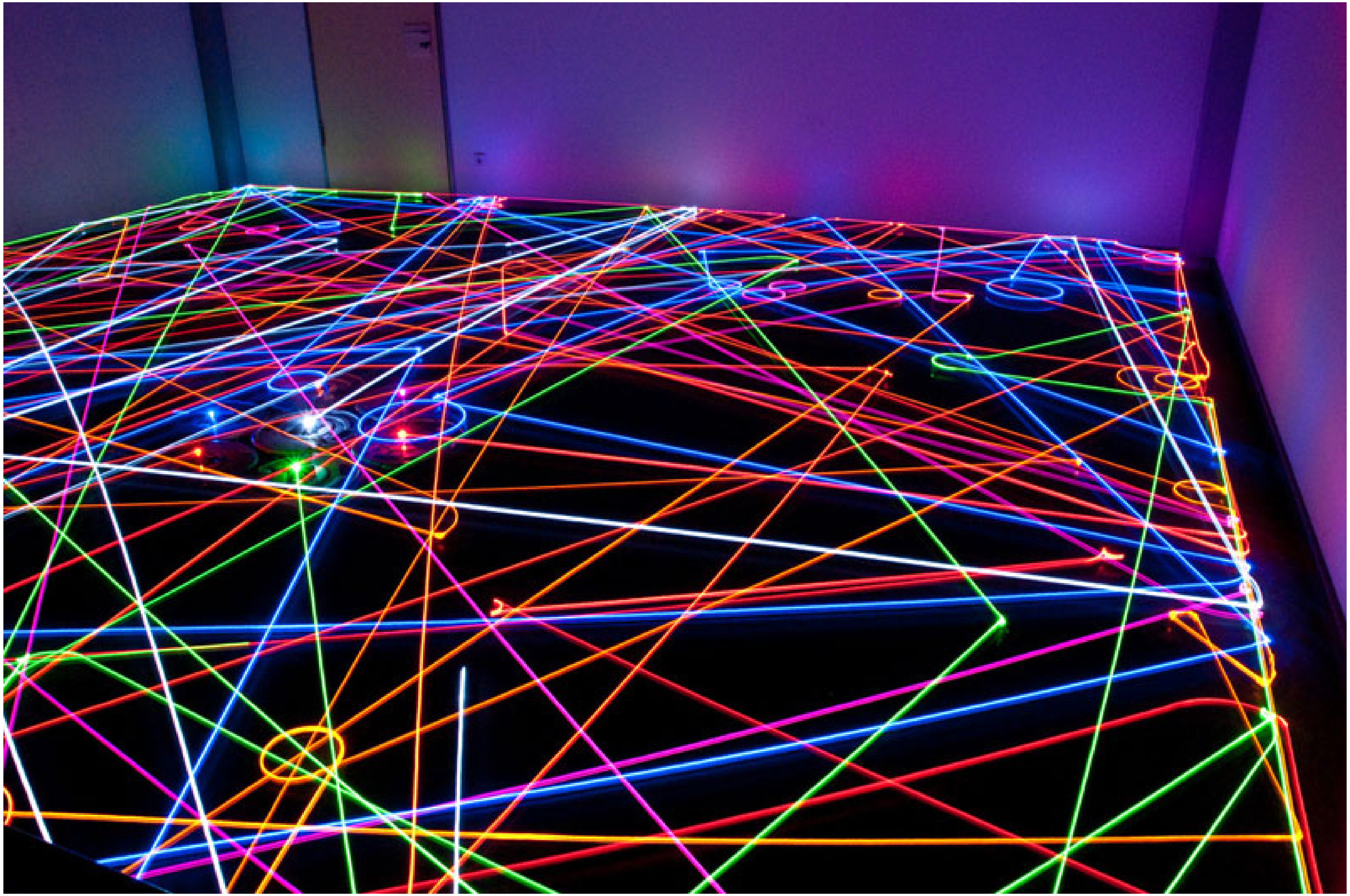
Gambler's Ruin

- Zwei Spieler A und B spielen ein Spiel um m Franken.
- Spieler A hat a Franken, Spieler B hat $b = m - a$ Franken.
- In jeder Runde wird um 1 Franken gespielt. A gewinnt eine Runde mit W'keit p, B mit W'keit $q = 1 - p$.
- Wer gewinnt mit welcher Wahrscheinlichkeit?
- Wie lange dauert das Spiel?
- Was ändert sich, wenn B unendlich viel Geld hat? („B ist die Bank")

