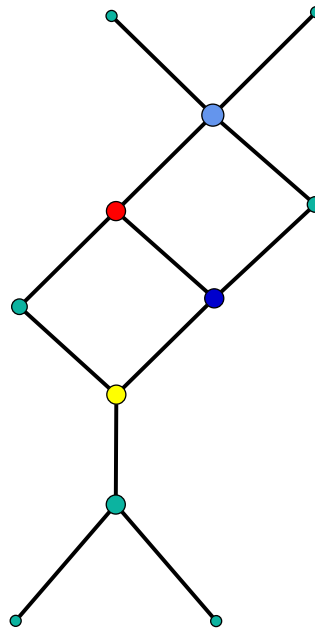


Kapitel 2

Zentralität

“One of the primary uses of graph theory in social network analysis is the identification of the most important actors in a social network.”

(Wasserman/Faust, 1994)



[*AKTIV: Wer ist zentral? Warum?*]

Die relative Wichtigkeit der Knoten eines Multigraphen interessiert auch in zahlreichen anderen Kontexten, wir wollen uns jedoch vordringlich mit den Gemeinsamkeiten der dahinter stehenden Methoden beschäftigen. Da der modellierte Gegenstand – und insbesondere seine nicht durch den Multigraphen erfassten Eigenschaften – für uns unbedeutend sind, wollen wir nur strukturell verschiedene Multigraphen unterscheiden.

2.1 Definition (Isomorphie)

Zwei Multigraphen $G = (V, E)$ und $G' = (V', E')$ heißen isomorph, $G \cong G'$, falls es eine Bijektion $\alpha : V \rightarrow V'$ mit $(v, w) \in_k E \iff \alpha(v, w) := (\alpha(v), \alpha(w)) \in_k E'$ gibt. In diesem Fall nennen wir α einen zugehörigen (Multigraphen-)Isomorphismus und schreiben auch $\alpha(G) = G'$. Falls $V = V'$ heißt α auch Automorphismus von G .

2.2 Definition (Strukturindex)

Seien \mathcal{K} eine unter Bildung von Zusammenhangskomponenten abgeschlossene Klasse nicht-isomorpher Multigraphen und $\mathbb{R}_{\geq 0}^*$ die Menge aller Vektoren über den nicht-negativen reellen Zahlen. Eine Funktion $s : \mathcal{K} \rightarrow \mathbb{R}_{\geq 0}^*$ heißt (Knoten- bzw. Kanten-)Strukturindex auf \mathcal{K} , falls

- für alle $G = (V, E) \in \mathcal{K}$

$$s(G) \in \mathbb{R}_{\geq 0}^V \quad \text{bzw.} \quad s(G) \in \mathbb{R}_{\geq 0}^E$$

(d.h. s ist ein Knoten- bzw. Kantenindex),

- für alle $G = (V, E) \in \mathcal{K}$ und Automorphismen α von G

$$s(G)_v = s(\alpha(G))_{\alpha(v)} \quad \text{bzw.} \quad s(G)_e = s(\alpha(G))_{\alpha(e)}$$

für alle $v \in V$ bzw. $e \in E$

(d.h. s bewertet ausschließlich die Struktur des Graphen), und

- für alle $G = (V, E) \in \mathcal{K}$ und Zusammenhangskomponenten $C = (V_C, E_C) \subseteq G$

$$s(G)_v \cdot s(C)_w = s(C)_v \cdot s(G)_w \quad \text{bzw.} \quad s(G)_e \cdot s(C)_f = s(C)_e \cdot s(G)_f$$

für alle $v, w \in V_C$ bzw. $e, f \in E_C$ (d.h. s ist konsistent).

Gibt es zu einem Multigraph $G \notin \mathcal{K}$ einen isomorphen Multigraphen $\alpha(G) \in \mathcal{K}$, so definieren wir $s(G) = s(\alpha(G))$.

Nicht jeder Strukturindex misst etwas, das unserem intuitiven Verständnis von „Zentralität“ entspricht. Wir formulieren daher (sehr schwach erscheinende) Mindestanforderungen, die außerdem noch zwischen grundsätzlich verschiedenen Formen struktureller Zentralität unterscheiden.

2.3 Definition (Zentralität)

Ein Strukturindex c auf einer unter Hinzufügen von Kanten abgeschlossenen Klasse \mathcal{K} von Multigraphen heißt (schwacher) (Knoten-)Zentralität(sindex), falls eine der drei folgenden Bedingungen für alle $G = (V, E) \in \mathcal{K}$ und $v, w \in V$ gilt:

- \rightarrow • Für alle $x \in V$

$$c(G)_v \geq c(G)_x \implies c(G + (v, w))_v \geq c(G + (v, w))_x$$

(Zentralität basierend auf Einfluss, Zugang, usw.)

- \rightarrow • Für alle $x \in V$

$$c(G)_w \geq c(G)_x \implies c(G + (v, w))_w \geq c(G + (v, w))_x$$

(Zentralität basierend auf Reputation, Erreichbarkeit, usw.)

- \rightarrow • Für alle $x \in V$

$$\begin{aligned} c(G)_v + c(G)_w &\geq c(G)_x \\ \implies c(G + (v, w))_v + c(G + (v, w))_w &\geq c(G + (v, w))_x \end{aligned}$$

(Zentralität basierend auf Kontrolle, Mediation, usw.)

Wir schreiben auch kurz $c \in \bullet \rightarrow \circ(\mathcal{K})$, $c \in \circ \rightarrow \bullet(\mathcal{K})$, bzw. $c \in \bullet \rightarrow \bullet(\mathcal{K})$ und sagen, c sei vom Typ $\bullet \rightarrow \circ$, $\circ \rightarrow \bullet$ bzw. $\bullet \rightarrow \bullet$.

Normierte Zentralitäten erlauben den Vergleich der relativen Wichtigkeit von Knoten in Netzwerken unterschiedlicher Größe, aber auch bzgl. unterschiedlicher Indizes.

2.4 Definition (Normalisierung)

Ein Knoten- bzw. Kanten-Strukturindex s auf \mathcal{K} heißt normiert, falls

$$\sum_{v \in V} s(G)_v = 1 \quad \text{bzw.} \quad \sum_{e \in E} s(G)_e = 1$$

für alle $G = (V, E) \in \mathcal{K}$.

2.5 Bemerkung

Zu jeder Zentralität c auf \mathcal{K} gibt es eine eindeutige normierte Zentralität \hat{c} auf \mathcal{K} mit

$$\hat{c}(G)_v = \begin{cases} \frac{c(G)_v}{\sum_{x \in V} c(G)_x} & \text{falls } c(G)_x > 0 \text{ für ein } x \in V \\ \frac{1}{n} & \text{sonst .} \end{cases}$$

Diese kann als Verteilung der Zentralität (der Wichtigkeit, des Einflusses, usw.) angesehen werden.

2.6 Beispiel (Gradzentralität)

Man prüft leicht nach, dass der Eingangsgrad eine $\circ \rightarrow \bullet$ -Zentralität, der Ausgangsgrad eine $\bullet \rightarrow \circ$ -Zentralität, und der Knotengrad eine $\circ \rightarrow \bullet$ -, $\bullet \rightarrow \circ$ - und auch $\bullet \rightarrow \bullet$ -Zentralität ist.

Die Gradzentralitäten sind *lokal* in dem Sinne, dass sie ausschließlich die Nachbarschaft eines Knotens berücksichtigen. Wir werden zwei verschiedene Arten von Verallgemeinerungen betrachten, welche die globale Struktur des Multigraphen berücksichtigen. Bei der ersten gehen die Entfernungen zu anderen Knoten ein, bei der zweiten die Bewertungen der Nachbarn.

Offene Frage: Folgt aus $c \in \circ \rightarrow \bullet(\mathcal{K}) \cap \bullet \rightarrow \circ(\mathcal{K}) \cap \bullet \rightarrow \bullet(\mathcal{K})$, dass c monoton im Knotengrad ist?

2.1 Abstandszentralitäten

Als Variante der Gradzentralitäten kann die Größe der Nachbarschaften eines Knotens zur Zentralitätsbestimmung herangezogen werden. Eine Möglichkeit, andere Knoten über die Nachbarschaft hinaus in die Bewertung einzu beziehen, besteht dann darin, die Zahl der Nachbarn von Nachbarn, die selbst keine Nachbarn des betrachteten Knotens sind, hinzuzuzählen, aber z.B. geringer zu gewichten. Deren noch nicht berücksichtigte Nachbarn könnten mit noch geringerem Gewicht hinzugezählt werden, usw.

2.7 Definition (Abstand)

Sei $G = (V, E)$ ein Multigraph. Gibt es für zwei Knoten $s, t \in V$ einen Weg von s nach t , so heißt die kürzeste Länge eines (s, t) -Weges Abstand (auch: Distanz), $d_G(s, t)$, von s nach t . Gibt es keinen (s, t) -Weg, so vereinbaren wir $d_G(s, t) = \infty$.

Um Schwierigkeiten mit unendlichen Abständen zu vermeiden, betrachten wir in diesem Abschnitt meist nur stark zusammenhängende Multigraphen. Sei daher im folgenden \mathcal{S} die Klasse der stark zusammenhängenden Multigraphen und \mathcal{G} die Klasse aller Multigraphen.

2.8 Definition (Exzentrizität, Durchmesser, Radius)

Für einen Multigraphen $G = (V, E)$ definieren wir

$$\begin{aligned} \text{die } \underline{\text{Exzentrizität}} \quad e_G(v) &= \max\{d_G(v, w), d_G(w, v) : w \in V\} , \\ \text{den } \underline{\text{Radius}} \quad rad(G) &= \min\{e_G(v) : v \in V\} \quad \text{und} \\ \text{den } \underline{\text{Durchmesser}} \quad diam(G) &= \max\{e_G(v) : v \in V\} . \end{aligned}$$

Üblicherweise wird das Zentrum $C(G) \subseteq V$ eines zusammenhängenden ungerichteten Graphen (d.h. eines ungerichteten Graphen mit endlichem Radius) als die Menge der Knoten mit kleinster Exzentrizität definiert,

$$C(G) = \{v \in V : e_G(v) = rad(G)\} .$$

Wir können diesen Zentrumsbegriff zu einem Zentralitätsindex für stark zusammenhängende Multigraphen erweitern, indem wir z.B. allen Knoten im analog definierten Zentrum den Wert 1, und allen außerhalb den Wert 0 zuweisen.

