Algorithmen 2

Kapitel: Randomisierte Algorithmen

Thomas Worsch

Fakultät für Informatik
Karlsruher Institut für Technologie

Wintersemester 2017/2018

Überblick

Sichtweisen für randomisierte Algorithmen

Erweiterung des Begriffs eines deterministischen Algorithmus:

- neuer *algorithmischer Elementarschritt*
  Beschaffung eines zufälligen Wertes
  RANDINT($c : \mathbb{N}$) liefere
  zufällig gleichverteilt eine ganze Zahl $x \in \{0, 1, \ldots, c-1\}$

alternative Sichtweisen (nicht in dieser Vorlesung)

- „quantifizierter Nichtdeterminismus“
  mit (schönen!) Wahrscheinlichkeiten
- zufällige Auswahl eines deterministischen Algorithmus

Fundamentale Änderung

Mehrere Ausführungen

des *gleichen* randomisierten Algorithmus
für die *gleiche* Eingabe

können verschieden sein!

Beispiel: Randomisierter Quicksort

RANDQS(s) mit $s = \langle e_0, \ldots, e_{n-1} \rangle$

1 **if** $n \leq 1$
2 **then** return s
3 $i \leftarrow$ RANDINT(n); $pvt \leftarrow e_i$
4 $a \leftarrow \langle e \in s : e < pvt \rangle$
5 $b \leftarrow \langle e \in s : e = pvt \rangle$
6 $c \leftarrow \langle e \in s : e > pvt \rangle$
7 **return** CONCAT(RANDQS(a), b, RANDQS(c))

Zufallsvariablen überall

Selbst für eine einzelne festgehaltene Eingabe sind

- die benötigte Laufzeit
- der benötigte Speicherplatz
- das berechnete Ergebnis

Zufallsvariablen.

Erinnerung an W-Theorie

- Ω Menge der *Elementarereignisse*
- $E \subseteq 2^\Omega$ Menge von *Ereignissen* (σ-Algebra)
  - *diskrete* σ-Algebra: $E = 2^\Omega$ und $|\Omega| \leq |\mathbb{N}|$
- *W-Maß* $\mathbf{Pr} : E \to$ (σ-additiv, $\mathbf{Pr}[\Omega] = 1$)
- *Zufallsvariable* $X : \Omega \to \mathbb{R}$ mit Eigenschaften …
  (erfüllt, falls σ-Algebra diskret)
- Schreibweisen $\mathbf{Pr}[X \leq x]$ statt $\mathbf{Pr}[\{\omega \mid X(\omega) \leq x\}]$
  $\mathbf{Pr}[X = x]$ statt $\mathbf{Pr}[\{\omega \mid X(\omega) = x\}]$

Standardbeispiel: Würfeln

- $\Omega = \{\boxed{1}, \boxed{2}, \boxed{3}, \boxed{4}, \boxed{5}, \boxed{6}\}$
- $E = 2^{\Omega}$ diskret
  - z. B. $g = \{\boxed{2}, \boxed{4}, \boxed{6}\}$ «gerade Zahl gewürfelt»
- «fairer» Würfel:
  $\forall \omega \in \Omega : \mathbf{Pr}[\omega] = 1/6$
  - $\mathbf{Pr}[g] = 1/2$
  - allgemein: $\mathbf{Pr}[e] = |e|/6$
- $X(\omega) = \begin{cases} 0, & \text{falls } \omega \text{ Produkt zweier Primfaktoren} \\ 1, & \text{sonst} \end{cases}$
  - z. B. $\mathbf{Pr}[X = 1] = 2/3$

Erinnerung an W-Theorie (2)

- Zufallsvariable $X : \Omega \to \mathbb{R}$, nur abzählbar viele $X(\omega) \neq 0$
- *Erwartungswert* von X: $\mathbf{E}[X] = \sum_{x \in \mathbb{R}} x \cdot \mathbf{Pr}[X = x]$

  (falls das absolut konvergiert)
- immer: $\mathbf{E}[X + Y] = \mathbf{E}[X] + \mathbf{E}[Y]$

- *unabhängige* ZV: für alle $x, y \in \mathbb{R}$:
  $\mathbf{Pr}[X = x \wedge Y = y] = \mathbf{Pr}[X = x] \cdot \mathbf{Pr}[Y = y]$
- falls X, Y unabhängig: $\mathbf{E}[X \cdot Y] = \mathbf{E}[X] \cdot \mathbf{E}[Y]$