Stationäre Verteilung: Sichtweisen

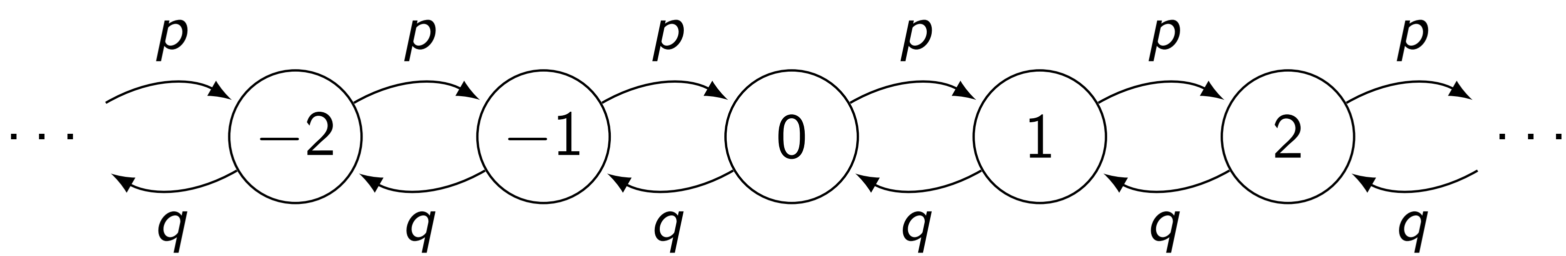
1. Eigenvektors der Übergangsmatrix zum Eigenwert 1
   - numerisch anspruchsvoll
2. Wahrscheinlichkeitsverteilung iterativ von Runde zu Runde bestimmen
   - Wann kann man aufhören?
3. **Random Walk** im Graphen der Markov-Kette mit Histogramm der Zustände
   - Wann kann man aufhören?

Roomba: „Random Walk“



Einfacher Random Walk in 1D

- Gambler's Ruin mit zwei unendlich reichen Spielern.
- Markov-Kette mit unendlich vielen Zuständen entsprechend $\mathbb{Z}$.
- Wechsel von Zustand z nach $z+1$ mit W'keit $p = \frac{1}{2}$ und umgekehrt mit $q = 1 - p = \frac{1}{2}$



- Was ist die **Hitting Time** $h_{0,1}$ von Zustand 0 nach Zustand 1?

Einfacher Random Walk $\geq$ 1D

- Markov-Kette mit Zuständen entsprechend $\mathbb{Z}^d$.

Lemma

Für einen Random Walk auf $\mathbb{Z}^d$ *für* $d = 1, 2$ *gilt* $f_{0,0} = 1$.

Lemma

Für einen Random Walk auf $\mathbb{Z}^3$ *gilt* $f_{0,0} \approx 0.34$.

Random Walk in Graphen

- Gegeben ist ein **ungerichteter verbundener Graph** $G = (V, E)$ mit n Knoten, sowie ein Startknoten $s \in V$.
- In jedem Zeitschritt wechselt man **uniform** zu einem der Nachbarknoten.
- **Cover Time** $\mathrm{cov}(s)$ = erwartete Anzahl Schritte bis ausgehend von s alle Knoten in V besucht sind.
- Verbindung zwischen **Cover Time** und **Hitting Time**:
  - Markov-Kette M mit Zuständen (v, S), wobei v = aktueller Knoten und S = bereits besuchte Knoten
  - $\mathrm{cov}(s) \approx h_{(s,\{s\}),(s,V)}$

Stationäre Verteilung von Random Walks

Lemma

Die stationäre Verteilung ist gegeben durch

$$\pi_u = \frac{d_u}{2m} \quad (d_u = \text{Grad von } u)$$

Beweis.