Frage: Um welche Sorte Zentralität handelt es sich dann?

Der folgende Index macht feinere Unterschiede, indem statt der Exzentrizität (also des maximalen Abstands) der mittlere Abstand zu anderen Knoten zugrunde gelegt wird.

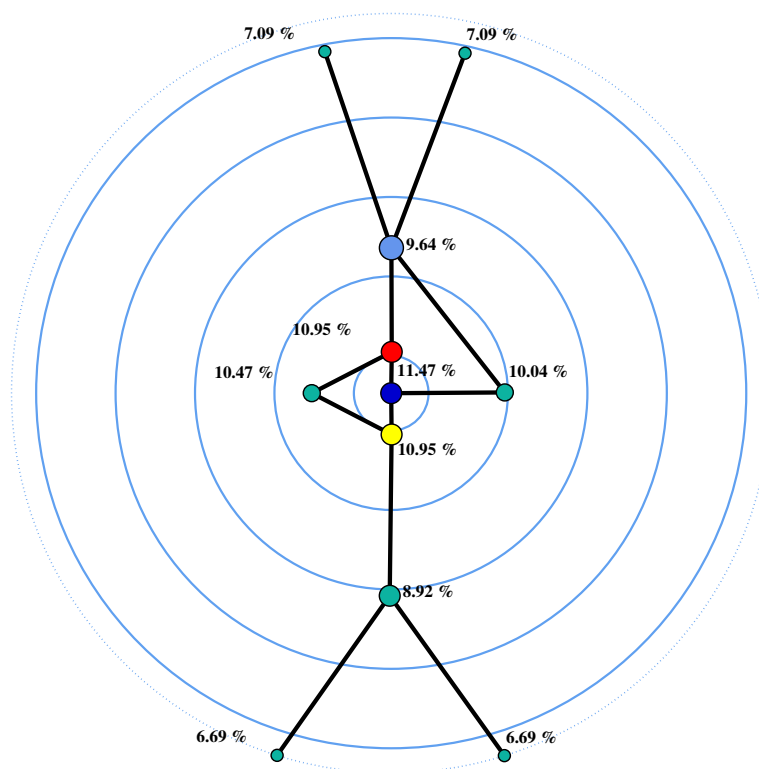
2.9 Definition (Closeness; Beauchamp 1965)

Die Closeness-Zentralität c_C ist definiert durch

$$c_C(G)_v = \frac{1}{\sum_{t \in V} d_G(v, t)}$$

für alle $G = (V, E) \in \mathcal{S}$, wobei $\frac{1}{0} = 1$ gelte.

Frage: Warum wird Closeness-Zentralität nur auf stark zusammenhängenden Multigraphen definiert?



normierte Closeness-Zentralität im Beispielgraphen

2.10 Satz

$$c_C \in \bullet \rightarrow \circ(\mathcal{S})$$

■ **Beweis:** Seien $G = (V, E) \in \mathcal{S}$, $c = c_C(G)$, $v, w \in V$, $G' = G + (v, w)$ und $c' = c_C(G')$. Wir müssen zeigen, dass

$$c_v \geq c_x \implies c'_v \geq c'_x$$

für alle $x \in V$. Für ein festes $x \in V$ folgt aus $c_v \geq c_x$ zunächst $\sum_{t \in V} d_G(v, t) \leq \sum_{t \in V} d_G(x, t)$. Wir zeigen, dass

$$d_G(v, t) - d_{G'}(v, t) \geq d_G(x, t) - d_{G'}(x, t) \geq 0$$

für alle $t \in V$, woraus dann $\sum_{t \in V} d_{G'}(v, t) \leq \sum_{t \in V} d_{G'}(x, t)$ und somit die Behauptung folgt. Die hintere Ungleichung ist klar, da wir eine Kante hinzufügen und also keinen Abstand vergrößern. Ist $d_G(x, t) - d_{G'}(x, t) > 0$, so muss jeder kürzeste (x, t) -Weg in G' die neue Kante (v, w) benutzen. Es gilt daher $d_{G'}(x, t) = d_{G'}(x, v) + d_{G'}(v, t)$ und wir erhalten

$$\begin{aligned} d_G(x, t) - d_{G'}(x, t) &= d_G(x, t) - [d_{G'}(x, v) + d_{G'}(v, t)] \\ &\leq [d_G(x, v) + d_G(v, t)] - [d_{G'}(x, v) + d_{G'}(v, t)] \\ &= d_G(v, t) - d_{G'}(v, t) . \end{aligned}$$

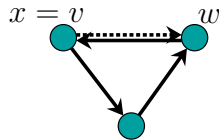
Die letzte Gleichung gilt, weil eine neue Kante (v, w) den Abstand von x nach v nicht verkleinert. □

2.11 Satz

$$c_C \notin \circ \rightarrow \bullet(\mathcal{S}) \cup \bullet \rightarrow \bullet(\mathcal{S})$$

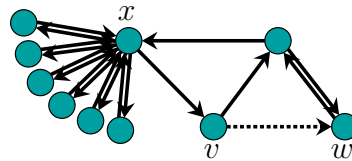
■ **Beweis:**

$$c_C \notin \circ \rightarrow \bullet(\mathcal{S}) :$$



$$\begin{aligned} c_w &= \frac{1}{3} \geq \frac{1}{3} = c_v \\ c'_w &= \frac{1}{3} < \frac{1}{2} = c'_v \end{aligned}$$

$$c_C \notin \bullet \rightarrow \bullet(\mathcal{S}) :$$



$$\begin{aligned} c_v + c_w &= \frac{1}{23} + \frac{1}{24} \geq \frac{1}{12} = c_x \\ c'_v + c'_w &= \frac{1}{22} + \frac{1}{24} < \frac{1}{11} = c'_x \end{aligned}$$

□

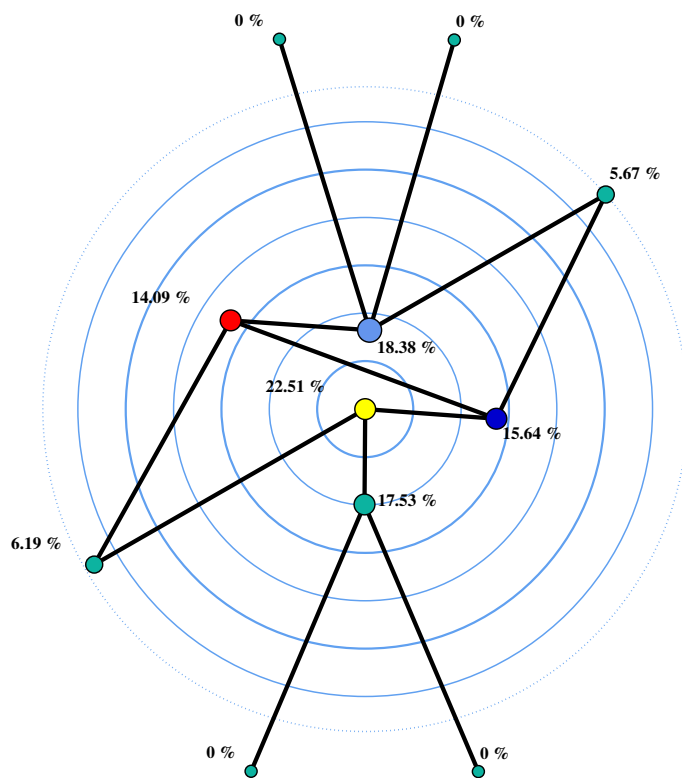
Aus c_C erhalten wir dennoch leicht eine $\circ \rightarrow \bullet$ -Zentralität, indem wir in der Definition die Distanzen *von* durch Distanzen *nach* v ersetzen. Die Bedingungen an $\bullet \rightarrow \bullet$ -Zentralitäten sind motiviert durch den folgenden, sehr populären Index auf der Klasse aller nicht-isomorphen Multigraphen \mathcal{G} .

2.12 Definition (Betweenness; Anthonisse 1971, Freeman 1977)

Die Betweenness-Zentralität c_B ist definiert durch

$$c_B(G)_v = \sum_{s,t \in V} \frac{\sigma_G(s,t|v)}{\sigma_G(s,t)}$$

für alle $G = (V, E) \in \mathcal{G}$. Dabei bezeichne $\sigma_G(s,t)$ die Anzahl der kürzesten Wege von s nach t , $\sigma_G(s,t|v)$ die Anzahl der kürzesten (s,t) -Wege, die v als inneren Knoten enthalten (d.h. v liegt auf dem Weg, aber $v \neq s, t$), und es gelte $\frac{0}{0} = 0$.



normierte Betweenness-Zentralität im Beispielgraphen

2.13 Satz

$$c_B \in \bullet \rightarrow \bullet(\mathcal{G})$$

■ **Beweis:** (verbesserungsbedürftig!) Seien wieder $G = (V, E) \in \mathcal{G}$, $v, w \in V$ und $G' = G + (v, w)$. Die Behauptung ist bewiesen, wenn wir zu jedem Paar $s, t \in V$ für alle $x \in V$ zeigen können, dass

$$\frac{\sigma_{G'}(s, t|v) + \sigma_{G'}(s, t|w) - \sigma_{G'}(s, t|x)}{\sigma_{G'}(s, t)} \geq \frac{\sigma_G(s, t|v) + \sigma_G(s, t|w) - \sigma_G(s, t|x)}{\sigma_G(s, t)}.$$

Wir wählen $s, t, x \in V$ beliebig und unterscheiden danach, ob die neue Kante (v, w) den Abstand von s nach t verringert.

