer to beviser.
- Først, læg mærke til at dette ikke gælder når $S = \emptyset$.
- Lad E_S være antallet af lige subsets og O_S antallet af ulige subsets af S.
- Sidst, lad $n = |S|$ (i.e., kardinaliteten af S.)
- **Første bevis:**

Notes on Combinatorial Proofs ii

- Der er præcis $\binom{n}{k}$ måder at vælge et sæt med k elementer fra S.
- Husk binomial theorem: $(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^{n-k} y^k$.
- Hvis vi siger at $x = 1$ og $y = -1$ får vi $0 = \sum_{k=0}^{n} \binom{n}{k} (-1)^k$.
- Vi kan dermed se at hvert lige subset bliver 1, og hvert ulige bliver -1.
- **Andet bevis:**
- Teoremet holder klart når $n = 1$, så tænk på et sæt S med $n > 1$ elementer.
- Vi "fikser" et element $s \in S$ og lader $S' = S \setminus \{s\}$.
- Lad e_s, o_s være antallet af lige og ulige subsets af S som indeholder s (dvs. det tal vi "fiksede" før.)

Notes on Combinatorial Proofs iii

- Hvert lige (ulige) subset af S som har s i sig, har s plus et ulige (lige) subset af S'.
- Dermed har vi $e_s = O_{S'}$ og $o_s = E_{S'}$.
- Til sidst, observér at, gennem sum-reglen, antallet af lige (ulige) subsets fa S er lig med antallet af lige (ulige) subsets der indeholder s plus antallet af lige (ulige) subsets af S der ikke indeholder s.
- Dermed:

$$E_S = e_s + E_{S'} = O_{S'} + E_{S'} = O_{S'} + o_s = O_S$$

Andet teorem i

- Vi giver nu endnu et eksempel af hvordan kombinatoriske argumenter kan bruges.
- For givne heltal k, n lader vi
  $S_{n,k} = \{(n_1, n_2, \ldots, n_k) | n_i \geq 0 \text{ og } n_1 + n_2 + \cdots + n_k = n\}$
- Læg mærke til at $S_{n,k}$ er sættet af alle ordnet (ordered) k-tupler, med et ikke-negativt tal for hvilke summen af elementerne i tuplen er n.
- Fra Rosen 6.5.3 ved vi at der er $\binom{n+(k-1)}{n}$ af disse.

Andet teorem ii

Theorem

$$\sum_{(n_1,n_2,\ldots,n_k)\in S_{n,k}} \frac{n!}{n_1! \cdot n_2! \cdot \cdots \cdot n_k!} = k^n$$

- **Bevis:**
- Vi påstår at begge sider af lighedstegnet tæller antallet af måder at distribuere n distinkte bolde i k distinkte bokse.
- Det er nemt at se på højresiden: vi har k valg for hver af de n bolde, så k^n i alt.
- For venstresiden tæller vi det samme. Af Rosen Theorem 4 side 452 ved vi at for fiksede $n_1, n_2, \ldots, n_k$ således at $\sum_{i=1}^{k} n_i = n$ er der $\frac{n!}{n_1! \cdot n_2! \cdot \cdots \cdot n_k!}$ måder at distribuere dem.

Kombinatoriske Beviser i

- Håber I er klar gutter!

Definition

A *combinatorial proof* of an identity is a proof that uses counting arguments to prove that both sides of the identity count the same objects but in different ways or a proof that is based on showing that there is a bijectino between the sets of objects counted by the two sides of the identity. These two types of proofs are called *double counting proofs* and *bijective proofs* respectively.

- Så, med **dobbelttælningsbeviser** er målet af vise at begge sider af en identitet i essensen tæller det samme sæt af objekter (se f.eks. $\binom{n}{k} = \binom{n}{n-k}$).

Kombinatoriske Beviser ii

- Et **bijektivt bevis** arbejder ved at etablere en en-til-en korrespondance mellem de to sæt, som tælles. Det skal bekræftes at hvert element i et sæt svarer **præcist** til et objekt i et andet sæt. Hvis dette gælder, tæller de det samme, og er dermed lig hinanden.
- Lad os, for eksempel, kigger på beviserne for $\binom{n}{r} = \binom{n}{n-r}$.
- Antag at S er et sæt med n elementer.
- Funktionen som "mapper" et subset A af S til $\overline{A}$ er en bijketion mellem subsetsne af S med r elementer, og subsetsne med $n - r$ elementer.
- Du kan tænke på det på den her måde: For hvert sæt der bliver talt i A bliver der udeladt et antal af elementer, $|\overline{A}|$.