Annahme: Für alle j gilt $\pi_j = d_j/2m$. Dann folgt für beliebiges u

$$\pi_u = \sum_{v \in N(u)} \pi_v \cdot p_{v,u} = \sum_{v \in N(u)} \frac{d_v}{2m} \cdot \frac{1}{d_v} = \frac{d_u}{2m}.$$

Folglich ist $\pi_j = d_j/2m$ die eindeutige stationäre Verteilung. □

- Damit gilt $h_{uu} = \frac{2m}{d_u}$.

Cover Time von Random Walks

Lemma

*Für die **Cover Time** eines Graphen $G = (V, E)$ ausgehend von einem beliebigen Knoten $s \in V$ gilt $\mathrm{cov}(s) < 4m(n-1)$.*

Beweis.

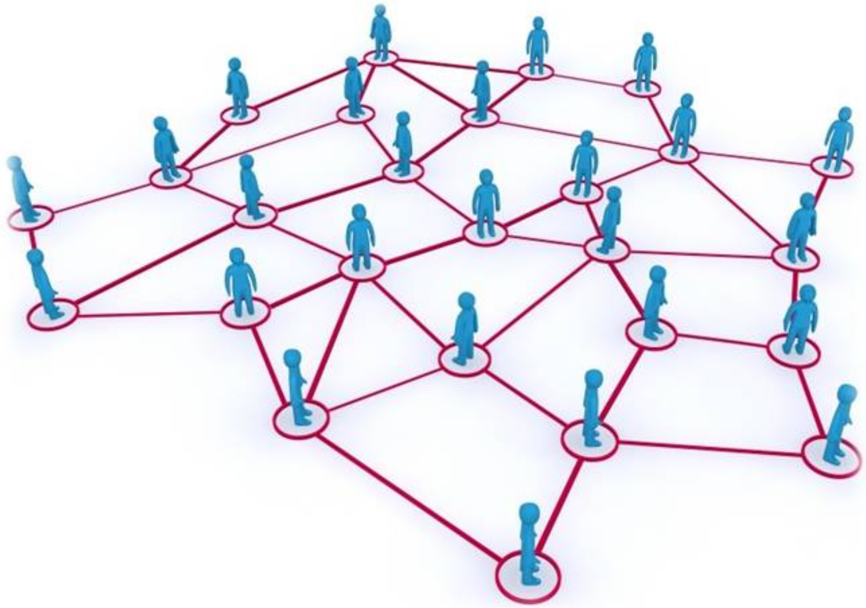
- Für $(u, v) \in E$ folgt $h_{uv} < 2m$ aus $h_{uu} = \frac{2m}{d_u}$, denn

$$\frac{2m}{d_u} = h_{uu} = \frac{1}{d_u} \sum_{w \in N(u)} (h_{wu} + 1).$$

- Es existiert eine Traversierung der Knoten von G ausgehend von s, die $2n - 2$ Kanten besucht (beliebigem Spannbaum mit Wurzel s doppelt ablaufen). Mit $h_{uv} < 2m$ für $(u, v) \in E$ folgt

$$\mathrm{cov}(s) < (2n-2) \cdot 2m = 4m(n-1). \qquad \square$$

Cover Time verschiedener Graphklassen



Wie schnell werden verschiedene Graph-Klassen exploriert?

- Clique $\Rightarrow$ Coupon Collector
- Lineare Liste
- Baum
- Worst-Case?