1. *Fall:* Bleibt der Abstand gleich, so ist jeder kürzeste (s, t) -Weg in G auch ein solcher in G' . Es gilt daher $\sigma_{G'}(s, t) = \sigma_G(s, t) + k$, wobei (v, w) auf den $k \geq 0$ neuen kürzesten (s, t) -Wegen in G' liegt.

Ist $s = w$ oder $t = v$, dann ist $k = 0$, sodass beide Seiten der Ungleichung identisch sind. Für $s = v$ und $t = w$ bestehen alle (alte und neue) kürzeste (s, t) -Wege nur aus einer Kante. Weder v oder w noch x sind dann innerer Knoten, sodass beide Seiten identisch Null sind.

Im Fall $s = v$ und $t \neq w$ ist $\sigma_G(s, t|v) = \sigma_{G'}(s, t|v) = 0$, aber wieder $\sigma_{G'}(s, t|w) = \sigma_G(s, t|w) + k$ und $\sigma_{G'}(s, t|x) \leq \sigma_G(s, t|x) + k$ (wobei $k = \sigma_G(w, t)$). Die Ungleichung folgt sofort, da immer $\sigma_G(s, t|w) - \sigma_G(s, t|x) \leq \sigma_G(s, t)$. Der Fall $s \neq v$ und $t = w$ geht analog.

Ist schließlich $\{s, t\} \cap \{v, w\} = \emptyset$, so sind v und w innere Knoten aller k neuen kürzesten (s, t) -Wege, d.h. wir haben $\sigma_{G'}(s, t|v) = \sigma_G(s, t|v) + k$ und $\sigma_{G'}(s, t|w) = \sigma_G(s, t|w) + k$. Da nicht jeder neue Weg auch x enthalten muss, ist $\sigma_{G'}(s, t|x) \leq \sigma_G(s, t|x) + k$. Für $\sigma_G(s, t|v) + \sigma_G(s, t|w) - \sigma_G(s, t|x) \leq \sigma_G(s, t)$ oder $\sigma_G(s, t|x) = 0$ folgt damit die Ungleichung. Andernfalls ist $0 < \sigma_G(s, t|x) < \min\{\sigma_G(s, t|v), \sigma_G(s, t|w)\}$, sodass x in G und G' entweder auf einem kürzesten (s, v) -Weg oder auf einem kürzesten (w, t) -Weg liegt, jeweils jedoch nicht auf allen. Die Anzahl $\sigma_{G'}(s, t|x) - \sigma_G(s, t|x)$ der neuen kürzesten (s, t) -Wege über x wächst damit auch nur anteilig und ist dann höchstens $k \cdot \frac{\sigma_G(s, t|x)}{\sigma_G(s, t|v)}$ bzw. $k \cdot \frac{\sigma_G(s, t|x)}{\sigma_G(s, t|w)}$. Die Ungleichung folgt durch Einsetzen.

2. *Fall:* Ist $d_{G'}(s, t) < d_G(s, t)$, dann benutzt jeder der $k > 0$ kürzesten (s, t) -Wege in G' die neue Kante (v, w) , d.h. $\sigma_{G'}(s, t) = k$, kein kürzester (s, t) -Weg in G ist auch ein solcher in G' sowie $s \neq w$ und $t \neq v$.

Für $\{s, t\} = \{v, w\}$ sind dann also $s = v$ und $t = w$, woraus $\sigma_{G'}(s, t|x) = 0$ folgt. Die linke Seite der Ungleichung ist damit entweder 0 oder 2, jedenfalls aber nicht kleiner als die rechte.

Für $\{s, t\} \cap \{v, w\} = \emptyset$ sind auch $\sigma_{G'}(s, t|v) = \sigma_{G'}(s, t|w) = k$. Wir haben also $1 = \frac{\sigma_{G'}(s, t|v)}{\sigma_{G'}(s, t)} \geq \frac{\sigma_G(s, t|v)}{\sigma_G(s, t)}$ und entsprechend für w . Liegt nun x auf keinem kürzesten (s, t) -Weg in G' , ist also $0 = \sigma_{G'}(s, t|x) \leq \sigma_G(s, t|x)$, gilt damit die Ungleichung. Ist andererseits $\sigma_{G'}(s, t|x) > 0$, dann muss x wieder auf einem kürzesten (s, v) - oder (w, t) -Weg liegen. Da sich deren Anzahl aber in G' gegenüber G nicht verändert, gilt $\frac{\sigma_{G'}(s, t|v) - \sigma_{G'}(s, t|x)}{\sigma_{G'}(s, t)} \geq \frac{\sigma_G(s, t|v) - \sigma_G(s, t|x)}{\sigma_G(s, t)}$ bzw. $\frac{\sigma_{G'}(s, t|w) - \sigma_{G'}(s, t|x)}{\sigma_{G'}(s, t)} \geq \frac{\sigma_G(s, t|w) - \sigma_G(s, t|x)}{\sigma_G(s, t)}$, woraus die Ungleichung folgt.

Abschließend betrachten wir wieder nur den Fall $s = v$ und $t \neq w$. Es gilt $\sigma_G(s, t|v) = \sigma_{G'}(s, t|v) = 0$ und $\sigma_{G'}(s, t|w) = k = \sigma_{G'}(s, t)$. Da alle kürzesten (s, t) -Wege in G' mit der Kante $(s, w) = (v, w)$ beginnen, folgt $\sigma_{G'}(s, t|x) \leq \sigma_{G'}(s, t|w) = \sigma_G(w, t)$. Liegt x auf keinem kürzesten (w, t) -Weg, ist $\sigma_{G'}(s, t|x) = 0$ und die Ungleichung gilt. Andernfalls können wir wie gerade eben $\frac{\sigma_{G'}(s, t|w) - \sigma_{G'}(s, t|x)}{\sigma_{G'}(s, t)} \geq \frac{\sigma_G(s, t|w) - \sigma_G(s, t|x)}{\sigma_G(s, t)}$ schließen. \square

2.14 Bemerkung

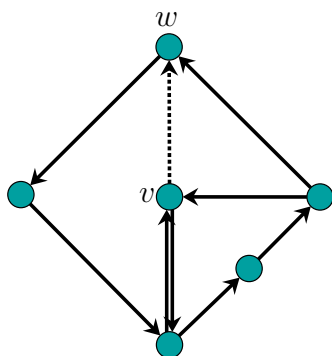
Wir haben sogar gezeigt, dass $[c_B(G')_v + c_B(G')_w] - [c_B(G)_v + c_B(G)_w] \geq c_B(G')_x - c_B(G)_x$, d.h. von einer neuen Kante (v, w) profitieren v und w zusammen mehr als jeder andere Knoten.

2.15 Satz

$$c_B \notin \circ \rightarrow \bullet(\mathcal{S}) \cup \bullet \rightarrow \circ(\mathcal{S})$$

■ Beweis:

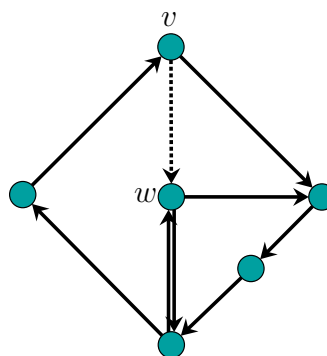
$$c_B \notin \circ \rightarrow \bullet(\mathcal{S}) :$$



$$c_w = 4 \geq 3 = c_v$$

$$c'_w = 4 < 6 = c'_v$$

$$c_B \notin \bullet \rightarrow \circ(\mathcal{S}) :$$



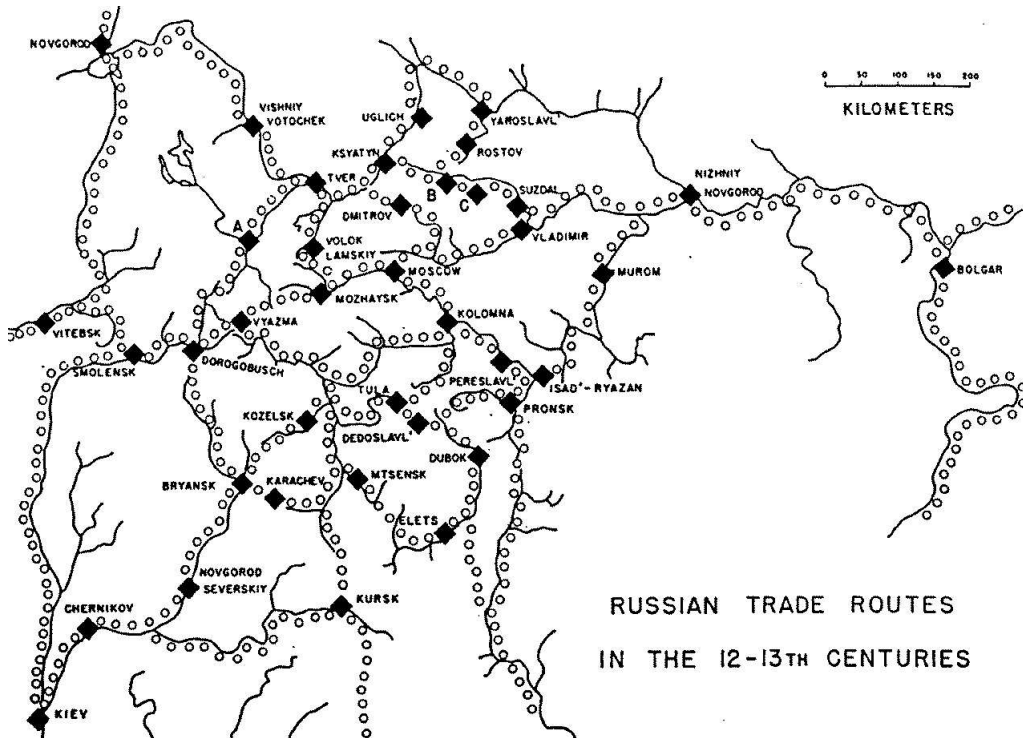
$$c_v = 4 \geq 3 = c_w$$

$$c'_v = 4 < 6 = c'_w$$

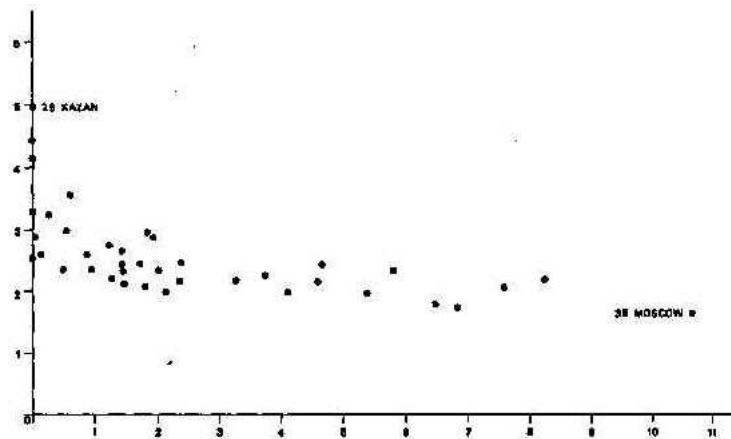
\square

2.16 Beispiel (Der Aufstieg Moskaus, Pitts 1965/1978)

Netzwerk russischer Handelsrouten im 12./13. Jahrhundert:



Betweenness vs. Gesamtabstand der einzelnen Städte:



Um Closeness- und Betweenness Zentralitäten zu berechnen, benötigen wir außer den paarweisen Abständen zwischen Knoten auch die paarweisen Anzahlen der kürzesten Wege und die Anzahlen von kürzesten Wegen über Dritte. Ein naheliegender Ansatz ergibt sich unmittelbar aus der Repräsentation von Multigraphen durch Adjazenzmatrizen.

2.17 Lemma

Sei $G = (V, E)$ ein Multigraph, $A = A(G)$ seine Adjazenzmatrix und $A^k = \left(a_{s,t}^{(k)} \right)_{s,t \in V}$ deren k -te Potenz. Für zwei Knoten $s, t \in V$ ist $a_{s,t}^{(k)}$ gerade die Anzahl aller gerichteten Kantenfolgen von s nach t der Länge k .

■ **Beweis:** Durch Induktion über k : Für $k = 0$ ist $a_{s,t}^{(0)} = 0$ für $s \neq t$ und $a_{s,t}^{(0)} = 1$ andernfalls. Das ist gerade die Anzahl der gesuchten Kantenfolgen der Länge 0.

Jede Kantenfolge von s nach t der Länge $k > 0$ endet mit einer Kante (v, t) für ein $v \in V$. Davon gibt es jeweils $a_{v,t}$ viele und nach Induktionsvoraussetzung ist $a_{s,v}^{(k-1)}$ die Anzahl der gerichteten Kantenfolgen von s nach v der Länge $k - 1$. Für die Einträge von $A^k = A^{k-1} \cdot A$ gilt aber

$$a_{s,t}^{(k)} = \sum_{v \in V} a_{s,v}^{(k-1)} \cdot a_{v,t}$$

und damit die Behauptung. □

Für zwei Knoten $s, t \in V$ eines Multigraphen $G = (V, E)$ mit endlichem Abstand von s nach t und einen weiteren Knoten $s, t \neq v \in V$ sind daher

$$d_G(s, t) = \min \left\{ k \in \mathbb{N}_0 : a_{s,t}^{(k)} \neq 0 \right\} \leq n - 1$$

$$\sigma_G(s, t) = a_{s,t}^{(d_G(s,t))}$$

$$\sigma_G(s, t|v) = \begin{cases} \sigma_G(s, v) \cdot \sigma_G(v, t) & \text{falls } d_G(s, t) = d_G(s, v) + d_G(v, t) \\ 0 & \text{sonst .} \end{cases}$$

Mit der naheliegenden $\mathcal{O}(n^3)$ Implementation der Matrixmultiplikation erhalten wir daraus einen $\mathcal{O}(n^4)$ Algorithmus für die Berechnung der Closeness- und Betweenness-Zentralität.

Wir werden die Laufzeit zunächst auf $\mathcal{O}(n^3)$ und dann auf $\mathcal{O}(nm)$ verbessern. Da für viele soziale und große Netzwerke $m \in \mathcal{O}(n)$ gilt, entspricht das noch einmal einer Größenordnung.

Der folgende Algorithmus ist die Erweiterung eines Standard-Algorithmus' zur Bestimmung der paarweisen Abstände um die Anzahlen der kürzesten Wege.

Algorithmus 5: Längen und Anzahlen kürzester Wege
(Warshall 1962; Floyd 1962; Batagelj 1993)

Eingabe: Multigraph $G = (V, E)$

Ausgabe: Matrix $D = (d_{s,t})_{s,t \in V}$ (paarweise Abstände)

Matrix $\Sigma = (\sigma_{s,t})_{s,t \in V}$ (Anzahlen kürzester Wege)

initialisiere D mit $d_{v,w} = \begin{cases} 0 & v = w \\ 1 & v \neq w \text{ und } (v, w) \in E \\ \infty & \text{sonst} \end{cases}$

initialisiere Σ mit $\sigma_{v,w} = \begin{cases} 1 & v = w \\ k & v \neq w \text{ und } (v, w) \in_k E \\ 0 & \text{sonst} \end{cases}$

foreach $v \in V$ **do**

foreach $s \in V \setminus \{v\}$ **do**

foreach $t \in V \setminus \{v\}$ **do**

if $d_{s,t} = d_{s,v} + d_{v,t}$ **then**

$\sigma_{s,t} \leftarrow \sigma_{s,t} + \sigma_{s,v} \cdot \sigma_{v,t}$

if $d_{s,v} + d_{v,t} < d_{s,t}$ **then**

$d_{s,t} \leftarrow d_{s,v} + d_{v,t}$

$\sigma_{s,t} \leftarrow \sigma_{s,v} \cdot \sigma_{v,t}$

2.18 Satz

Nach Beendigung des Algorithmus gilt $d_{s,t} = d_G(s, t)$ und $\sigma_{s,t} = \sigma_G(s, t)$ für alle $s, t \in V$.

■ **Beweis:** Wir nehmen der Einfachheit halber an, dass die Knoten in jeder Schleife in der festen Reihenfolge v_1, \dots, v_n durchlaufen werden. Für alle $s, t \in V$ seien $d_{s,t}^{(k)}$ und $\sigma_{s,t}^{(k)}$ die nach $0 \leq k \leq n$ Durchläufen der äußersten Schleife berechneten Einträge.

Wir zeigen, dass die $d_{s,t}^{(k)}$ und $\sigma_{s,t}^{(k)}$ gerade die Länge und Anzahl der kürzesten (s, t) -Wege ist, in denen nur Knoten aus $V^{(k)} = \{v_1, \dots, v_k\}$ als innere Knoten vorkommen.

Die Aussage ist für $k = 0$ sicher richtig. Nehmen wir also an, sie gilt nach $k - 1 \geq 0$ Durchläufen der äußersten Schleife, wählen ein festes Paar $s, t \in V$ und betrachten den Knoten v_k . Wenn v_k auf einem kürzesten (s, t) -Weg liegt, dessen innere Knoten alle aus $V^{(k)}$ stammen, dann haben die Teilwege von s nach v_k und von v_k nach t innere Knoten nur aus $V^{(k-1)}$. Die Länge eines solchen Weges ist nach unserer Invariante gerade $d_{s,v_k}^{(k-1)} + d_{v_k,t}^{(k-1)}$. Die $\sigma_{s,v_k}^{(k-1)}$ solcher (s, v_k) - und $\sigma_{v_k,t}^{(k-1)}$ solcher (v_k, t) -Wege können wir beliebig kombinieren. \square

Da die Laufzeit offensichtlich in $\Theta(n^3)$ ist, und wir aus den berechneten Informationen anschließend die Closeness- und Betweenness-Zentralitäten in $\Theta(n^2)$ bzw. in $\Theta(n^3)$ Zeit bestimmen können, erhalten wir insgesamt zwei $\Theta(n^3)$ Algorithmen.

Breitensuche

Zumindest für die Closeness-Zentralität wird die Laufzeit von der Berechnung der paarweisen Abstände dominiert. Statt mit Matrizen zu rechnen werden wir den Multigraphen daher wie bei den verschiedenen Typen von Zusammenhangskomponenten durchlaufen, um ausnutzen zu können, dass die meisten relevanten Netzwerke „wenige“ Kanten haben, d.h. $m \in o(n^2)$.

Anders als bei der Tiefensuche setzen wir die Suche allerdings nicht vom zuletzt, sondern vom zuerst gefundenen Knoten aus fort; suchen also in die Breite statt in die Tiefe. Die wesentliche Änderung in der Implementation besteht daher in der Verwendung einer Queue (FIFO: *first-in, first-out*) anstelle des Stacks (LIFO: *last-in, first-out*).

Außerdem beschränken wir uns auf die Suche ausgehend von einem bestimmten Knoten. Da kein Backtracking stattfindet, wird der vom Argumentknoten aus durchsuchte Teilgraph erst zum Schluss ausgewertet.

Algorithmus 6: (Gerichtete) Breitensuche (*breadth-first search*, BFS)

Eingabe : Multigraph $G = (V, E)$, Wurzel $s \in V$

Daten : Queue Q (für Knoten in BFS-Front)

Knoten- und Kantenmarkierungen

markiere s

$Q \leftarrow (s)$

→ **root**(s)

while Q nicht leer **do**

 entferne $v \leftarrow Q$

foreach nicht markierte $e = (v, w) \in E$ **do**

 markiere e

if w nicht markiert **then**

 markiere w

 füge an $Q \leftarrow w$

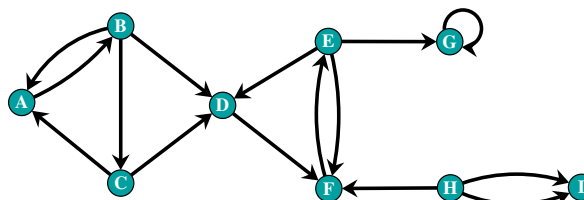
 → **traverse**(v, e, w)

→ **done**(s)

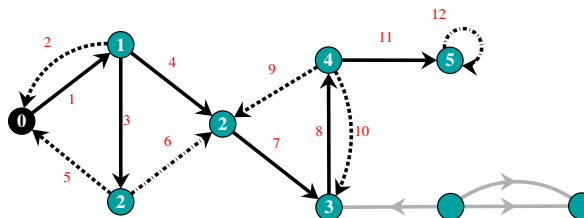
Analog zur Tiefensuche definieren wir eine BFS-Numerierung der Knoten, die jedoch eher der DFS-Numerierung der Kanten gleicht, da sie Knoten mit gleich numerierten Vorgängern nicht unterscheidet. Der Eingabeknoten s erhält die Breitensuchnummer $BFS(s) = 0$, und wir nennen s *Wurzel* der Breitensuche. Wird ein Knoten $w \in V$ beim Durchlaufen einer Kante (v, w) markiert, erhält er die Nummer $BFS(w) = BFS(v) + 1$. Beachte, dass v bei Durchlaufen von (v, w) bereits zuvor markiert wurde und daher numeriert ist. Die Breitensuchnummer aller nicht markierten Knoten sei ∞ .