Kombinatoriske Beviser iii

- Disse elementer bliver undlandt lige så mange gange, som du tæller de originale tal.
- **Dobbelttælningsbevis**:[1] Af deginitionen er antallet af subsets af S med r elementer $\binom{n}{r}$. Men hvert subset A af S bliver også valgt ved at specificere hvilke elementer der **ikke** er en del af A og dermed er i $\overline{A}$.

Theorem (1. The Binomial Theorem)

Let x and y be variables, and let n be a nonnegative integer. Then

$$(x+y)^n = \sum_{j=0}^{n} \binom{n}{j} x^{n-j} y^j = \binom{n}{0} x^n \binom{n}{1} x^{n-1} y + \cdots + \binom{n}{n-1} x y^{n-1} + \binom{n}{n}$$

- Vi beviser kombinatorisk.
- Leddene i produktet, når de bliver udvidet, er af formen $x^{n-j}y^j$ for $j = 0, 1, 2, \ldots, n$.

- For at tælle antallet af led af formen $x^{n-j}y^j$, notér at for at få sådan et led skal man vælge $n-j$ x'er fra de n binnomiale faktorer (således at de andre j led i produktet er y'er).
- Dermed er koefficienten af $x^{n-j}y^j = \binom{n}{n-j}$, hvilket er lig med $\binom{n}{j}$. Dette beviser teoremet.

Theorem (2. Pascal's Identity)

Let n and k be positive integers with $n \geq k$. Then

$$\binom{n+1}{k} = \binom{n}{k-1} + \binom{n}{k}$$

.

Kombinatoriske Beviser vi

- Igen beviser vi kombinatorisk (haha, man skulle næsten tro at det var målet med den her fremlæggelse.)
- Antag at T er et sæt tilhørende $n+1$ elementer.
- Lad a være et element i T og lad $S = T - \{a\}$.
- Læg mærke til at der er $\binom{n+1}{k}$ subsets af T med k elementer.
- (igen, $n+1$ fordi T er defineret, ikke til at have n elementer, men $n+1$)
- Et subset af T med k elementer indeholder enten a sammen med $k-1$ elementer af S, eller k elementer af S og inderholder ikke a.
- Fordi der er $\binom{n}{k-1}$ subsets af $k-1$ elementer af S, er der $\binom{n}{k-1}$ subsets af k elementer af T der indeholder a.

- Der er $\binom{n}{k}$ subsets af k elementer af T som ikke indeholder a, fordi der er $\binom{n}{k}$ subsets af k elementer af S.
- Dermed teoremet.

Theorem (3. Vandermonde's Identity)

Let m, n and r be nonnegative integers with $r \leq m, n$. Then

$$\binom{m+n}{r} = \sum_{k=0}^{r} \binom{m}{r-k}\binom{n}{k}$$

- Antag at der er m elementer i et sæt, og n i et andet.
- Så er der i alt $\binom{m+n}{r}$ måder at vælge r elementer fra begge sæt.

Kombinatoriske Beviser viii

- En anden måde at vælge r elementer fra fællesmængden er at vælge k elementer fra det andet sæt, og så $r - k$ elementer fra det første sæt, hvor k er et heltal $0 \leq k \leq r$.
- Fordi der er $\binom{n}{k}$ måder at vælge k elementer fra det andet sæt på, og $\binom{m}{r-k}$ måder at vælge $r - k$ elementer fra det første sæt, fortæller product rule os at det kan blive gjort på $\binom{m}{r-k}\binom{n}{k}$ måder.
- Dermed er det fulde antal af måder du kan vælge r elementer på fra fællesmængden lig med

$$\sum_{k=0}^{r} \binom{m}{r-k} \binom{n}{k}$$

- Vi har fundet du udtryk for antallet af måder at vælge r elementer på fra fællesmængden af et sæt med m elementer, og et sæt med n elementer.
- Dette beviser Vandermonde's Identitet

Theorem (4)

Let n and r be nonnegative integers with $r \leq n$. Then

$$\binom{n+1}{r+1} = \sum_{j=r}^{n} \binom{j}{r}$$

- Et tidligere eksempel har vist at $\binom{n+1}{r+1}$ tæller antallet af bit-strenge af længde $n+1$ der har $r+1$ et'taller

- Vi vil vise at højresiden tæller de samme objekter ved at se på antallet af korresponderende mulige lokationer af det sidste 1 i en streng med $r+1$ 1'ere.
- Det sidste 1 må være ved position $r+1, r+2, \ldots,$ eller $n+1$.
- Ydermere, hvis det sidste et er det k'e bit, må der være r et'ere imellem de første $k-1$ positioner.
- Vi ved at der er $\binom{k-1}{r}$ af disse slags bitstrenge.
- Hvis vi summerer over k med $r+1 \leq k \leq n+1$, finder vi at der er

$$\sum_{k=r+1}^{n+1} \binom{k-1}{r} = \sum_{j=r}^{n} \binom{j}{r}$$

bit strenge af længde n med præcis $r+1$ et'taller.