Random Walks vs. elektrische Schaltkreise

- Kante e im Graphen $\widehat{=}$ Widerstand mit $R_e = 1\Omega$
- Zwei Knoten u und v werden über eine 1V-Batterie verbunden.
- $R(u, v)$ = Ersatzwiderstand des Netzwerks $= \frac{1\text{V}}{I_{u,v}}$

Lemma

Für die ***Commute Time*** c_{uv} *zwischen zwei Knoten u und v in einem Graphen mit m Kanten gilt*

$$c_{uv} = 2m \cdot R(u, v).$$

Foster's Network Theorem

- **Hinzufügen**/**entfernen** einer Kante im Graph kann den Ersatzwiderstand nur **reduzieren** bzw. **erhöhen**

Satz (Foster's Network Theorem)

Für jeden verbundenen Graphen $G = (V, E)$ mit n Knoten gilt

$$\sum_{(u,v)\in E} R(u, v) = n - 1.$$

Anwendung: PageRank (1)

Wie rankt Google Webseiten?

- **Gesucht**: Vektor $\mathbf{v} \in \mathbb{R}^n$ für n Webseiten, so dass $v_i \geq 0$ die Wichtigkeit von Seite i misst.
- Annahme: „Link von Seite i nach j“ $=$ „i empfiehlt j“
- Ansatz: $A = (n \times n)$-Matrix mit

$$a_{ij} = \begin{cases} 1 & \text{wenn Seite } i \text{ nach } j \text{ linkt} \\ 0 & \text{sonst} \end{cases}$$

- Ranking: $\mathbf{v} = \mathbf{1}^n \cdot A \quad (\mathbf{v}$ und $\mathbf{1}^n$ wie üblich Zeilenvektor)
- **Schwäche**: Link-Sammlungen haben viel Gewicht

Anwendung: PageRank (2)

- **Verbesserung 1**: „Empfehlung" von Seite i normalisieren durch Anzahl Links ausgehend von i.

$$d_i = \sum_j a_{ij}$$

- Neue Matrix P:

$$p_{ij} = \begin{cases} 1/d_i & \text{wenn Seite } i \text{ nach } j \text{ linkt} \\ 0 & \text{sonst} \end{cases}$$

- Ranking $\mathbf{v} = \mathbf{1}^n \cdot P$
- **Schwäche**: Wichtigkeit der verlinkenden Seite wird ignoriert.

Anwendung: PageRank (3)

- **Verbesserung 2**: Iterative Berechnung
- Gewicht der Empfehlung hängt von Wichtigkeit der empfehlenden Seite ab.

$$v_i = \sum_j p_{ji} \cdot v_j$$

- Ranking $\mathbf{v} = \mathbf{v} \cdot P$
- $\mathbf{v}$ ist Eigenvektor zum Eigenwert $1 \Rightarrow$ stationäre Verteilung!
- **Schwäche**: „Link-Markov-Kette“ muss nicht ergodisch sein.

Anwendung: PageRank (4)

- **Verbesserung 3**: „Random Surfer“
- Annahme: In jedem Schritt wirft der User eine Münze mit W'keit für Kopf r und W'keit für Zahl $1 - r$.
- Wirft er Kopf, so wechselt er uniform zu einer verlinkten Seite. Wirft er Zahl, so wechselt er zufällig zu einer beliebigen aller n Seiten.
- Neue Matrix:

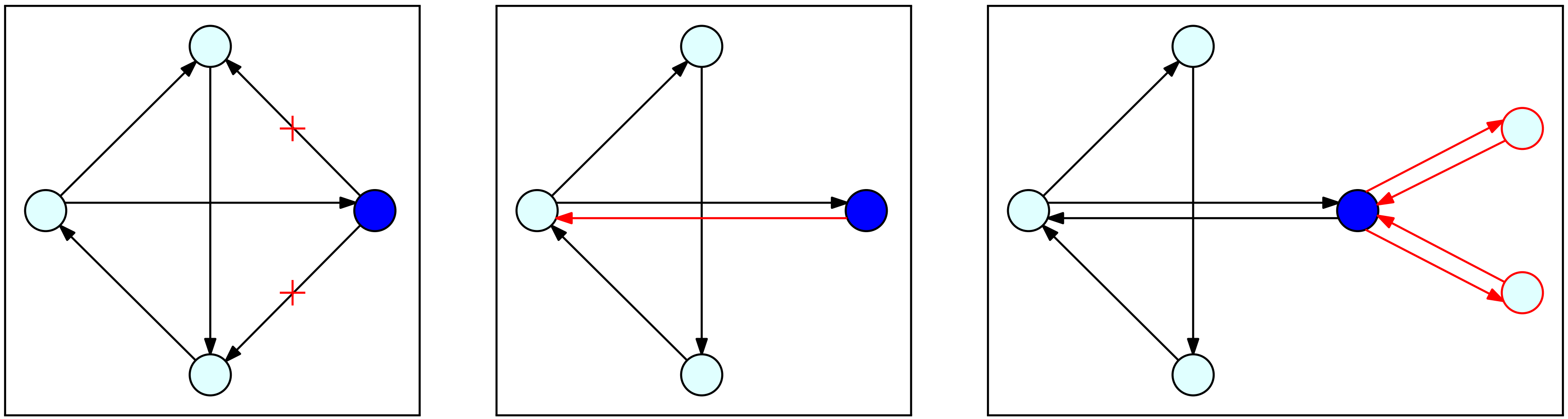
$$Q = r \cdot P + (1 - r) \cdot \begin{pmatrix} 1/n & \cdots & 1/n \\ \vdots & \ddots & \vdots \\ 1/n & \cdots & 1/n \end{pmatrix}^{n \times n}$$

- Die zugehörige Markov-Kette M_Q ist nun ergodisch.
- Ranking $\mathbf{v} = \mathbf{v} \cdot Q$ ist stationäre Verteilung von M_Q.

Google Bombing

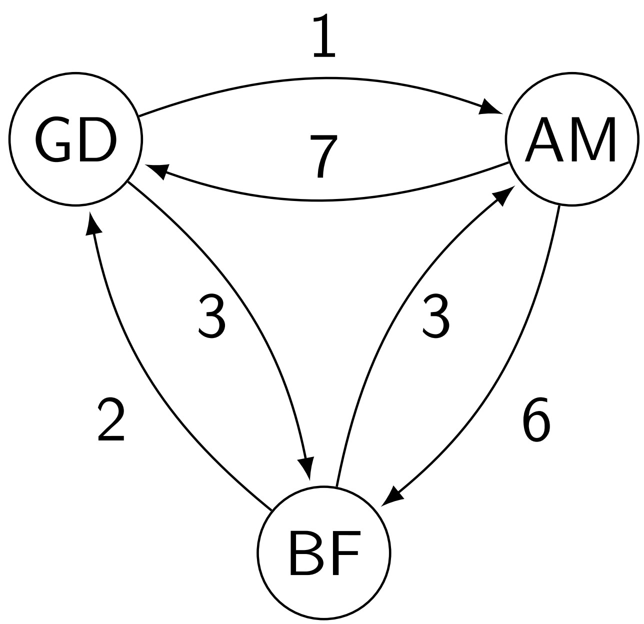
- New Oxford American Dictionary, 2005:

  „The activity of designing Internet links that will bias search engine results so as to create an inaccurate impression of the search target.“
- **Schritt 1:** Ausgehende Links entfernen.
- **Schritt 2:** $\pi_x = 1/h_{xx} \Rightarrow$ Auf Seiten linken, die „schnell“ zurücklinken.
- **Schritt 3:** Seiten erstellen, die direkt zurücklinken (Sybil-Angriff)