Die Kanten werden während der Breitensuche wie folgt klassifiziert, wobei die Bedeutung der Nicht-Baumkanten gegenüber der Tiefensuche leicht modifiziert ist. Zum Zeitpunkt, da die Kante (v, w) markiert wird, wird sie zu einer

- Baumkante (\longrightarrow), falls w nicht markiert,
- Rückwärtskante (\dashrightarrow), falls w markiert und $BFS(w) < BFS(v)$,
- Querkante (\dashrightarrow), falls w markiert und $BFS(w) = BFS(v)$ und
- Vorwärtskante (\dashrightarrow), falls w markiert und $BFS(w) > BFS(v)$.



Multigraph



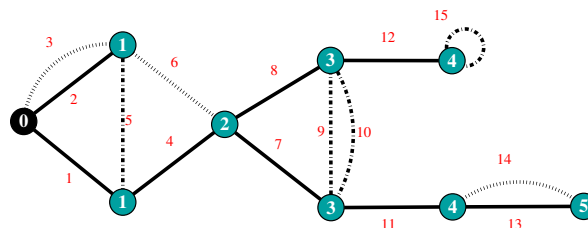
(gerichtete) Breitensuche

Indizes entsprechen Breitensuchnummer der Knoten bzw. Durchlaufreihenfolge der Kanten (graue wurden nicht durchlaufen)

Wieder erhalten wir eine ungerichtete Version des Durchlaufs, indem wir die eingekastete Kantenauswahlbedingung durch

$$e = (v, w) \in E \text{ bzw. } e = (w, v) \in E$$

ersetzen.



ungerichtete Breitensuche
 Indizes entsprechen Breitensuchnummer der Knoten bzw.
 Durchlaufreihenfolge Kanten

Man sieht leicht, dass eine Breitensuche mit $\mathcal{O}(m)$ Laufzeit implementiert werden kann.

2.19 Lemma

Sei $G = (V, E)$ ein Multigraph und $s \in V$. Nach Breitensuche mit Wurzel s gilt $d_G(s, v) = BFS(v)$ für alle $v \in V$.

■ **Beweis:** Wird ein Knoten w an die Queue angehängt, dann gibt es einen Weg von der Wurzel s nach w , und umgekehrt werden alle Knoten, für die es einen solchen Weg gibt, während der Breitensuche einmal angehängt. Insbesondere erhalten genau diese Knoten eine endliche BFS-Nummer.

Wir zeigen zunächst folgende Invariante: Sei $Q = (v_1, \dots, v_k)$ der Zustand der Queue zu irgendeinem Zeitpunkt der Breitensuche, dann gilt $BFS(v_i) \leq BFS(v_{i+1})$ für alle $1 \leq i < k$ und $BFS(v_k) \leq BFS(v_1) + 1$.

Die Invariante gilt natürlich zu Beginn, wenn $Q = (s)$. Die Queue ändert sich entweder durch Anfügen oder Entfernen eines Knotens. Wird ein Knoten w angefügt, so wurde zuvor ein Knoten v entfernt und wir können die Invariante annehmen. Das heißt aber, für v_k (falls es überhaupt existiert), gilt $BFS(v_k) \leq BFS(v) + 1 = BFS(w)$ und die Invariante bleibt erhalten. Der andere Fall (Entfernen eines Knotens) ist noch einfacher.

Sei $V_k = \{v \in V : d_G(s, v) = k\}$ für alle $0 \leq k < n$. Wir zeigen nun mit Hilfe der Invariante, dass $v \in V_k$, falls die Breitensuche v die BFS-Nummer k zuweist. Zusammen mit der Beobachtung, dass alle erreichbaren Knoten eine BFS-Nummer erhalten, beweist das die Behauptung. Für $V_0 = \{s\}$ ist nichts zu zeigen. Für einen Knoten $w \in V_k$, $0 < k < n$, gilt für jeden unmittelbaren Vorgänger v auf einem kürzesten (s, w) -Weg, dass $v \in V_{k-1}$. Zum Zeitpunkt da der erste solche Knoten aus Q entfernt wird, ist w wegen der obigen Invariante noch nicht in Q , hat also keine zu kleine BFS-Nummer erhalten. Die Behauptung folgt, da mit Induktion über k die Nachfolger aller Knoten aus V_{k-1} numeriert werden, bevor ein Knoten aus V_k aus der Queue entfernt wird. \square

Algorithmus 7: Closeness-Zentralität von $s \in V$
(Spezialisierung der Breitensuche mit Wurzel s)

Ausgabe: Zentralität c_s

root(vertex s) begin

| $c_s \leftarrow 0$

end

traverse(vertex v , edge e , vertex w) begin

| **if** e *ist Baumkante* **then** $c_s \leftarrow c_s + BFS(w)$

end

done(vertex s) begin

| **if** $c_s \neq 0$ **then** $c_s \leftarrow \frac{1}{c_s}$

end

2.20 Satz

Die Closeness-Zentralitäten der Knoten eines stark zusammenhängenden Multigraphen können in $\mathcal{O}(nm)$ Zeit berechnet werden.

■ **Beweis:** Breitensuche von jedem Knoten aus. □

Wir zeigen als nächstes, dass auch Betweenness-Zentralität durch Breitensuche von jedem Knoten aus berechnet werden kann. Dazu müssen wir zunächst zeigen, wie nicht nur die Länge, sondern auch die Anzahl der kürzesten Wege von der Wurzel aus bestimmt werden kann. Sei dazu

$$P_G^-(s, v) = \{(u, v) \in E : d_G(s, v) = d_G(s, u) + 1\}$$

die Multimenge aller eingehenden Kanten von v auf kürzesten (s, v) -Wegen (jeweils mit gleicher Vielfachheit wie in E).

2.21 Lemma

$$\sigma_G(s, v) = \begin{cases} \sum_{(u,v) \in P_G^-(s,v)} \sigma_G(s, u) & \text{falls } s \neq v^1 \\ 1 & \text{sonst.} \end{cases}$$

■ **Beweis:** Falls $s \neq v$ gibt es auf jedem kürzesten (s, v) -Weg einen eindeutigen Vorgänger u . Jeder kürzeste (s, u) -Weg kann dann durch jede Kante (u, v) zu einem kürzesten (s, v) -Weg verlängert werden. □

¹Beachte zur Schreibweise: Für jede Kante aus $P_G^-(s, v)$ werden entsprechend ihrer Vielfachheit viele Summanden addiert.

Da wir bei der Breitensuche immer nur einen Bezugspunkt (die Wurzel) haben, schreiben wir die Definition der Betweenness-Zentralität um. Dazu definieren wir die Abhängigkeit des Paares $s, t \in V$ bzw. des Knotens $s \in V$ vom Knoten $v \in V$ als

$$\delta_G(s, t|v) = \frac{\sigma_G(s, t|v)}{\sigma_G(s, t)}$$

$$\delta_G(s|v) = \sum_{t \in V} \delta_G(s, t|v) .$$

Die Betweenness-Zentralität des Knotens $v \in V$ ist dann die Summe der einseitigen Abhängigkeiten aller anderen Knoten von v ,

$$c_B(G)_v = \sum_{s \in V} \delta_G(s|v),$$

und wir haben folgende Rekursionsgleichung.

2.22 Lemma

Für $s \neq v \in V$

$$\delta_G(s|v) = \sum_{\substack{(v,w) \in P_G^-(s,w) \\ \text{für ein } w \in V}} \frac{\sigma_G(s, v)}{\sigma_G(s, w)} \cdot (1 + \delta_G(s|w)) .$$

■ **Beweis:** Wir erweitern den Begriff der Abhängigkeit um benötigte Kanten zu

$$\delta_G(s, t|v, e) = \frac{\sigma_G(s, t|v, e)}{\sigma_G(s, t)}$$

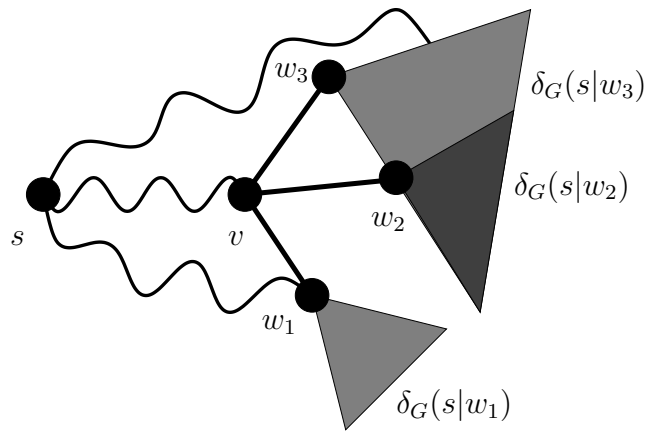
wobei $\sigma_G(s, t|v, e)$ die Anzahl der kürzesten (s, t) -Wege sei, die sowohl den inneren Knoten v als auch die Kante e enthalten. Wir haben dann

$$\begin{aligned} \delta_G(s|v) &= \sum_{t \in V} \delta_G(s, t|v) \\ &= \sum_{t \in V} \sum_{\substack{(v,w) \in P_G^-(s,w) \\ \text{für ein } w \in V}} \delta_G(s, t|v, (v, w)) \\ &= \sum_{\substack{(v,w) \in P_G^-(s,w) \\ \text{für ein } w \in V}} \sum_{t \in V} \delta_G(s, t|v, (v, w)) . \end{aligned}$$

Zu $v \in V$ sei $w \in V$ ein Knoten mit $(v, w) \in P_G^-(s, w)$. Von den $\sigma_G(s, w)$ kürzesten (s, w) -Wegen enthalten $\sigma_G(s, v)$ viele kürzeste (s, v) -Wege, die dann die Kante (v, w) benutzen, sodass $\frac{\sigma_G(s, v)}{\sigma_G(s, w)} \cdot \sigma_G(s, t|w)$ kürzeste Wege von s nach $t \neq w$ sowohl v als auch (v, w) benutzen.