[1] jeg tænker bare jeg skriver double countign fra nu af, det er godt nok irriterende at skrive et så langt ord.

Recurrence Relations

- En recurrence relation er en rekursiv definition med mere end et initial term.
- En sekvens er en løsning af recurrence relationen hvis dets led satisfy relationen.
- **Eksempel med fibonacci:**
  $f_n = f_{n-1} + f_{n-2}, n \geq 3, f_1 = 1, f_2 = 1.$
- **Eksempel: Tower of Hanoi**

Algorithms and Recurrence Relations i

- Recurrence Relations kan bruges til at finde kompleksiteten af divide-and-conquer algoritmer.
- Vi introducerer nu **Dynamic Programming**.
- Dynamic Porgramming er et paradigme en algoritme følger, hvis den rekursivt "breaker down" et problem i mindre, men overlappende subproblemer, og så udregner løsningne ved brug af løsningerne af subproblemerne.

Solving Linear Recurrence Relations i

- En vigtigt klasse af recurrence relations kan blive løst på en systematisk måde.

Definition

A *linear homogeneous recurrence relation of degree k with constant coefficients* is a recurrence relation of the form

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdots + c_k a_{nk}$$

where $c_1, c_2, \ldots, c_k$ are real numbers, and $c_k \neq 0$.

- Den er **lineær** fordi højresiden er summen af de tidligere led af sekvensen, hvert led ganget med en funktion af n.

Solving Linear Recurrence Relations ii

- Den er **homogen** fordi ingen led forekommer som er multiplum af a_j'erne.
- Koefficienterne i sekvensen er alle **konstanter**, i stedet for funktioner der afhænger af n.
- **Degreeen** (dansk?) er k fordi a_n er udtrykt ved brug af de tidligere k led i sekvensen.

Eksempel på linear recurrence

$P_n = (1.11)P_{n-1}$ er et eksempel på en homogen rekursionsligning (siger chatgpt det hedder på dansk, ret mig lige hvis jeg tager fejl). Ligningen har "degree" 1. Fibonacci $f_n = f_{n-1} + f_{n-2}$ er også lineær, men med en degree på to. $a_n = a_{n-5}$ har en degree af 5.

Conenction between Stirling Numbers and the number of onto functions i

- Stirling Numbers $S(m, n)$ er antallet af måder man kan distribuere m distinguishable elementer i n ikke-distinguishable bokse, således at hver boks har mindst et element.
- VI laver en onto funktion fra et sæt X af m elementer og et sæt Y af n elementer.
- Tænk på et andet problem hvor vi har n bokse der kan skælnes mellem, navngivet 1, 2, n. Ignorer derefter navnene
- Derefter gør følgenede:
    1. Distribuer elementer fra X i de n bokse så der er ingen tomme

Conenction between Stirling Numbers and the number of onto functions ii

2. Kig på navnene af boksene og få en funktion f fra X til Y ved at mappe et element $x \in X$ til y_i hvis det var palceret i boksen hvis navn fra i.

- På denne måde, for hver måde man kan lave step 1 på, er der $n!$ onto funktioner, nemlig antallet af måder vi kan navngive de n bokse.
- Dette viser at antallet af onto funktioner fra et sæt med m elementer til et sæt af n elementer er præcis $n! \cdot S(m, n)$
- Det følger dermed at:

$$S(m,n) = \frac{1}{n!}\left[n^m - \binom{n}{1}(n-1)^m + \binom{n}{2}(n-2)^m - \cdots + (-1)^{n-1}\right.$$

Ramsey Number i

- Den probabilistiske metode er en teknik brugt til at lave nonkonstrutive eksistensbeviser.
- Nonkonstruktiv = Du viser ikke hvordan man finder elementet
- Eksistensbevis = du beviser at elementet eksisterer
- For at bevise at et element i et sæt S eksisterer, giver vi sandsynligheder til elementerne i S med en specifik property.
- Vi kan så vise at et element eksisterer ved at vise at sandsynligheden for at $x \in S$ har en property er positiv.

Ramsey Number ii

Theorem (3. The Probabilistic Method)

If the probability that an element chosen at random from a S does not have a particular property is less than 1, there exists an elementin S with this property.

- Det er herfra tydeligt at se hvorfor et bevis baseret på probabilistic method er nonkonstruktivt.
- Før vi viser et lower bound på Ramsey refreshes Ramsey Number.