Exkursion: Rank Aggregation

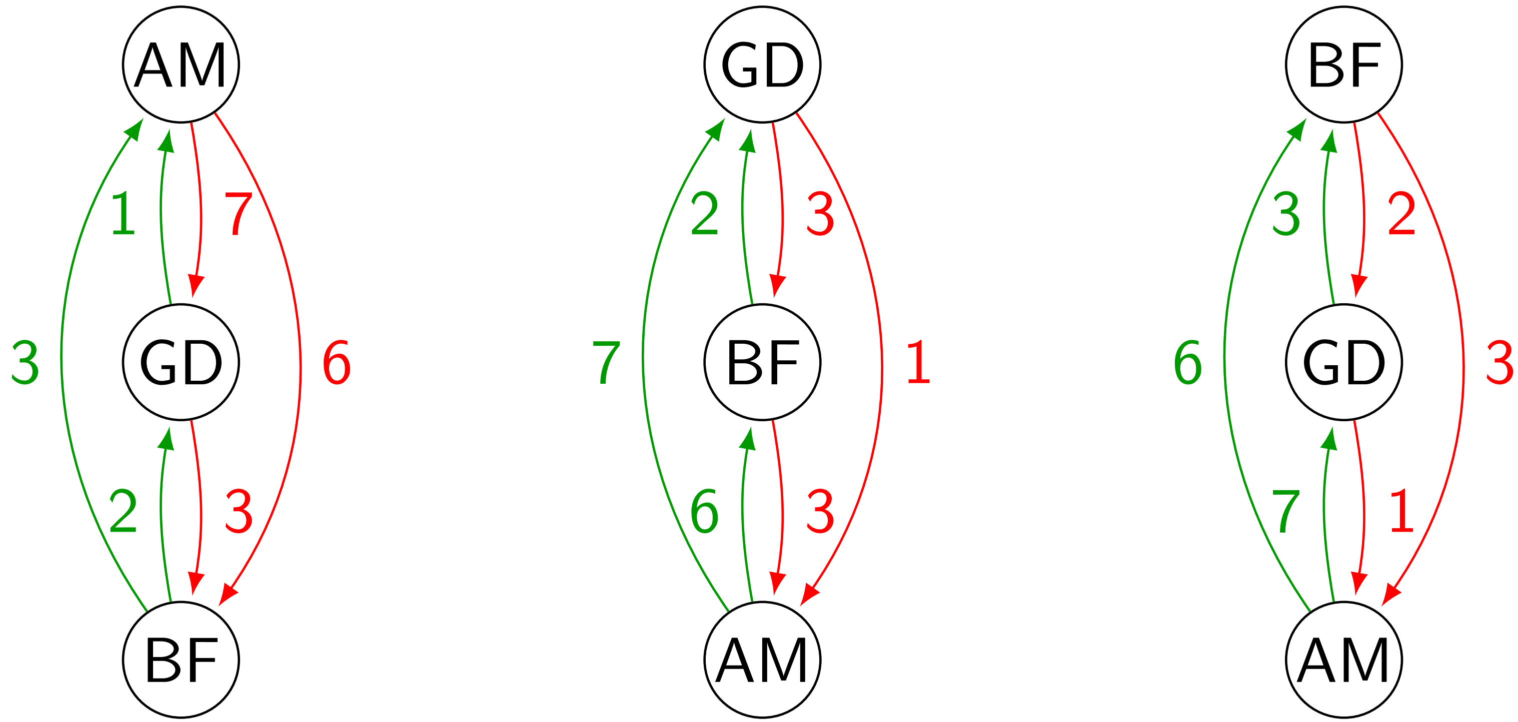
- Verschiedene User bewerten drei Filme relativ zueinander
  (GD = Groundhog Day, AM = 12 Angry Men, BF = Back To The Future)



- Welcher Film ist der beste?

Feedback Arc Set Problem

- Filme so ordnen, dass „Gegenmeinungen“ minimiert werden.