- *Indikator-ZV:* 0 und 1 einzig mögliche Funktionswerte

Standardbeispiel: Würfeln

- $\Omega = \{\boxed{1}, \boxed{2}, \boxed{3}, \boxed{4}, \boxed{5}, \boxed{6}\}$
- $E = 2^{\Omega}$
- Indikator-ZV $X(\omega) = 1$, falls ω prim
- Indikator-ZV $Y(\omega) = 1$, falls ω gerade
- X und Y *nicht* unabhängig:

$$\mathbf{Pr}[X = 1] = 1/2$$
$$\mathbf{Pr}[Y = 1] = 1/2$$
$$\mathbf{Pr}[X = 1] \cdot \mathbf{Pr}[Y = 1] = 1/4$$
$$\mathbf{Pr}[X = 1 \wedge Y = 1] = 1/6$$

Beispiel: Ausführung randomisierter Algorithmen

zu randomisiertem Algorithmus R und Eingabe x

- $\Omega = \{$«Programmlauf» von R für Eingabe $x\}$
  - Programmlauf: Folge der durchlaufenen globalen Speicherzustände
- $E = 2^{\Omega}$
- Beispiele möglicher Zufallsvariablen:
  - X(Prog.lauf) = Anzahl Schritte
  - X(Prog.lauf) = Ausgabe

Algorithmen mit unbekannter Laufzeit

- Will man das?
- «Wie unbekannt?»
  - *Quantifizierung?*

Algorithmen, die «variierende Ausgaben» liefern

- vielleicht erträglich bei Optimierungsproblemen
  - *Quantifizierung?*
- Erzeugung «zufälliger Objekte»
  - *Eigenschaften?*

Algorithmen, die «falsche Ausgaben» liefern

- Will man das?
- Schlägt in diesem Hörsaal gleich ein Meteorit ein?
- Ist $2^{400} - 593$ die größte Primzahl kleiner als 2^{400}?
- Soll die Ampel jetzt auf grün schalten?

- *Fehlerwahrscheinlichkeit?*

Vorteile randomisierter Algorithmen

Sie sind manchmal

- leichter zu formulieren und zu implementieren
- «schneller»
- «besser»
- die einzige Möglichkeit

Vorteile randomisierter Algorithmen

Sie sind manchmal

- leichter zu formulieren und zu implementieren
- «schneller»
- «besser»
- die einzige Möglichkeit

Aber: Man zahlt einen Preis!

Überblick

Motivation

auch wenn sie unrealistisch ist …

- gegeben: Listen $\langle e_1, \ldots, e_n \rangle$ und $\langle a_1, \ldots, a_n \rangle$
  - die Werte seien Elemente eines Körpers $\mathbb{F}$
  - z. B. Ein- und Ausgabe eines (Sortier?)Algorithmus
- Frage: Beinhalten die Listen die gleichen Elemente?
- naiver Algorithmus: quadratische Laufzeit

Polynome

- definiere $e(z) = \prod_{i=1}^{n}(z - e_i)$ und $a(z) = \prod_{i=1}^{n}(z - a_i)$
  - Ist $e(z) - a(z)$ das Nullpolynom?

Polynome

- definiere $e(z) = \prod_{i=1}^{n}(z - e_i)$ und $a(z) = \prod_{i=1}^{n}(z - a_i)$
  - Ist $e(z) - a(z)$ das Nullpolynom?
- Algorithmus

$c \leftarrow$ zufällig gleichverteilt $\in \mathbb{F}$
$y \leftarrow e(c) - a(c)$
if $y = 0$
  then return «$e = a$»
  else return «$e \neq a$»

Polynome

- definiere $e(z) = \prod_{i=1}^{n}(z - e_i)$ und $a(z) = \prod_{i=1}^{n}(z - a_i)$
  - Ist $e(z) - a(z)$ das Nullpolynom?
- Algorithmus

  $c \leftarrow$ zufällig gleichverteilt $\in \mathbb{F}$
  $y \leftarrow e(c) - a(c)$
  if $y = 0$
  then return «$e = a$»
  else return «$e \neq a$»

- Fehlerwahrscheinlichkeit
  $\mathbf{Pr}\left[e(c) - a(c) = 0 \wedge e(z) - a(z) \text{ nicht Nullpolynom}\right] \leq \frac{n}{|\mathbb{F}|}$