Ramsey Numbers i

- Ramsey Tallet $R(m, n)$ hvor m og n er positive heltal større end eller lig med 2, er det minimum antal af personer til en fest således at der er enten m fælles venner eller n fælles fjender.

Example (13)

Assume that in a group of six people, each pair of individuals consists of two frients or two enemies. Show that there are either three mutual friends or three mutual enemies in the group.

- Altså, for hvert par af 2 personer kan de enten være fjender eller venner.

Ramsey Numbers ii

- Eksempler siger at vi skal vise at i en gruppe af 6 personer er der enten 3 fælles venner eller 3 fælles fjender.
- Løsning:
- Lad A være en af de 6 personer.
- Af de andre 5 personer i gruppen er der enten 3 eller flere som er venner af A, eller tre eller flere som er fjender af A.
- Dette følger fra dueslagsprincippet. $\lceil 5/2 \rceil = 3$

Bound v.h.a Probabilistic Method i

Theorem (4)

If k is an integer with $k \geq 2$, then $R(k,k) \geq 2^{k/2}$

- Bevis:
- Vi ved at teoremet holder ved $k = 2$ og $k = 3$ fordi $R(2,2) = 2$ og $R(3,3) = 6$, som vist tidligere. Antag nu at $k \geq 4$.
- Vi vil bruge probabilistik metode til at vise at hvis der er færre end $2^{k/2}$ personer til en fest, er det muligt at ingen af de k er fælles venner eller fælles fjender, dermed kan vi konkludere at $R(k,k) \geq 2^{k/2}$.

Bound v.h.a Probabilistic Method ii

- Vi antager at chancen for at to venner er fjender er lig chancen for at to personer er venner.
- Antag at der er n personer til festen.
- Det følger at der er $\binom{n}{k}$ forskellige sæt af k personer til denne fest.
- Vi lader det skrives $S_1, S_2, \ldots, S_{\binom{n}{k}}$.
- Lad E_i være begivenheden at alle k personer i S_i er enten fælles venner eller fælles fjender.
- Sandsynligheden at der er enten k fælles venner eller k fælels fjender imellem de n personer er lig $p(\bigcup_{i=1}^{\binom{n}{k}} E_i)$

Bound v.h.a Probabilistic Method iii

- Ifølge vores antagelse er der lige sandsynlighed for to personer til at være venner eller fjender.
- Ydermere er der $\binom{k}{2} = k(k-1)/2$ par af personer i S_i fordi der er k personer i S_i.
- Dermed er sandsynligheden for at alle k personer i S_i er fælles venner og resp. fælles fjender $(1/2)^{k(k-1)/2}$.
- Det følger at $P(E_i) = 2(1/2)^{k(k-1)/2}$
- Dermed er sandsynligheden for at der er enten k fælles venner eller k fælles fjender i en gruppe af n personer lig med $p(\bigcup_{i=1}^{\binom{n}{k}} E_i)$.

Bound v.h.a Probabilistic Method iv

- Hvis vi bruger Sum Bound får vi at:

$$p\left(\bigcup_{i=1}^{\binom{n}{k}} E_i\right) \leq \sum_{i=1}^{\binom{n}{k}} p(E_i) = \binom{n}{k} \cdot 2\left(\frac{1}{2}\right)^{k(k-1)/2}$$

- Fra en tidligere exercise (21, 6.4) Har vi at $\binom{n}{k} \leq n^k/2^{k-1}$. Dermed:

$$\binom{n}{k} 2\left(\frac{1}{2}\right)^{k(k-1)/2} \leq \frac{n^k}{2^{k-1}} 2\left(\frac{1}{2}\right)^{k(k-1)/2}$$

Bound v.h.a Probabilistic Method v

- Vi vil gerne vise at $R(n,n) \leq 2^{n/2}$ dermed, hvis vi antager at $n < 2^{k/2}$ har vi at

$$\frac{n^k}{2^{k-1}} 2 \left(\frac{1}{2}\right)^{k(k-1)/2} < \frac{2^{k(k/2)}}{2^{k-1}} 2 \left(\frac{1}{2}\right)^{k(k-1)/2} = 2^{2-(k/2)} \leq 1$$

- Det sidste led følger fordi $k \geq 4$.

Linearity of Expectations i

Theorem (3)

Let $X_i, i = 1, 2, \ldots, n$ with n a positive integer, be random variables on S, and let a and b be real numbers. Then:

(i) $E(X_1 + X_2 + \cdots + X_n) = E(X_1) + E(X_2) + \cdots + E(X_n)$

(ii) $E(aX + b) = aE(X) + b$

- Vi fokuserer på (ii).