Die Abhängigkeit des Paares s, t von v und (v, w) ist daher

$$\delta_G(s, t|v, (v, w)) = \begin{cases} \frac{\sigma_G(s, v)}{\sigma_G(s, w)} & \text{falls } t = w \\ \frac{\sigma_G(s, v)}{\sigma_G(s, w)} \cdot \frac{\sigma_G(s, t|w)}{\sigma_G(s, t)} & \text{falls } t \neq w \end{cases}$$



Durch Einsetzen erhalten wir

$$\begin{aligned} \delta_G(s|v) &= \sum_{t \in V} \delta_G(s, t|v) \\ &= \sum_{\substack{(v, w) \in P_G^-(s, w) \\ \text{für ein } w \in V}} \sum_{t \in V} \delta_G(s, t|v, (v, w)) \\ &= \sum_{\substack{(v, w) \in P_G^-(s, w) \\ \text{für ein } w \in V}} \left(\frac{\sigma_G(s, v)}{\sigma_G(s, w)} + \sum_{t \in V \setminus \{w\}} \frac{\sigma_G(s, v)}{\sigma_G(s, w)} \cdot \frac{\sigma_G(s, t|w)}{\sigma_G(s, t)} \right) \\ &= \sum_{\substack{(v, w) \in P_G^-(s, w) \\ \text{für ein } w \in V}} \frac{\sigma_G(s, v)}{\sigma_G(s, w)} \cdot (1 + \delta_G(s|w)) . \end{aligned}$$

□

Die folgende Spezialisierung der Breitensuche berechnet die Abhängigkeiten der Wurzel von allen anderen Knoten, indem nach Ende der Breitensuche für alle Knoten in umgekehrter Reihenfolge die Abhängigkeiten mittels der Rekursionsgleichung bestimmt werden. Dazu reicht es, für einen Knoten $w \in V$ anstelle der Kanten aus $P_G^-(w)$ nur deren Anfangsknoten (mit gleicher Vielfachheit) zu speichern.

Algorithmus 8: Abhängigkeiten eines Knotens $s \in V$ von allen anderen

(Spezialisierung der Breitensuche mit Wurzel s)

Daten : Knotenarray σ_s (Anzahl der kürzesten Wege von s)
 Knotenarray P_s (Liste der Vorgänger auf kürzesten Wegen)
 Stack S (Knoten in Reihenfolge ihres Abstands von s)
Ausgabe: Knotenarray δ_s (Abhängigkeiten, initialisiert mit 0)

root(vertex s) begin

 | $\sigma_s(s) = 1$

end

traverse(vertex v , edge e , vertex w) begin

 | **if e ist Baumkante then**

 | füge an $P_s(w) \leftarrow v$

 | $\sigma_s(w) \leftarrow \sigma_s(v)$

 | push $w \rightarrow S$

 | **if e ist Vorwärtskante then**

 | füge an $P_s(w) \leftarrow v$

 | $\sigma_s(w) \leftarrow \sigma_s(w) + \sigma_s(v)$

end

done(vertex s) begin

 | **while S nicht leer do**

 | $w \leftarrow pop(S)$

 | **foreach $v \in P_s(w)$ do**

 | $\delta_s(v) \leftarrow \delta_s(v) + \frac{\sigma_s(v)}{\sigma_s(w)} \cdot (1 + \delta_s(w))$

end

2.23 Satz

Die Betweenness-Zentralitäten der Knoten eines Multigraphen können in $\mathcal{O}(nm)$ Zeit berechnet werden.

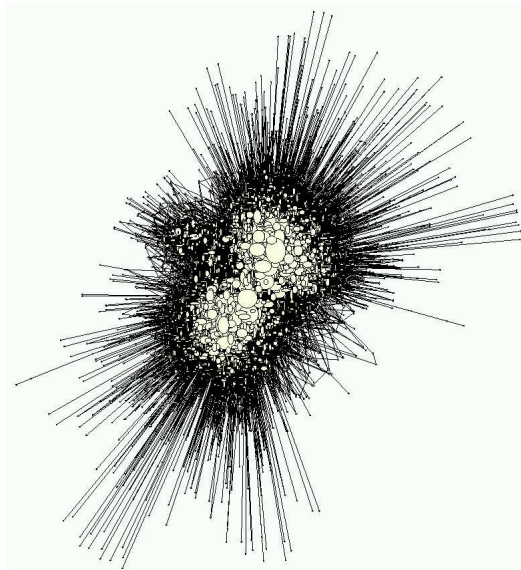
■ **Beweis:** Summiere für alle Knoten deren Abhängigkeiten von jeweils allen anderen. □

2.24 Bemerkung

Die beiden breitensuchbasierten Algorithmen lassen sich so implementieren, dass der Speicherplatzbedarf linear in der Größe des Multigraphen ist, und sind auch gut parallelisierbar.

2.25 Beispiel (Needle-Exchange Network, Valente und Foreman 2000)

Studie in Baltimore: Analyse eines Multigraphen, dessen Knoten Drogenabhängige repräsentieren, die an einem Spritzenumtauschprogramm teilnehmen. Darin können gebrauchte Spritzen gegen saubere eingetauscht werden. Die Spritzen sind markiert, und bringt ein Abhängiger eine Spritze zurück, die ein anderer entgegengenommen hat, induziert dies eine Kante zwischen den beiden, welche die Übergabe repräsentiert.



4259 Knoten, 61693 Kanten (sehr dünn: 0,3% aller möglichen)

Auf einem Standard-PC (Stand 2000, ca. 300 MHz) hätte Betweenness-Zentralität mit kubischer Laufzeit ca. 2 Tage benötigt (die dabei verwendeten Matrizen passten wegen des quadratischen Platzbedarfs aber gar nicht erst in den Speicher). Mit linearem Speicherbedarf und $\mathcal{O}(nm)$ Algorithmus dauerte die Berechnung weniger als 10 Minuten.

Eine Laufzeit von $\mathcal{O}(nm)$ kann für sehr große Netzwerke, selbst wenn sie dünn sind, immer noch zu lang sein. Wir betrachten daher abschließend zwei Spezialfälle, nämlich eine Methode zur näherungsweise Berechnung von Closeness-Zentralität in Netzwerken mit kleinem Durchmesser und Linearzeitalgorithmen für beide Zentralitäten in dem Fall, dass die Eingabe ein ungerichteter Baum ist.

Näherungsweise Berechnung von Closeness-Zentralität

Ein wesentlicher Unterschied in den beiden Verfahren zur Berechnung von Zentralitäten besteht darin, dass nach einer einzelnen Breitensuche bei Closeness-Zentralität der *Zentralitätswert* der Wurzel bekannt ist, wohingegen bei der Betweenness-Zentralität nur der *Beitrag* der Wurzel zu allen anderen Zentralitätswerten ermittelt wird.

Ein weiterer Unterschied besteht darin, dass eine für manche Arten von Netzwerken typische Eigenschaft, nämlich ein beschränkter Durchmesser („kleine Welt“), bei Closeness-Zentralität für eine Beschränkung der auftretenden Größen sorgt, wohingegen bei Betweenness der tatsächliche Abstand zweier Knoten weitgehend irrelevant ist.

Wir nutzen diese Tatsache aus, um Closeness-Zentralität mit weniger Aufwand angenähert zu berechnen. Damit sich der gemachte Fehler unabhängig von der Größe des Netzwerks abschätzen lässt, betrachten wir statt des Kehrwerts des Gesamtabstands den (gleichwertigen) Kehrwert des mittleren Abstands zu allen anderen Knoten, d.h. wir betrachten für einen Knoten $v \in V$ die standardisierte Closeness-Zentralität

$$\bar{c}_C(G)_v = \frac{n-1}{\sum_{t \in V} d_G(v, t)},$$

deren Werte alle im Intervall $[0, 1]$ liegen.

Sei im folgenden $G = (V, E) \in \mathcal{S}$ ein stark zusammenhängender Multigraph mit kleinem Durchmesser, d.h. $\text{diam}(G) \leq D$ für eine Konstante D . Weil die auftretenden paarweisen Abstände dadurch auf ein konstant großes Intervall beschränkt sind, gilt bei zufälliger Auswahl von k Knoten $t_1, \dots, t_k \in V$ für

genügend großes k

$$\frac{\sum_{t \in V} d_G(v, t)}{n} \approx \frac{\sum_{i=1}^k d_G(v, t_i)}{k}$$

und damit

$$\bar{c}_C(G)_v \approx \frac{n-1}{n \cdot \frac{\sum_{i=1}^k d_G(v, t_i)}{k}} = \frac{(n-1) \cdot k}{n \cdot \sum_{i=1}^k d_G(v, t_i)}.$$

Wählen wir also Knoten $t_1, \dots, t_k \in V$ zufällig und unabhängig voneinander, dann können zunächst die Abstände $0 \leq d_G(v, t_i) \leq D$ für alle $v \in V$ durch k Breitensuchen *entgegen* der Kantenrichtungen in G bestimmt und aufsummiert werden. Der so erhaltene Wert kann dann einfach in die Näherungsformel eingesetzt werden.

