3-SAT Algorithmus mit Reset

Idee: Falls X_i nicht nahe bei n ist, starte neu.

Algorithmus 3-SAT$_{Reset}$

EINGABE: $\phi(x_1, \ldots, x_n)$

1. REPEAT
2. Wähle eine zufällige Belegung.
   1. FOR $i = 1$ to $3n$
      1. Wähle beliebige nicht-erfüllte Klausel k. Falls nicht vorhanden, EXIT.
      2. Ändere für ein zufälliges Literal in k die Belegung der Variable.
3. UNTIL (erfüllende Belegung gefunden)

AUSGABE: erfüllende Belegung von $\phi(x_1, \ldots, x_n)$

3-SAT Algorithmus mit Reset

Satz 3-SAT Algorithmus mit Reset

Für erfüllbare ϕ besitzt 3-SAT$_{Reset}$ erwartete Laufzeit $\mathcal{O}(n^{\frac{3}{2}}(4/3)^n)$.

Beweis:

- Sei q_j die Ws, dass S in $3n$ Schritten erreicht wird, wenn anfangs in Schritt 2 genau j Variablen nicht mit S übereinstimmen.
- Man muss in Summe j-mal in die "richtige" Richtung gehen. D.h.

$$q_j \geq \max_{k=0,\ldots,j}\{(\tfrac{1}{3})^{j+k}(\tfrac{2}{3})^k\tbinom{j+2k}{k}\} \overset{k=j}{\geq} (\tfrac{1}{3})^{2j}(\tfrac{2}{3})^j\tbinom{3j}{j} \text{ für } 0 < j \leq n.$$

- Aus der Stirling-Formel folgt $\binom{3j}{3} \geq \frac{c}{\sqrt{j}}(\frac{27}{4})^j$, $c < 1$ konstant. D.h.

$$q_j \geq \frac{c}{\sqrt{j}}(\tfrac{1}{3})^{2j}(\tfrac{2}{3})^j(\tfrac{27}{4})^j \geq \frac{c}{\sqrt{j}}(\tfrac{1}{2})^j.$$

- Sei q die Ws, dass wir in Schleife 2.1 erfolgreich sind. Wir erhalten

$$\begin{aligned} q &= \tfrac{1}{2^n} + \textstyle\sum_{j=1}^{n} \binom{n}{j}(\tfrac{1}{2})^n q_j \geq \tfrac{c}{\sqrt{n}}(\tfrac{1}{2})^n \textstyle\sum_{j=0}^{n} \binom{n}{j}(\tfrac{1}{2})^j(1)^{n-j} \\ &= \Omega(n^{-\frac{1}{2}})(\tfrac{1}{2})^n(\tfrac{3}{2})^n = \Omega(n^{-\frac{1}{2}})(\tfrac{3}{4})^n. \end{aligned}$$

- Insgesamt benötigt man erwartet Laufzeit $\frac{1}{q}3n = \mathcal{O}(n^{\frac{3}{2}}(4/3)^n)$. $\square$

Gambler's Ruin

Szenario: Spielen bis zum Ruin

- Spieler 1 besitze ℓ_1 Euro, Spieler 2 besitze ℓ_2 Euro.
- Pro Runde gewinne jeder Spieler 1 Euro vom anderen mit Ws $\frac{1}{2}$.
- Ein Spieler gewinnt, falls er das Geld des anderen besitzt.

Frage: Mit welcher Ws q_0 gewinnt Spieler 1?

Modellierung als Random Walk:

- Wir betrachten den Gewinn von Spieler 1.
- D.h. der Random Walk beginnt in 0 und endet in $-\ell_1$ bzw. ℓ_2.

Gambler's Ruin

Satz Spielen bis zum Ruin

Spieler 1 gewinnt mit Ws $q_0 = \frac{\ell_1}{\ell_1+\ell_2}$.

Beweis:

- Sei q_j die Ws des Gewinns bei Zwischengewinn von j Euro.
- Uns interessiert q_0. Es gilt $q_{-\ell_1} = 0, q_{\ell_2} = 1$ und

$$q_j = \frac{q_{j-1}}{2} + \frac{q_{j+1}}{2} \text{ für } -\ell_1 < j < \ell_2.$$

$$\overset{(1)}{\Rightarrow} \quad 1 = q_{\ell_2} = 2q_{\ell_2-1} - q_{\ell_2-2} = 3q_{\ell_2-2} - 2_{\ell_2-3} = (\ell_2+1)q_0 - \ell_2 q_{-1}$$

$$\overset{(2)}{\Rightarrow} \quad 0 = q_{-\ell_1} = 2q_{-\ell_1+1} - q_{-\ell_1+2} = 3q_{-\ell_1+2} - 2q_{-\ell_1+3} = \ell_1 q_{-1} - (\ell_1-1)q_0$$

- Addiere ℓ_1-mal Gleichung (1) zu ℓ_2-mal Gleichung (2).
- Es folgt $\ell_1 = \ell_1(\ell_2+1)q_0 - \ell_2(\ell_1-1)q_0 = (\ell_1+\ell_2)q_0$. $\square$

Übung: Zeigen Sie, dass $q_j = \frac{\ell_1+j}{\ell_1+\ell_2}$ für $-\ell_1 < j < \ell_2$.