Linearity of Expectations ii

$$
\begin{aligned}
E(aX+b) &= \sum_{s\in S} p(s)(aX(s)+b) \\
&= a\sum_{s\in S} p(s)X(s) + b\sum_{s\in S} p(s) \\
&= aE(X)+b \quad \text{because} \quad \sum_{s\in S} p(s) = 1
\end{aligned}
\tag{1}
$$

bevis

Proof: To prove this formula, we use the key observation that the event $XY = r$ is the disjoint union of the events $X = r_1$ and $Y = r_2$ over all $r_1 \in X(S)$ and $r_2 \in Y(S)$ with $r = r_1 r_2$. We have

$$E(XY) = \sum_{r \in XY(S)} r \cdot p(XY = r) \qquad \text{by Theorem 1}$$

$$= \sum_{r_1 \in X(S), r_2 \in Y(S)} r_1 r_2 \cdot p(X = r_1 \text{ and } Y = r_2) \qquad \text{expressing } XY = r \text{ as a disjoint union}$$

$$= \sum_{r_1 \in X(S)} \sum_{r_2 \in Y(S)} r_1 r_2 \cdot p(X = r_1 \text{ and } Y = r_2) \qquad \text{using a double sum to order the terms}$$

$$= \sum_{r_1 \in X(S)} \sum_{r_2 \in Y(S)} r_1 r_2 \cdot p(X = r_1) \cdot p(Y = r_2) \qquad \text{by the independence of } X \text{ and } Y$$

$$= \sum_{r_1 \in X(S)} \Big(r_1 \cdot p(X = r_1) \cdot \sum_{r_2 \in Y(S)} r_2 \cdot p(Y = r_2) \Big) \qquad \text{by factoring out } r_1 \cdot p(X = r_1)$$

$$= \sum_{r_1 \in X(S)} r_1 \cdot p(X = r_1) \cdot E(Y) \qquad \text{by the definition of } E(Y)$$

$$= E(Y) \Big(\sum_{r_1 \in X(S)} r_1 \cdot p(X = r_1) \Big) \qquad \text{by factoring out } E(Y)$$

$$= E(Y)E(X) \qquad \text{by the definition of } E(X).$$

Misc 2

Chernoff Bounds i

- :(
- Lad X være et random variable som er summen af flere uafhængige $0-1$-værdi random variables:
  $X = X_1 + X_2 + \cdots + X_n$ hvor X_i tager værdien 1 med sandsynlighed p_i og 0 ellers.
- Vi ved at $E(X) = \sum_{i=1}^{n} p_i$.
- Intuitivt, siden X_i er uafhængige, tænker man nok at de vil "cancel out" så deres sum er lig deres expected value med ret stor sandsynlighed.
- Vi bruger Chernoff Bounds til at finde denne sandsxynlighed.

Chernoff Bounds ii

- Det kan finde sandsynligheden for både et upper og lower bound.

Theorem

Let $X, X_1, X_2, \ldots, X_n$ be defined as above, and assume that $\mu \geq E(X)$. Then, for any $\delta > 0$, we have

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

- Så, i sin essens, er Chernoff Bounds sandsynligheden for at en værdi deviater meget fra $E(X)$.

Chernoff Bounds iii

- μ er oftest bare $E(X)$ (very often, ifølge Jørgen), så længe $\mu \geq E(X)$ er det ok.
- Givet følgende observation:

$$\forall t > 0 \;\; p(X > (1+\delta)\mu) = p(e^{tX} > e^{t(1+\delta)\mu})$$

  da e^{ty} er monotont increasing med y (altså, når y bliver større, bliver hele værdien større).
- Så dermed siger den bare at sandsynlighederne er de samme, da vi erstatter y med hhv. X og $(1+\delta)\mu$.
- Så; hvorfor gider vi gøre det endnu mere kompliceret at kigge på?

Chernoff Bounds iv

- Fordi den eksponentielle funktion har nogle dejlige properties vi kan bruge.
- Vi ved fra Markov's inequality at

$$p(Y > \gamma) \leq \frac{E(Y)}{\gamma}$$

og

$$\gamma p(Y > \gamma) \leq E(Y)$$

- Fra dette kan vi finde:

$$p(X > (1+\delta)\mu) = p(e^{tX} > e^{t(1+\delta)\mu}) \leq e^{t(1+\delta)\mu} E[e^{tX}]$$

Chernoff Bounds v

- Nu vil vi så gerne bounde $E[e^{tX}]$:

$$E(e^{tX}) = E(e^{t\sum x_i} = E(e^{\sum tX_i}) = E(\prod_{i=1}^{n} e^{tX_i}) = \prod_{i=1}^{n} E(e^{tX_i})$$