Ausblick: polynomial identity testing

Verallgemeinerung auf Polynome mit mehreren Veränderlichen

- gegeben: $P(z_1, \ldots, z_n)$ und $Q(z_1, \ldots, z_n)$
  Frage: Ist $P = Q$?
- dafür gibt es Anwendungen
- Idee von eben anpassbar
- kein deterministischer Polynomialzeit-Algorithmus bekannt!
  - man multipliziere $\prod_{i=1}^{n-1}(z_i + z_{i+1})$ aus … oder nicht …

Überblick

Randomisierter Quicksort

RANDQS(s) mit $s = \langle e_0, \ldots, e_{n-1} \rangle$, e_i paarweise verschieden

1 **if** $n \leq 1$
2 **then** return s
3 $i \leftarrow$ RANDINT(n); $pvt \leftarrow e_i$
4 $a \leftarrow \langle e \in s : e < pvt \rangle$
5 $b \leftarrow \langle e \in s : e = pvt \rangle$
6 $c \leftarrow \langle e \in s : e > pvt \rangle$
7 **return** CONCAT(RANDQS(a), b, RANDQS(c))

randQS: Anzahl Vergleiche

Erwartungswert (aus Algorithmen 1)

- es sei $\textsc{randQS}(e_1, \ldots, e_n) = (a_1, \ldots, a_n)$
- Laufzeit-Analyse nutzt

$$X_{ij} = \begin{cases} 1, & \text{falls } a_i \text{ und } a_j \text{ miteinander verglichen} \\ 0, & \text{sonst} \end{cases}$$

- Anzahl Vergleiche $= \mathbf{E}\left[\sum_{i<j} X_{ij}\right]$:

$$\mathbf{E}\left[\sum_{i<j} X_{ij}\right] = \sum_{i<j} \mathbf{E}\left[X_{ij}\right]$$

$$= \sum_{i<j} \mathbf{Pr}\left[X_{ij} = 1\right] = \sum_{i<j} \frac{2}{j-i+1} \leq 2n \ln n$$

randQS: Anzahl Vergleiche (2)

mit hoher Wahrscheinlichkeit wie klein?

- Anzahl Vergleiche immer zwischen $n \ln n$ und n^2
- erwartete Laufzeit in $\Theta(n \ln n)$.
- zufrieden?

randQS: Anzahl Vergleiche (2)

mit hoher Wahrscheinlichkeit wie klein?

- Anzahl Vergleiche immer zwischen $n \ln n$ und n^2
- erwartete Laufzeit in $\Theta(n \ln n)$.
- zufrieden?
- Wahrscheinlichkeit für Laufzeit «nahe am Erwartungswert» z. B. $\leq 28n \ln n$?

randQS: Anzahl Vergleiche mit hoher Wkt.

- e_i beliebiges Element
- L_j Liste auf Rekursionsstufe j, die e_i enthält
- Zufallsvariable Y_j

$$Y_j = \begin{cases} 1, & \text{falls } \frac{1}{4}|L_j| \leq |L_{j+1}| \leq \frac{3}{4}|L_j| \\ 0, & \text{sonst} \end{cases}$$

- $\mathbf{E}\left[Y_j\right] = \mathbf{Pr}\left[Y_j = 1\right] = \frac{1}{2}$
- die Y_j sind unabhängig voneinander

randQS: Anzahl Vergleiche mit hoher Wkt.

- $Y_j = 1 \quad$ falls $\frac{1}{4}|L_j| \leq |L_{j+1}| \leq \frac{3}{4}|L_j|$
- nenne Rekursionsstufe *erfolgreich*, falls $Y_j = 1$
- wenn r von k Stufen erfolgreich, dann $|L_k| \leq \left(\frac{3}{4}\right)^r n$
- setze $r = 3.5 \ln n$, denn

$$r = 3.5 \ln n \implies \ln n \leq r \ln \frac{4}{3} \implies n \leq \left(\frac{4}{3}\right)^r$$

$$\implies \left(\frac{3}{4}\right)^r n \leq 1 \implies |L_k| \leq 1$$

- nach r erfolgreichen Stufen ist e_i in einelementiger Liste

randQS: Anzahl Vergleiche mit hoher Wkt.