Daraus ergibt sich der folgende Algorithmus, in dem wir noch offen lassen, wieviele Referenzknoten gewählt werden sollen.

Algorithmus 9: Closeness-Zentralität in sehr großen Graphen

Eingabe: stark zusammenhängender Multigraph $G = (V, E)$

Wiederholungszahl k

Ausgabe: standardisierte Closeness c

(näherungsweise; initialisiert mit 0)

repeat k mal **this**

 wähle zufälliges $t \in V$
 umgekehrt gerichtete Breitensuche mit Wurzel t
 foreach $v \in V$ **do** $c(v) \leftarrow c(v) + BFS(v)$

foreach $v \in V$ **do** $c(v) \leftarrow \frac{(n-1) \cdot k}{n \cdot c(v)}$

In der Analyse des Algorithmus verwenden wir für die Fehlerabschätzung die folgende Aussage über die Abweichung des Mittelwertes einer Stichprobe vom erwarteten Mittelwert. Der Beweis kann mit elementaren Mitteln erfolgen, ist hier aber nicht von Interesse.

2.26 Lemma (Hoeffding 1963)

Sind X_1, \dots, X_k unabhängige Zufallsvariablen mit $0 \leq X_i \leq M$ für alle $i = 1, \dots, k$, dann gilt

$$P\left(\left|\frac{X_1 + \dots + X_k}{k} - E\left(\frac{X_1 + \dots + X_k}{k}\right)\right| \geq \xi\right) \leq e^{-2k\left(\frac{\xi}{M}\right)^2}.$$

2.27 Satz (Eppstein und Wang 2001)

Sei D eine Konstante und $G = (V, E)$ ein stark zusammenhängender Multigraph mit $\text{diam}(G) \leq D$. Dann können in Zeit $\mathcal{O}\left(\frac{1}{\varepsilon^2} \cdot m \log n\right)$ Werte $(c_v)_{v \in V}$ so berechnet werden, dass für alle $v \in V$

$$\left|\frac{1}{c_v} - \frac{1}{\bar{c}_C(G)_v}\right| \leq \varepsilon D$$

mit großer Wahrscheinlichkeit.

■ **Beweis:** Wir benutzen den obigen Algorithmus und wenden das Lemma auf die Kehrwerte der berechneten Zentralitäten an. Dazu setzen wir

$$\begin{aligned} M &= \frac{n}{n-1} \cdot D \\ X_i &= \frac{n}{n-1} \cdot d_G(v, t_i) \quad (0 \leq X_i \leq M) \\ \xi &= \varepsilon D \end{aligned}$$

und zeigen zunächst, dass der Erwartungswert der gemittelten Summe gerade der Kehrwert der tatsächlichen Zentralität ist:

$$\begin{aligned} \sum_{t_1 \in V} \dots \sum_{t_k \in V} \frac{1}{n^k} \cdot \frac{\sum_{i=1}^k \frac{n}{n-1} \cdot d_G(v, t_i)}{k} &= \frac{1}{n^k} \cdot \frac{n}{(n-1) \cdot k} \cdot \sum_{t \in V} k \cdot n^{k-1} \cdot d_G(v, t) \\ &= \frac{\sum_{t \in V} d_G(v, t)}{n-1} \\ &= \frac{1}{\bar{c}_C(G)_v}. \end{aligned}$$

Wählen wir außerdem

$$k = \left\lceil \frac{2 \log n}{\varepsilon^2} \right\rceil ,$$

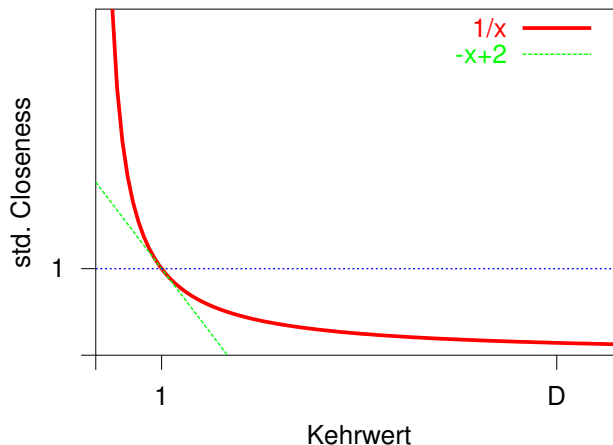
dann ist für ein festes $v \in V$ die Wahrscheinlichkeit einer Abweichung um mindestens εD höchstens

$$\begin{aligned} P \left(\left| \frac{n \cdot \sum_{i=1}^k d_G(v, t_i)}{(n-1) \cdot k} - \frac{1}{\bar{c}_C(G)_v} \right| \geq \varepsilon D \right) &\leq e^{-2 \lceil \frac{2 \log n}{\varepsilon^2} \rceil \left(\frac{\varepsilon D}{\frac{n}{n-1} \cdot D} \right)^2} \\ &\leq e^{-2 \log n} = \frac{1}{n^2} . \end{aligned}$$

Die Wahrscheinlichkeit, dass ein solcher Fehler an irgendeinem Knoten auftritt, ist damit höchstens $\frac{1}{n}$. \square

Frage: Ist das gut?

Der Satz macht eine Aussage über die Kehrwerte der Zentralitäten. Bei Durchmesser höchstens D liegt der Kehrwert der standardisierten Closeness-Zentralität zwischen 1 und D .



Für Knoten mit großer Zentralität wirkt sich eine Abweichung vom Kehrwert also stärker aus, als für solche mit kleiner Zentralität, der Fehler in der Zentralität ist aber kleiner als der im Kehrwert.

Beachte auch, dass wir im Anschluss (mit noch einmal der gleichen Laufzeit) für die $k = \lceil \frac{2 \log n}{\varepsilon^2} \rceil$ Knoten mit größtem Schätzwert die Zentralität exakt bestimmen könnten.

Frage: Funktioniert das auch mit Betweenness?

Zentralitäten auf ungerichteten Bäumen

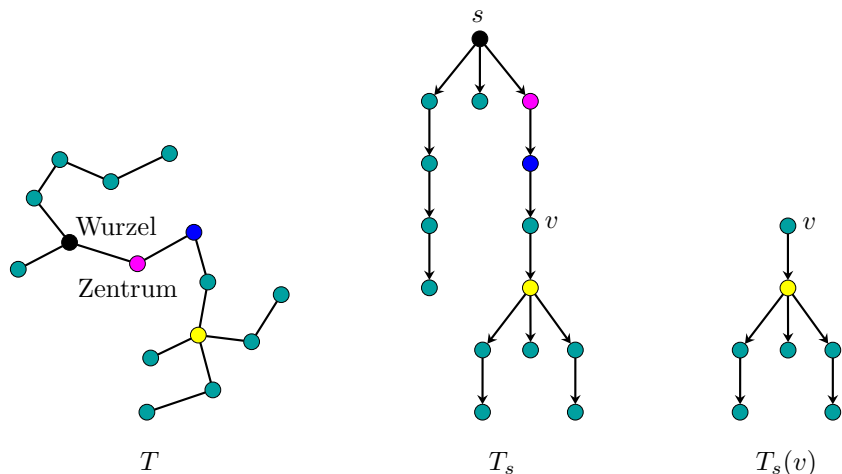
Ein ungerichteter Baum ist ein zusammenhängender schlichter ungerichteter Graph ohne einfache Kreise. Da die bisher betrachteten Zentralitäten auf kürzesten Wegen basieren, legt die folgende Eigenschaft nahe, dass solche Zentralitäten auf Bäumen leichter zu berechnen sind.

2.28 Lemma

Zwischen je zwei Knoten eines ungerichteten Baumes gibt es genau einen einfachen Weg.

■ **Beweis:** Da der Graph zusammenhängend ist, gibt es mindestens einen Weg. Gäbe es zwischen irgendzwei Knoten zwei verschiedene einfache Wege, so enthielte deren Vereinigung einen einfachen Kreis. \square

Um die Zentralitäten aller Knoten eines ungerichteten Baumes zu berechnen, durchlaufen wir zu einem ungerichteten Baum T einen zugehörigen Wurzelbaum T_s , d.h. einen gerichteten Graphen, den wir dadurch erhalten, dass wir einen Knoten s als Wurzel auszeichnen und alle Kanten von der Wurzel weg orientieren. Der Teilgraph aller von einem Knoten v in T_s aus erreichbaren Knoten heißt *Wurzelteilbaum* $T_s(v)$.



Durch das obige Lemma wissen wir, dass alle Wege zwischen einem Knoten aus einem Wurzelteilbaum und einem Knoten im Rest des Graphen die Teilbaumwurzel v enthalten müssen (es gibt ja nur genau einen). Wir sammeln daher die für die Zentralitäten benötigten Informationen (Abstandssumme, Knotenzahlen) inner- und außerhalb der Wurzelteilbäume und kombinieren sie dann.

Der gesamte Wurzelbaum wird dazu zweimal durchlaufen, wobei wir beim ersten Mal Informationen „von unten nach oben“ (*bottom-up*) und beim zweiten Mal „von oben nach unten“ (*top-down*) weiterreichen. Das Weiterreichen geschieht jeweils in Abhängigkeit von der zu berechnenden Zentralität.

Algorithmus 10: Zentralitätsschema für ungerichtete Bäume

Eingabe : ungerichteter Baum $T = (V, E)$

Ausgabe: Knotenarray n^+ (Knotenzahl im Wurzelteilbaum)

bottom_up(vertex v) begin

```

|  $n_v^+ \leftarrow 1$ 
| foreach  $w \in N_{T_s}^+(v)$  do
|   | bottom_up( $w$ )
|   |  $n_v^+ \leftarrow n_v^+ + n_w^+$ 
|   |  $\longrightarrow$  uplabel( $v, w$ )
|

```

end

top_down(vertex v) begin

```

|  $\longrightarrow$  downlabel( $v$ )
| foreach  $w \in N_{T_s}^+(v)$  do
|   | top_down( $w$ )
|

```

end

begin

```

| wähle ein  $s \in V$ 
| bottom_up( $s$ )
| top_down( $s$ )
|

```

end

2.29 Lemma

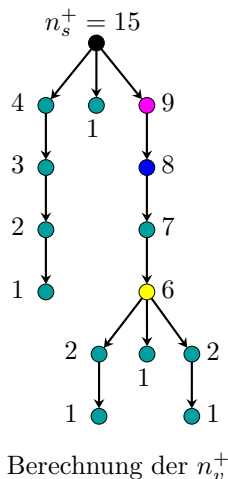
Nach Ausführung des Algorithmus ist n_v^+ die Zahl der Knoten im Wurzelteilbaum $T_s(v)$.