- Siden X_i er et indicator random variable

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

- Så

$$E(e^{tX_i}) \leq e^{p_i(e^t - 1}$$

- Da

$$1 + x \leq e^x$$

så længe $x \geq 0$

Chernoff Bounds vi

- Nu har vi et upper bound. Det vil vi gerne sætte ind:

$$
\begin{aligned}
E(e^{tX}) = \prod_{i=1}^{n} e^{tX_i} &\leq \prod_{i=1}^{n} e^{P_i(e^t-1} \\
&= e^{\sum p_i(e^t-1} \\
&= e^{(e^t-1)\Sigma p_i} \\
&\leq e^{(e^t-1)\mu}
\end{aligned}
$$

(2)

Chernoff Bounds vii

- Fordi $\sum p_i = E(X) \leq \mu$
- Vi kan nu få det ind således at
  $p(X > (1+\delta)\mu) \leq e^{-t(1+\delta)\mu} \cdot E(e^{tX})$
- Vi sætter $p(X > (1+\delta)\mu) \leq e^{-t(1+\delta)\mu} \cdot e^{(e^t-1)\mu}$
- Dette holder for alle $t > 0$, så hvis $t = \ln(1+\delta)$ får vi

$$\begin{aligned} p(X > (1+\delta)\mu) &\leq e^{-\ln(1+\delta)\cdot(1+\delta)\mu} \cdot e^{(e^{\ln(1+\delta)}-1)\mu} \\ &= (1+\delta)^{-(1+\delta)\mu} \cdot e^{(1+\delta-1)\mu} \\ &= \left[\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}}\right]^{\mu} \end{aligned}$$

Chernoff Bounds viii

- Og det var beviset:)
- Men vi er ikke færdige!
- Det kan også vises at X er langt mindre (lower bound).

$$p(X < (1-\delta)\mu) < e^{-\frac{1}{2}\mu\delta^2}$$

- Nogle formler der er nemmere at bruge:

$$p(X > (1+\delta)\mu) \leq e^{-\frac{\delta}{3}^2}\mu \text{ når } 0 < \delta$$

$$p(X < (1-\delta)\mu) \leq e^{-\frac{\delta}{2}^2}\mu \text{ når } 0 < \delta < 1$$

The Problem i

- Antag at vi har n processer, $P_1, P_2, \ldots, P_n$, hver af dem konkurrerer for adgang til en delt database.
- Vi antager at tiden er opdelt i *runder*.
- Databasen har den egenskab at den kun kan blive accessed af én process per runde, og hvis mere end én proces forsøger så er alle låst ude.
- Så, vi vil gerne have så mange som muligt til at få adgang, men der er ingen grund til at få alle til det på en gang.
- Hvordan finder vi så en algoritme når ingen af processerne kan kommunikere med hinanden?

Design af algoritmen i

- For en given proces P_i og en given runde t, lad $A[i,t]$ være hændelsen at P_i forsøger at få adgang til databasen i runde t.
- Vi ved at hver proces forsøger at få edgang i hver runde med sandsynlighed p, så sandsynligheden $P(A[i,t]) = p$.
- Ligeledes er sandsynligheden for at dette ikke sker $\overline{P(A[i,t])} = 1 - p$
- Ydermere, lad $S[i,t]$ være hændelsen at proces P_i **får** adgang.
- $S[i,t]$ sker kun hvis det udelukkende er proces i der forsøger at få adgang.

Design af algoritmen ii

- Dermed:

$$S[i,t] = A[i,t] \cap \left(\bigcup_{j \neq i} \overline{A[j,t]} \right)$$

- Alle hændelserne i snittet er uafhængige. Dermed, kan vi gange sandsynlighedern sammen for at få sandsynligheden således:

$$P(S[i,t]) = P(A[i,t]) \cdot \prod_{j \neq i} P(\overline{A[j,t]}) = p(1-p)^{n-1}$$

- Nu har vi sandsynligheden, men hvad skal sandsynligheden for at p forsøger adgang så være?

Design af algoritmen iii

- Funktionen $f(p) = p(1-p)^{n-1}$ er positiv for værdierne af p således at $0 < p < 1$, og dets derivative $f'(p) = (1-p)^{n-1} - (n-1)p(1-p)^{n-2}$ har en enkelt nul-værdi ved værdien $p = 1/n$, hvor dets maximum findes.
- Når vi sætter $p = 1/n$ får vi $P(S[i,t]) = \frac{1}{n}\left(1 - \frac{1}{n}\right)^{n-1}$.