- sei nun $k = 8r = 28 \ln n$
- $Y = \sum_{j=1}^{k} Y_j$, $\mathbf{E}[Y] = k/2$
- e_i höchstens dann *nicht* in einelementiger Liste, falls $Y < r$
- Wie groß ist
  $\mathbf{Pr}[Y < r] = \mathbf{Pr}\left[Y < \frac{k}{8}\right] = \mathbf{Pr}\left[Y < (1 - \frac{3}{4})\mathbf{E}[Y]\right]$?
- gleich *Chernoff-Schranken*
  liefern z. B. $\mathbf{Pr}[Y < r] \leq e^{-\frac{k}{4}\cdot(\frac{3}{4})^2} = n^{-63/16}$
- nach k Rekursionsstufen
  - Wkt. ein bestimmtes e_i nicht in Einerliste: $\leq n^{-63/16}$
  - Wkt. mindestens ein e_i nicht in Einerliste:
    $\leq n \cdot n^{-63/16} = n^{-47/16}$

randQS: Anzahl Vergleiche mit hoher Wkt.

- $k = 8r = 28 \ln n$
- nach k Rekursionsstufen
  - Wkt. ein bestimmtes e_i nicht in Einerliste: $\leq n^{-63/16}$
  - Wkt. mindestens ein e_i nicht in Einerliste: $\leq n \cdot n^{-63/16} = n^{-47/16}$
- mit Wkt. $1 - n^{-47/16}$ nach $k = 28 \ln n$ Rekursionsstufen *alle* Elemente in Einerlisten
- mit Wkt. $1 - n^{-47/16}$ reichen $28n \ln n$ Vergleiche

Überblick

Einfache Schranken

Markov- und Chebyshev-Ungleichung

- *Markov-Ungleichung:* ZV $Y \geq 0$, Erwartungswert μ_Y.
  Dann gilt für alle $t, k \in \mathbb{R}_+$:

$$\mathbf{Pr}\left[Y \geq t\right] \leq \frac{\mu_Y}{t} \quad \text{bzw.} \quad \mathbf{Pr}\left[Y \geq k\mu_Y\right] \leq \frac{1}{k}$$

- *Chebyshev-Ungleichung:* ZV X mit
  Erwartungswert μ_X, Standardabweichung σ_X.
  Dann gilt für alle $t \in \mathbb{R}_+$:

$$\mathbf{Pr}\left[|X - \mu_X| \geq t\sigma_X\right] \leq \frac{1}{t^2} \quad \text{bzw.} \quad \mathbf{Pr}\left[|X - \mu_X| \geq t\right] \leq \frac{\sigma_X^2}{t^2}$$

Chernoff-Schranken

Es seien

- $X_1, \ldots, X_n$ *unabhängige 0-1-ZV* mit $\mathbf{Pr}[X_i = 1] = p_i$
- $X = \sum_{i=1}^{n} X_i$ und $\mu = \mathbf{E}[X] = \sum p_i$

Dann gilt

- für alle $0 \leq \delta$:

$$\mathbf{Pr}[X \geq (1+\delta)\mu] \leq \left(\frac{e^{\delta}}{(1+\delta)^{(1+\delta)}} \right)^{\mu}$$

- für alle $1 > \delta \geq 0$ also $0 < 1 - \delta \leq 1$:

$$\mathbf{Pr}[X \leq (1-\delta)\mu] \leq \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}} \right)^{\mu}$$

Chernoff-Schranken: Beweis von Teil 1

- sei t positiv; Markov-Ungleichung liefert:

$$\mathbf{Pr}\left[X \geq (1+\delta)\mu\right] = \mathbf{Pr}\left[e^{tX} \geq e^{t(1+\delta)\mu}\right] \leq \frac{\mathbf{E}\left[e^{tX}\right]}{e^{t(1+\delta)\mu}}$$

- mit den X_i sind auch die e^{tX_i} unabhängig:

$$\mathbf{E}\left[e^{tX}\right] = \mathbf{E}\left[e^{t\sum X_i}\right] = \mathbf{E}\left[\prod e^{tX_i}\right] = \prod \mathbf{E}\left[e^{tX_i}\right]$$

- $\mathbf{E}\left[e^{tX_i}\right] = p_i \cdot e^t + (1-p_i)\cdot 1 = 1 + p_i(e^t - 1) \leq e^{p_i(e^t-1)}$
- $\mathbf{Pr}\left[X \geq (1+\delta)\mu\right] \leq \dfrac{\prod e^{p_i(e^t-1)}}{e^{t(1+\delta)\mu}} = \dfrac{e^{\sum p_i(e^t-1)}}{e^{t(1+\delta)\mu}}$

$$= \frac{e^{\mu(e^t-1)}}{e^{t(1+\delta)\mu}} = \left(\frac{e^{(e^t-1)}}{e^{t(1+\delta)}}\right)^{\mu}$$

- wähle $t = \ln(1+\delta)$ (positiv!)