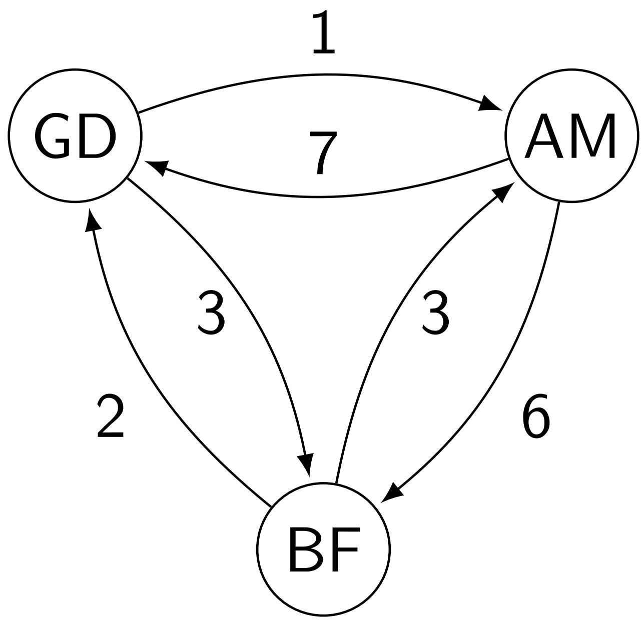


$\sum = 16$ $\sum = 7$ $\sum = 6$

- **Back To The Future** ist der beste Film.
- Minimum Feedback Arc Set Problem ist **NP-schwer**.

Stationäre Verteilung

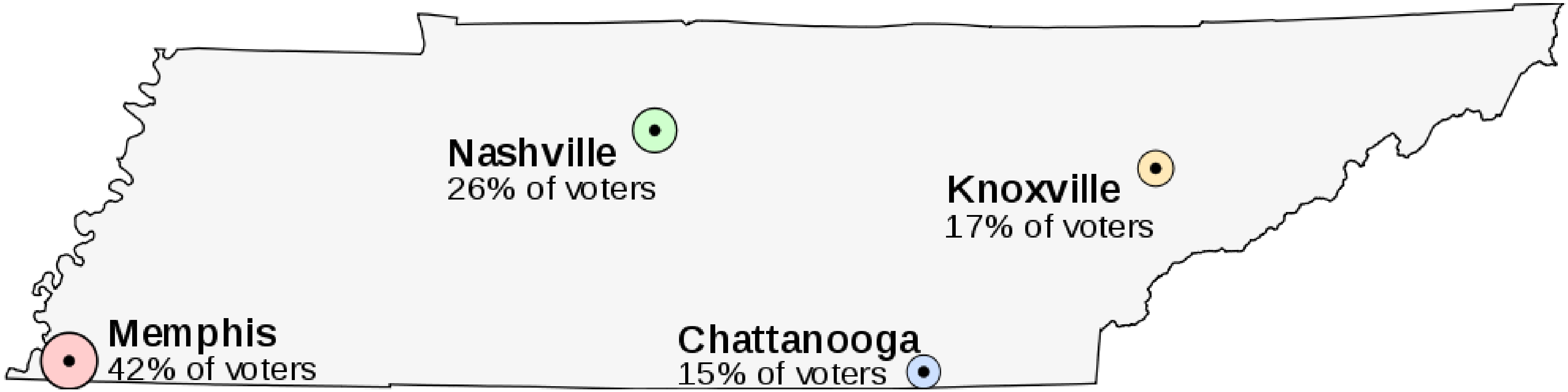
- Abstimmungsgraph als Markov-Kette interpretieren und stationäre Verteilung bestimmen.



$$\pi(GD, BF, AM) = \left(\frac{47}{106}, \frac{45}{107}, \frac{14}{106}\right)$$

- **Groundhog Day** ist der beste Film.

Tennessees Hauptstadt



42% of voters (close to Memphis)	26% of voters (close to Nashville)	15% of voters (close to Chattanooga)	17% of voters (close to Knoxville)
1. **Memphis**	1. **Nashville**	1. **Chattanooga**	1. **Knoxville**
2. Nashville	2. Chattanooga	2. Knoxville	2. Chattanooga
3. Chattanooga	3. Knoxville	3. Nashville	3. Nashville
4. Knoxville	4. Memphis	4. Memphis	4. Memphis

Quelle: Wikipedia

Arrow-Theorem

Satz

Es existiert kein Rangordnungssystem, dass die folgenden drei „Fairness"-Kriterien gleichzeitig erfüllt:

- *Wenn jeder Wähler Alternative A gegenüber Alternative B bevorzugt, dann bevorzugt auch das System A gegenüber B.*
- *Für das Ranking von zwei Alternativen A und B sind ausschließlich die Präferenzen der Wähler bezüglich dieser beiden Alternativen relevant.*
- *Es gibt keinen Diktator, der allein über die Präferenzordnung der Gesellschaft entscheidet.*