Theorem (13.1)

(a) The function $\left(1 - \frac{1}{n}\right)^n$ *converges monotonically from* $\frac{1}{4}$ *up to* $\frac{1}{e}$ *as n increases from 2*
(b) The function $\left(1 - \frac{1}{n}\right)^{n-1}$ *converges monotonically from* $\frac{1}{2}$ *down to* $\frac{1}{e}$ *as n increases from 2.*

Design af algoritmen iv

- Hvis vi bruger dette teorem, kan vi se at $\frac{1}{en} \leq Pr[S[i,t]] \leq \frac{1}{2n}$, og dermed er $P(S[i,t])$ asymptotisk lig med $\Theta(1/n)$.

Venter på en specifik proces i

- Vi vil gerne finde ud af hvor lang tid det tager for en specifik proces p før den får adgang.
- Det er tydeligt at for én runde, er chancen $1/p$, men hvad men flere runder?
- Lad $F[i,t]$ være hændelsen at P_i ikke får success i runderne $1..t$.
- Dette er tydeligt $\overline{S[i,r]}, r = 1, 2, \ldots, t$.

Venter på en specifik proces ii

- Ydermere, siden hver af hændelserne er uafhængige, kan vi udregne sandsynligheden af $F[i,t]$ gennem gange:

$$\mathrm{P(F[i,t])} = \mathrm{P}(\bigcup_{r=1}^{t} \overline{S[i,r]}) = \prod_{r=1}^{t} P(\overline{S[i,r]}) = \left[1 - \tfrac{1}{n}\left(1 - \tfrac{1}{n}\right)\right]^{t}$$

- Givet den asymptotiske teorem fra tidligere kan vi i stedet få:

$$\prod_{r=1}^{t} p(\overline{S[i,r]}) \leq (1 - \frac{1}{en})^{t}$$

Venter på en specifik proces iii

- Endvidere, hvis $t = en$, så

$$P(F[i,t]) \leq \left(1 - \frac{1}{en}\right)^{\lceil en \rceil} \leq \left(1 - \frac{1}{en}\right)^{en} \leq \frac{1}{e}$$

- Dette er upper bounded af e^{-1} hvilket vil sige at hvis $t = \lceil en \rceil \cdot (c \ln n)$ har vi at

$$P[F[i,t]] \leq \left(1 - \frac{1}{en}\right)^{t} = \left(\left(1 - \frac{1}{en}\right)^{\lceil en \rceil}\right)^{c \ln n} \leq e^{-c \ln n} = n^{-c}$$

Complexity of the Edmonds-Karp Algorithm i

- Recall: Edmonds-Karp algoritmen finder et max-flow ved at augmentere det nuværende flow langs den korteste (s, t)-path i residual netværk $G - f$.
- Vi kan bevise køretiden $O(|V||E|^2)$ som følger:
  1. VI kan finde det næste augmenting path eller bestemme at der ikke er noget i $O(|V| + |E|) = O(|E|)$ tid, da va antager at G er *connected*. Af Max-Flow Min-Cut toremet, hvis der ikke er noget (s, t)-path i G_f så er f max flow. Det Følger at vi skal vise at antallet af augmenting paths brugt i algoritmen er $O(|V||E|)$.
  2. Lad $P_1, P_2, \ldots, P_r$ være sekvensen af augmenterende veje som algoritmen finder før den terminerer. Lad også $f_0 \equiv 0$ og lad $f_1, f_2, \ldots, f_r$ være flows efter hver augmentering. Dermed er f_{i+1} fundet fra f_i augmentered af $\delta(P_i)$ units igennem P_i (path i), hvor $\delta(P_i)$ er minimums residual capacity af en kant på P_i.

Complexity of the Edmonds-Karp Algorithm ii

- **Påstand 1:** For alle $i \in \{1, 2, \ldots, r-1\}$ har vi $|E(P_i)| \leq |E(P_{i+1})|$.
- For at bevise dette bruger vi at augmenting path P_i findes ved brug af BFS i $G_{f_{i-1}}$, for $i = 1, 2, \ldots, r$.
- Antag at distancen fra s til t i $G_{f_{i-1}}$ er k.
- Så definerer BFS fra s distanceklasser $L_0, L_1, \ldots, L_k$ fra s hvor $L_o = \{s\}$ og $t \in L_k$.
- I.e., hver klasse har noderne fundet efter distance L_f hvor f er nuværende distance.
- Vi kalder en kant fra L_a til L_v **forward, flat** eller **backwards**, hvis, respektivt, $a = b - a, a = b,$ eller $a > b$.
- Da P_i er den korteste vej, så er hver kant (u, v) på P_i fremadgående.