Chernoff-Schranken: Vereinfachungen

- statt $\left(\frac{e^{\delta}}{(1\pm\delta)^{(1\pm\delta)}}\right)^{\mu}$ betrachte $f(\delta) = \delta - (1 \pm \delta)\ln(1 \pm \delta)$
- je nach Bereich, aus dem δ stammt,
  kann man $f(\delta)$ nach oben abschätzen in der Form $-\delta^2/c$

Chernoff-Schranken: Korollare

obige Abschätzungen manchmal etwas unhandlich
diverse Vereinfachungen, unter anderem:

- Für $1 > \delta \geq 0$ gilt:

$$\mathbf{Pr}\left[X \leq (1-\delta)\mu\right] \leq \left(\frac{e^{-\delta}}{(1-\delta)^{(1-\delta)}}\right)^{\mu} \leq e^{-\delta^2\mu/2}$$

Überblick

Und-Oder-Bäume

T_k vollständiger binärer Baum der Höhe $2k$

- innere Knoten abwechselnd mit "$\wedge$" und "$\vee$" markiert
- Wurzel von T_1 ist $\wedge$-Knoten mit zwei $\vee$-Knoten als Nachfolger
- Jeder dieser Knoten hat zwei Blätter als Nachfolger.
- T_k entsteht aus T_1, indem die Blätter durch Kopien von T_{k-1} ersetzt werden

- T_k hat $n = 4^k$ Blätter
- im folgenden $x_1, \ldots, x_{4^k}$ genannt

Auswertung von Und-Oder-Bäumen

- boolesche Werte an allen Blättern
  $\rightsquigarrow$ boolscher Wert für Wurzel (und andere innere Knoten)
  - (für die naheliegende Definition von Auswertung …)
- Problem:
  - gegeben: Werte $x_1, \ldots, x_n$ an den Blättern
  - gesucht: Wert $T_k(x_1 \ldots, x_n)$ der Wurzel
- Berechnung des Wurzelwertes „bottom up“ durch
  - Besuch aller $n = 4^k$ Blätter und
  - Berechnung der Werte aller inneren Knoten möglich
- **Frage:** Geht es besser?

Satz

Für jeden *deterministischen* Alg. A und jedes $k \geq 1$ gilt:

- Es gibt eine Folge $x_1, \ldots, x_n$ von Bits so, dass A bei Berechnung von $T_k(x_1, \ldots, x_n)$ *alle* $n = 4^k$ Blätter besucht.

Satz

Für jeden *deterministischen* Alg. A und jedes $k \geq 1$ gilt:

- Es gibt eine Folge $x_1, \ldots, x_n$ von Bits so, dass A bei Berechnung von $T_k(x_1, \ldots, x_n)$ *alle* $n = 4^k$ Blätter besucht.
- Wert der Wurzel = Wert des zuletzt besuchten Blattes
- sowohl Wurzelwert $= 0$ als auch Wurzelwert $= 1$ kann erzwungen werden

Beweis

Induktion über k

- Induktionsanfang $k = 1$:
  - A muss mindestens ein Blatt besuchen, o. B. d. A. x_1.
  - setze $x_1 = 0 \quad \leadsto$ kein innerer Wert festgelegt
  - A muss ein weiteres Blatt besuchen.
  - O. B. d. A. zwei Möglichkeiten:
    1. Zweites besuchtes Blatt ist x_2. Setze $x_2 = 1$.
       Damit nur Wert des $\vee$-Knotens klar, Wurzelwert nicht.
       A muss weiteres Blatt besuchen, o. B. d. A. x_3.
       Setze $x_3 = 0$. A muss x_4 besuchen $\leadsto$ Wurzelwert.
    2. Zweites besuchtes Blatt ist x_3. Setze $x_3 = 0$.
       Kein innerer Wert festgelegt
       A muss weiteres Blatt besuchen, o. B. d. A. x_2.
       Setze $x_2 = 1$. A muss x_4 besuchen $\leadsto$ Wurzelwert

Beweis (2)

- Induktionsschritt $k-1 \rightsquigarrow k$:
  - Fasse T_k als T_1-Baum auf, dessen Blätter durch T_{k-1}-Bäume ersetzt sind.
  - Bezeichne die „Blätter“ von T_1 mit $y_1, \ldots, y_4$.
  - Analog Induktionsanfang kann durch geeignete Wahl der y_i erzwungen werden, dass A *alle 4* Werte ermitteln muss.
  - Induktionsvoraussetzung: für jeden T_{k-1}-Baum gibt es Blattwerte, die gewünschtes y_i liefern und erzwingen, dass A alle darunter liegenden Blätter besuchen muss.
  - Also muss A in diesem Fall alle Blätter überhaupt besuchen.