Random Walks auf Graphen

Definition Stationäre Zustandsverteilung

Ein Ws-Verteilung $\bar{\pi}$ für eine Markov Kette heißt *stationär*, falls $\bar{\pi} = \bar{\pi}\mathbf{P}$.

Anmerkung:

- Wir berechnen stationäres $\bar{\pi}$ durch Lösen des linearen Systems
$$\bar{\pi} = \bar{\pi}\mathbf{P}.$$
- Sei $h_{i,i}$ die erwartete Anzahl von Schritten von Zustand i nach i.
- Man kann zeigen, dass $h_{i,i} = \frac{1}{\bar{\pi}_i}$.

Random Walk auf $G = (V, E)$:

- Wir betrachten endliche, ungerichtete, zusammenhängende G.
- Zusätzlich soll G nicht bipartit sein. D.h. für jeden Knoten $v \in V$ existiert ein Pfad von v nach v ungerader Länge.
- Für jedes $v \in V$ bezeichne $d(v)$ den Grad von v.
- Angenommen wir sind im Zeitpunkt t in Knoten v. Dann gehen wir zum Zeitpunkt $t + 1$ mit Ws jeweils $\frac{1}{d(v)}$ zu einem der Nachbarn.

Random Walk besitzt stationäre Verteilung.

Satz Random Walk besitzt stationäre Verteilung.

Random Walks auf G konvergieren zu einem stationären π mit

$$\pi_v = \frac{d(v)}{2|E|}.$$

Beweis:

- Es gilt $\sum_{v \in V} \pi_v = \sum_{v \in V} \frac{d(v)}{2|E|} = 1$. D.h. π_v ist eine Ws-Verteilung.
- Sei $\mathbf{P}$ die Übergangsmatrix und $N(v)$ die Nachbarn von v.
- Das lineare Gleichungssystem $\bar{\pi} = \bar{\pi}\mathbf{P}$ ist äquivalent zu

$$\pi_v = \sum_{u \in N(v)} \pi_u \frac{1}{d(u)}.$$

- Die Setzung $\pi_v = \frac{d(v)}{2|E|}$ löst das System, denn

$$\sum_{u \in N(v)} \pi_u \frac{1}{d(u)} = \sum_{u \in N(v)} \frac{d(u)}{2|E|} \frac{1}{d(u)} = \frac{d(v)}{2|E|} = \pi_v. \qquad \square$$

Korollar

Es gilt für $h_{v,v} = \frac{2|E|}{d(v)}$ für alle $v \in V$.

Laufzeit für Pfade

Lemma Laufzeit für Pfade

Falls $(u, v) \in E$, so gilt $h_{v,u} < 2|E|$.

Beweis:

- Seien $N(u)$ die Nachbarn von u. Es gilt
$$\frac{2|E|}{d(u)} = h_{u,u} = \sum_{w \in N(u)} \frac{1}{d(u)}(1 + h_{w,u}).$$
- Es folgt $2|E| = \sum_{w \in N(u)}(1 + h_{w,u})$.
- Wegen $v \in N(u)$ folgt sicherlich $h_{v,u} < 2|E|$. $\square$

Definition Überdeckungszeit

Für $G = (V, E)$ sei T_v die erwartete Zeit bis ein Random Walk gestartet in $v \in V$ alle Knoten von V besucht.
Wir bezeichnen $T_G = \max_{v \in V}\{T_v\}$ als *Überdeckungszeit* von G.

Überdeckungszeit

Satz Überdeckungszeit

Für alle $G = (V, E)$ gilt $T_G < 4|V| \cdot |E|$.

Beweis:

- Wähle einen Spannbaum S von G. S besitzt $|V| - 1$ Kanten.
- Wir starten Tiefensuche auf S in einem beliebigem Startknoten.
- Dies liefert eine Traversierung, die jede Kante genau einmal in beide Richtungen durchläuft.
- Ferner entspricht der Startknoten dem Endknoten.
- Sei $v_0, v_1, \ldots, v_{2(|V|-1)} = v_0$ die Tour dieser Traversierung.
- Die erwartete Zeit für diese Tour ist eine obere Schranke für T_G.
- D.h. $T_G \leq \sum_{i=0}^{2|V|-3} h_{v_i,v_{i+1}} < (2|V| - 2) \cdot 2|E| < 4|V| \cdot |E|$. $\square$