Complexity of the Edmonds-Karp Algorithm iii

- Ydermere, hver (s, t)-vej af længde k i $G_{f_{i-1}}$ bruger kun forward kanter.
- Læg nu mærke til hvilke nye kanter G_{f_i} kan indeholde.
- De eneste nye kante der kan fremkomme når du går fra $G_{f_{i-1}}$ til G_{f_i} er de kanter der er omvendt af dem på P_i, og jeg forstår ikke resten:(
- **Påstand 2:** der er højest $|E|$ veje mellem $P_1, P_2, \ldots, P_r$ som har samme længde.
- Dette følger fra faktum at hver gang vi augmenterer langs den korteste vej P_i er der mindst en kant (u, v) som er fremadgående i forhold til den nuvreædne distanceklasse.
- Jeg forstår heller ikke helt det her

The Integrality Theorem for Flows i

Theorem (26.10 (Integrality Theorem))

If the capacity function c takes on only integral values, then the maximum flow f produced by the Ford-Fulkerson method has the property that $|f|$ is an integer. Moreover, for all vertices u and v, the value of $f(u, v)$ is an integer.

- Så, hvis vi er givet et flow netværk $G = (V, E)$, og en kapacitetsfunktion $c : E \to Z_0$, og to distinkte knuder $s, t \in V$, så er der et maksimumsflow f^* således at $f^*(u, v)$ er et ikke-.negativt heltal for hver kant $(u, v) \in E$.
- Det er simpelt, men brugbart.

The Integrality Theorem for Flows ii

- En graf $G = (V, E)$ er $d-$regulær hvis, for hver knude $v < inV$, har præcis d kanter. En matching er **perfekt** hvis det er incident til alle knuder på grafen.

Theorem

Every $d-$regular bipartite graph $G = (X, Y, E)$ has a perfect matching.

- Bevis:
- X, Y er de to sæt er knuder. VI kan tælle antallet af kanter i E ved at summere graderne af knuderne i X eller i Y, så $|X| = |Y|$ holder da $|E| = d|X| = d|Y|$
- Jeg giver op.

Misc 3

Monte Carlo Algoritmer i

- Lad $S = s_1, s_2, \ldots, s_n$ *være en kollektion af n heltal. Et element* s_i af S er et **majority element** af S hvis $|\{j : s_j = s_i\}| > n/2$.

  Læg mærke til at hvis S har et majority element, har det kun ét majority element (da det skal forekomme som mere end halvdelen af alle elementer).

  Det er nemt at finde et majority element. Sortér elementerne og skan det sorterede sæt til at se om en af værdierne forekommer mere end $n/2$ gange. Det vil tage $O(n \log n)$ tid.

  Vi vil dog nu vise at det kan gøres på $O(n)$ tid

Monte Carlo Algoritmer ii

Lad A være algoritmen der tager input S, m, hvor S er sættet og m er en ikke-negativt konstant. Algoritmen kører som følger:

1. Bliv ved m gange:
   (a) Vælg et tilfældligt element $s \in S$.
   (b) Tjek om s er et majority element. Hvis så, returner `true` og stop.
2. returner `false`

Vi vil nu analysere køretiden.

Hvis A returnerer `true`, så har S et majority element.

Nemlig elementet s som fik algoritmen til at returnere `true`.

Monte Carlo Algoritmer iii

Men, hvis A returnerer `false`, kan der stadig være et majority element vi bare ikke har fundet endnu.

Sandsynligheden for at værdien af det valgte element s ikke er lig med vørdien af majority elementet er mindre end $1/2$, og siden vi kun laver uafhængige valg i hver af de m runder, er sandsynligheden at de alle er forskellige fra majority elementet højest $(1/2)^m$. Hvis $m = 20$ er sandsynligheden for dette mindre end $\frac{1}{1000000}$

Majority Element og Heavy Hitters

- x er et **majority** af $S = \{x_1, x_2, \ldots, x_n\}$ hvis $|\{x_i | x_i = x\}| > \frac{n}{2}$
- Vi har åbenbart set en randomiseret algoritme for dette med $O(n)$ expected runng time.
- Antag nu at vi har en lang datastrøm $\langle x_1, x_2, \ldots, x_m \rangle$, som kun må læses en enkelt gang.
- Følgende er en simpel algoritme til at finde majority element:

```
Majority(S) {
c = 0, l = ∅
for i = 1 to m
    if (x_i = l) then c = c + 1
    else c = c - 1
    if c <= 0 then
        c = 1, l = a_i
return l
}
```

Majority Element i

- Forvirrende algoritme, hva?
- Vi påstår at denne algoritme outputter majority elementet hvis der er en.