Zwischenüberlegung

- Jede Liste $x_1, \ldots, x_{4^k}$ der Länge $4^k = n$
  enthält eine Teilfolge $y_1, \ldots, y_{2^k}$ der Länge $2^k = \sqrt{n}$,
  die schon den Wurzelwert festlegt!
  - Beweis: Induktion …
- aber deterministisch manchmal *alle* Knoten benötigt
- «Die y_i sind schwer zu finden.»

- Ausweg Randomisierung … ? …

Algorithmus: randomisierte UOB-Auswertung

Algorithmus: randomisierte UOB-Auswertung

```
proc AndNodeEval(T)
if IsLeaf(T) then
        return value(T) fi
⟨andernfalls:⟩
T' ← ⟨zufälliger T-U.baum⟩
r ← OrNodeEval(T')
if r = 0 then
        return 0
else
        T'' ← ⟨and. T-U.baum⟩
        return OrNodeEval(T'')
fi
```

```
proc OrNodeEval(T)


T' ← ⟨zufälliger T-U.baum⟩
r ← AndNodeEval(T')
if r = 1 then
        return 1
else
        T'' ← ⟨and. T-U.baum⟩
        return AndNodeEval(T'')
fi
```

```
AndNodeEval(root)
```

Satz

Für die randomisierte Auswertung von UOB gilt:
Für *jede Folge* $x_1, \ldots, x_{4^k}$
ist Erwartungswert für die Anzahl besuchter Blätter
$\leq 3^k = n^{\log_4 3} \approx n^{0.792\ldots}$.

(Das ist übrigens beweisbar optimal.)

Beweis

Induktion

- E: ein Erwartungswert für die Anzahl besuchter Blätter
- Induktionsanfang $k = 1$: Überprüfen aller 16 möglichen Kombinationen $x_1, \ldots, x_4$. Z. B. 0100 (Rest analog):
  1. Falls erst linker Teilbaum: gleich wahrscheinlich wird
     erst und nur die 1 oder
     erst die 0 und dann die 1 besucht,
     anschließend im rechten Teilbaum beide Blätter.
     $E = 1/2 \cdot 1 + 1/2 \cdot 2 + 2 = 7/2$.
  2. Falls erst rechter Teilbaum; nach Besuch beider Blätter klar: der T_1-Baum den liefert Wert 0.
     $E = 2$.

  Beide Fälle gleich wahrscheinlich, also insgesamt
  $E = 1/2 \cdot 7/2 + 1/2 \cdot 2 = 11/4 < 3 \ (= 3^1 = 3^k)$.

Beweis (2)

Induktionsschritt $k-1 \rightsquigarrow k$:

- Betrachte zunächst $\vee$-Knoten mit zwei T_{k-1}-Bäumen darunter. Zwei Fälle:

  O1. $\vee$-Knoten wird 1 liefern:

  - Dann muss mindestens ein T_{k-1}-Baum dies auch tun.
  - Mit Wahrscheinlichkeit $p \geq 1/2$ wird ein Unterbaum untersucht, der 1 liefert.
  - Mit Wahrscheinlichkeit $1-p \leq 1/2$ werden beide Unterbäume untersucht.
  - $E \leq p \cdot 3^{k-1} + (1-p) \cdot 2 \cdot 3^{k-1} = (2-p) \cdot 3^{k-1} \leq 3/2 \cdot 3^{k-1}$.

  O2. Der $\vee$-Knoten wird 0 liefern:

  - Dann müssen beide T_{k-1}-Bäume dies auch tun.
  - Nach Induktionsvoraussetzung $E \leq 2 \cdot 3^{k-1}$ ist.

Beweis (3)

Induktionsschritt $k-1 \leadsto k$: Teil 2a

- Betrachte Wurzel des T_k-Baumes mit zwei eben untersuchten Bäumen darunter. Zwei Fälle:
  U1. Der $\wedge$-Knoten wird 0 liefern:
  - Dann muss mindestens ein U.baum dies auch tun.
  - Mit Wahrscheinlichkeit $p \geq 1/2$ wird ein Unterbaum untersucht, der 0 liefert.
  - Mit Wahrscheinlichkeit $1-p \leq 1/2$ werden beide Unterbäume untersucht.
  - Gemäß O1 und O2 ist folglich

$$\begin{aligned} E &\leq p \cdot 2 \cdot 3^{k-1} + (1-p) \cdot (3/2 \cdot 3^{k-1} + 2 \cdot 3^{k-1}) \\ &= 7/2 \cdot 3^{k-1} - p \cdot 3/2 \cdot 3^{k-1} \\ &\leq 11/4 \cdot 3^{k-1} \\ &\leq 3^k \end{aligned}$$

Beweis (4)

Induktionsschritt $k-1 \rightsquigarrow k$: Teil 2b

- Betrachte Wurzel des T_k-Baumes mit zwei eben untersuchten Bäumen darunter. Fall:
  - U2. Der $\wedge$-Knoten wird 1 liefern:
    - Dann müssen beide Unterbäume dies auch tun.
    - Gemäß O1 ist daher $E \leq 2 \cdot 3/2 \cdot 3^{k-1} \leq 3^k$.

Überblick

Erdős-Rényi-Zufallsgraphen

$G(n, p)$

1 $V \leftarrow \{1, \ldots, n\}$
2 $E \leftarrow \{\}$
3 **for** $i \leftarrow 1$ **to** $n - 1$ **do**
4 **for** $j \leftarrow i + 1$ **to** n **do**
5 with probability p: add edge $\{i, j\}$ to E
6 **return** (V, E)

mitunter: p von n abhängig, z. B. $p = p(n) = \frac{\ln n}{n}$

ER-Graphen: einfache Beobachtungen

- Wkt. für bestimmten Graphen mit n Knoten und m Kanten: $p^m(1-p)^{\binom{n}{2}-m}$
  - $p = 1/2$, n fest: alle Graphen gleichwahrscheinlich
- erwartete Anzahl Kanten: $p\binom{n}{2}$
- erwarteter Knotengrad: $p(n-1)$
- Wkt. g_d für Knotengrad d
  - $\binom{n-1}{d}$ mögliche «Nachbarmengen»
  - jede passiert mit Wkt. $p^d(1-p)^{n-1-d}$
  - insgesamt $g_d = \binom{n-1}{d}p^d(1-p)^{n-1-d}$
    Binomialverteilung

Manchmal interessieren sehr große n — und asymptotische Eigenschaften

- mitunter interessant: $n \to \infty$
- *monotone increasing property* $\mathcal{P}$
  - Hinzufügen von Kanten erhält die Eigenschaft
  - nichttrivial (leerer Graph: nein, vollständiger Graph: ja)
- $p^*(n)$ ist *Grenzfunktion*, falls
  $\lim_{n\to\infty} \mathbf{Pr}\left[G(n,p) \text{ hat } \mathcal{P}\right]$
  $$= \begin{cases} 0, & \text{falls } \lim_{n\to\infty} p(n)/p^*(n) = 0 \\ 1, & \text{falls } \lim_{n\to\infty} p(n)/p^*(n) = \infty \end{cases}$$
- *«Phasenübergang»*
  - passiert für jede monotone increasing property

Manchmal interessieren sehr große n — und asymptotische Eigenschaften

- $p^*(n)$ ist *scharfe* Grenzfunktion, falls für jedes $\varepsilon > 0$ gilt:

$$\lim_{n\to\infty} \mathbf{Pr}\left[G(n,p) \text{ hat } \mathcal{P}\right] = \begin{cases} 0, & \text{falls } p(n)/p^*(n) \leq 1-\varepsilon \\ 1, & \text{falls } p(n)/p^*(n) \geq 1+\varepsilon \end{cases}$$

- existiert *nicht* für jede monotone increasing property

ER-Graphen: Durchmesser ≤ 2

- $\mathcal{D}$=«hat Durchmesser ≤ 2» ist monotone increasing property
- **Satz** $p(n) = \sqrt{2}\sqrt{\frac{\ln n}{n}}$ ist scharfe Grenzfunktion für $\mathcal{D}$.
- $X_{ij} = \begin{cases} 1, & \text{falls Abstand zwischen } i \text{ und } j \text{ größer 2} \\ 0, & \text{sonst} \end{cases}$
- betrachte $X = \sum X_{ij}$
  - Durchmesser ≤ 2 genau dann, wenn $X = 0$
- $\lim_{n\to\infty} \mathbf{E}[X] = ?$
- $\mathbf{E}[X] = \binom{n}{2}(1-p)(1-p^2)^{n-2}$

ER-Graphen: Durchmesser ≤ 2

- $\mathbf{E}[X] = \binom{n}{2}(1-p)(1-p^2)^{n-2}$
- setze $p = c\sqrt{\frac{\ln n}{n}}$
- dann
$$\mathbf{E}[X] = \frac{n(n-1)}{2}\left(1 - c\sqrt{\frac{\ln n}{n}}\right)\left(1 - c^2\frac{\ln n}{n}\right)^{n-2}$$
$$\leq \frac{n^2}{2}\left(1 - \frac{c^2 \ln n}{n}\right)^n$$
$$\lim_{n\to\infty} \mathbf{E}[X] = \frac{n^2}{2}e^{-c^2 \ln n} = \frac{1}{2}n^{2-c^2}$$
- $c > \sqrt{2} \implies \lim_{n\to\infty} \mathbf{E}[X] = 0$ $\quad$ ($c < \sqrt{2}$ nicht hier …)

Zusammenhangskomponenten

Satz von Erdős- und Rényi

$t(n) = \frac{\ln n}{n}$ ist Grenzfunktion für Abwesenheit isolierter Knoten:

- Wenn $\lim_{n\to\infty} p(n)/t(n) = 0$,
  dann geht Wkt. für Existenz isolierter Knoten gegen 1
- $\lim_{n\to\infty} p(n)/t(n) \to \infty$,
  dann hat ER-Graph mit hoher Wkt. keine isolierten Knoten
  und sogar nur eine Zusammenhangskomponente.

Zusammenhangskomponenten

weitere Ergebnisse

- wenn $np < 1$,
  dann fast nie Zhk. größer als $O(\log n)$
- wenn $np = 1$,
  dann fast immer Zhk. größer als $O(\log n)$
- wenn $np \to c > 1$,
  dann fast immer eindeutige riesige Zhk., alle anderen nur $O(\log n)$ groß
- wenn $np > (1 + \varepsilon) \ln n \, (\varepsilon > 0)$,
  dann fast immer eine Zhk. der Größe n

Erwartet konstanter Knotengrad

- $p(n-1) = c$ konstant $\rightsquigarrow p(n) = \frac{c}{n-1}$
- dann für $n \to \infty$

$$
\begin{aligned}
g_d &= \binom{n-1}{d} p^d (1-p)^{n-1-d} \\
&= \frac{(n-1)!}{(n-1-d)!d!} \left(\frac{c}{n-1}\right)^d \left(1 - \frac{c}{n-1}\right)^{n-1-d} \\
&\approx \frac{(n-1-d)! \prod_{i=1}^{d}(n-i)}{(n-1-d)!(n-1)^d} \cdot \frac{c^d}{d!} \cdot e^{-c} \\
&\approx \frac{c^d}{d!} e^{-c} \qquad \text{Poisson-Verteilung}
\end{aligned}
$$

- große Graphen aus der Realität haben andere Verteilung …

Andere Anforderungen?

- z. B. so genannte skalenfreie Netzwerke «scale-free graphs»
- *Skalenfreiheit:* Anteil der Knoten mit Grad d «etwa proportional» zu $d^{-\alpha}$ für ein $\alpha > 0$
  - in Graphen aus der Realität beobachtbar
  - ER-Graphen im allgemeinen *nicht* skalenfrei
- Barabási-Albert Modell: $\alpha = 3$

Barabási-Albert Modell

- beginne mit einem «kleinen» zusammenhängenden Graphen G_{m_0} mit m_0 Knoten
- wähle $m < m_0$
- aus G_i erzeuge Graphen G_{i+1} so
  - $V_{i+1} = V_i \cup \{v_{i+1}\}$ neuer Knoten
  - wähle zufällig m Kanten $\{v_{i+1}, v_j\}$ mit $j \leq i$ wobei v_j gewählt mit Wkt. proportional zu seinem Grad in G_i
  - «preferential attachment»