■ **Beweis:** Wir beweisen die Behauptung durch Induktion über die Tiefe der Rekursion im ersten Durchlauf (im zweiten ändern sich die Werte nicht mehr). Hat ein Knoten $v \in V$ keine Nachfolger, erhält er korrekterweise den Wert $n_v^+ = 1$. Andernfalls sind die Werte der Nachfolger nach Induktionsvoraussetzung korrekt berechnet. Da die Wurzelteilbäume der Nachfolger

disjunkt sind, ist

$$n_v^+ = 1 + \sum_{w \in N_{T_s}^+(v)} n_w^+$$

die gewünschte Anzahl. □



2.30 Lemma

Sei $T = (V, E)$ ein ungerichteter Baum und $T_s = (V_s, E_s)$ der zugehörige Wurzelbaum mit Wurzel $s \in V$. Ist $T_s(v) = (V_s(v), E_s(v))$ der Wurzelteilbaum von $v \in V$, dann gilt

$$\sum_{t \in V} d_T(s, t) = \sum_{t \in V} d_{T_s}(s, t) = \sum_{v \in N_{T_s}^+(s)} \sum_{t \in V_s(v)} (1 + d_{T_s}(v, t)) .$$

■ **Beweis:** Die erste Gleichheit gilt, weil die gerichteten Wege von der Wurzel weg genau den einzigen Wegen im ungerichteten Baum entsprechen. Die zweite gilt, weil der Abstand von der Wurzel zu einem Knoten gerade um eins größer ist, als der Abstand von deren Nachfolger auf dem eindeutigen Weg. □

Algorithmus 11: Closeness-Zentralität in ungerichteten Bäumen
(Spezialisierung des Zentralitätsschemas für Bäume)

Daten : Knotenarray d (Abstandssumme, initialisiert mit 0)

Ausgabe: Knotenarray c (Closeness-Zentralität)

uplabel(vertex v , vertex w) begin

$d_v \leftarrow d_v + d_w + n_w^+$

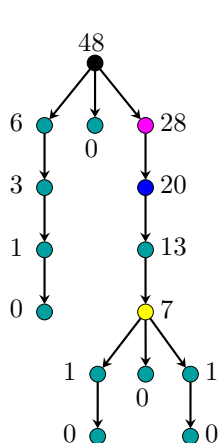
end

downlabel(vertex v) begin

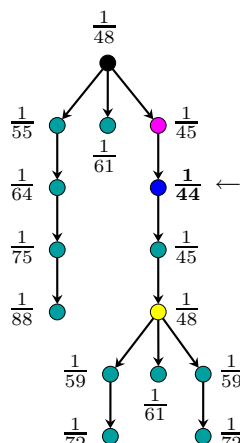
if $N_{T_s}^-(v) = \{u\}$ **then** $d_v \leftarrow d_u + n - 2 \cdot n_v^+$

$c_v \leftarrow \frac{1}{d_v}$

end



Ergebnis von *uplabel*



Ergebnis von *downlabel*

2.31 Satz

Die Closeness-Zentralitäten der Knoten eines ungerichteten Baumes können in $\mathcal{O}(n)$ Zeit berechnet werden.

■ **Beweis:** Wir zeigen, dass der obige Algorithmus korrekt ist. Mit Induktion über die Rekursionstiefe und dem Lemma folgt, dass im ersten Durchlauf für jeden Knoten die Abstände zu allen Knoten seines Wurzelteilbaums summiert werden.

Für die Wurzel sind damit alle Abstände korrekt aufsummiert. Für alle anderen Knoten v sind die Abstände zu den übrigen gerade um eins kleiner als vom Vorgänger im Wurzelbaum, wenn der Zielknoten im eigenen Wurzelteilbaum liegt, und sonst um eins größer. Die Differenz beträgt also $-n_v^+ + (n - n_v^+) = n - 2 \cdot n_v^+$. Die Behauptung folgt mit Induktion über den Abstand von der Wurzel. □

2.32 Lemma

Sei $T = (V, E)$ ein ungerichteter Baum und $T_s = (V_s, E_s)$ der zugehörige Wurzelbaum mit Wurzel $s \in V$. Ist $T_s(v) = (V_s(v), E_s(v))$ der Wurzelteilbaum von $v \in V$, dann gilt

$$c_B(T)_v = (n - n_v^+) \cdot (n_v^+ - 1) + \sum_{w \in N_{T_s}^+(v)} n_w^+ \cdot (n - n_w^+ - 1) .$$

■ **Beweis:** Wir zählen die Wege mit v als innerem Knoten. Ein solcher Weg beginnt entweder außerhalb des Wurzelteilbaums von v , oder innerhalb des Wurzelteilbaums eines Nachfolgers w von v . Im ersten Fall gibt es $n - n_v^+$ mögliche Anfangsknoten, zu denen jeder der $n_v^+ - 1$ Knoten im Wurzelteilbaum von v (ohne v selbst) als Endknoten in Frage kommt. Im zweiten Fall gibt es n_w^+ mögliche Anfangsknoten, zu denen jeder der $n - n_w^+ - 1$ Knoten außerhalb des Wurzelteilbaums von w und ungleich v als Endknoten in Frage kommt. \square

Algorithmus 12: Betweenness-Zentralität in ungerichteten Bäumen
(Spezialisierung des Zentralitätsschemas für Bäume)

Ausgabe: Knotenarray c (Betweenness-Zentralität, initialisiert mit 0)

uplabel(vertex v , vertex w) begin

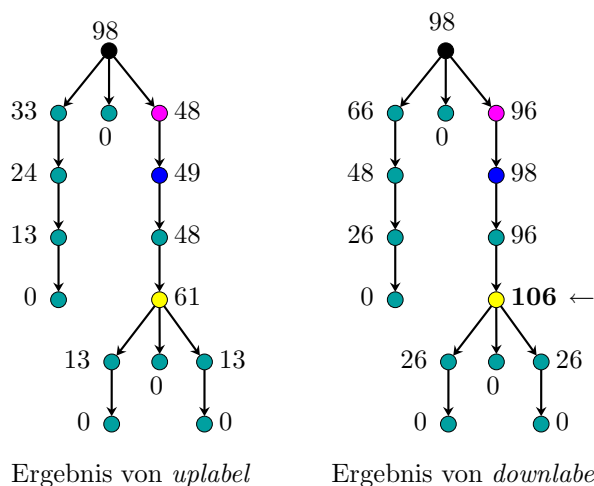
 | $c_v \leftarrow c_v + n_w^+ \cdot (n - n_w^+ - 1)$

end

downlabel(vertex v) begin

 | $c_v \leftarrow c_v + (n - n_v^+) \cdot (n_v^+ - 1)$

end



2.33 Satz

Die Betweenness-Zentralitäten der Knoten eines ungerichteten Baumes können in $\mathcal{O}(n)$ Zeit berechnet werden.

■ **Beweis:** Wir zeigen, dass der obige Algorithmus korrekt ist. Mit Induktion über die Rekursionstiefe und dem Lemma folgt, dass im ersten Durchlauf für jeden Knoten die Abhängigkeiten der Paare mit Anfangsknoten im Wurzelteilbaum aufsummiert werden. Da es nur eine Kante zum Vorgänger gibt, müssen im zweiten Durchlauf nur noch die Paare mit Anfangsknoten außerhalb des Wurzelteilbaums hinzugezählt werden. □

2.34 Bemerkung

Für große Multigraphen mit vielen zweifachen Zusammenhangskomponenten können die Baumalgorithmen mit den Spezialisierungen der Breitensuche kombiniert auf den Block-Schnittknoten-Baum angewandt werden. Der Teufel steckt wie üblich im Detail.

2.35 Bemerkung

Es gibt einige weitere Beispiele für Zentralitäten, die über kürzeste Wege definiert werden. Alle diese Indizes können durch Spezialisierung der Breitensuche bzw. des Baumschemas mit den jeweiligen Laufzeiten berechnet werden.

2.36 Bemerkung

Zentralitäten, die auf kürzesten Wegen basieren, können auf natürliche Weise auf Multigraphen mit positiven Kantengewichten (deren Summe die Länge

eines Weges angibt) verallgemeinert werden. Die verallgemeinerten Indizes können ähnlich berechnet werden, wenn man die Breitensuche durch eine ähnlich schematisierte Version von Dijkstras Algorithmus ersetzt. Bei effizienter Implementation führt das zu einer Laufzeit von $\mathcal{O}(nm + n^2 \log n)$. Auf Bäumen ist die Anpassung noch einfacher und sogar die Linearzeit bleibt erhalten.

2.2 Rückkopplungszentralitäten

Wir haben Knotengrade als lokales Zentralitätsmaß kennengelernt und nicht-lokale Zentralitätsmaße basierend auf Abständen und kürzesten Wegen behandelt. In diesem Abschnitt betrachten wir Zentralitätsmaße, die alle Wege (genauer: Kantenfolgen) berücksichtigen.

Als Verallgemeinerung des Ausgangsgrades haben wir bei der Closeness-Zentralität eines Knotens alle anderen Knoten berücksichtigt, aber einen Verlust an Einfluss bei wachsendem Abstand angenommen. Durch die Kehrwertbildung wurden diejenigen Knoten besonders zentral, deren Einfluss auf andere am wenigsten abgeschwächt wurde.

Eine andere Möglichkeit derselben grundlegenden Idee besteht darin, den ausgeübten Einfluss mit jeder verwendeten Kante um z.B. einen konstanten Faktor $0 < \alpha < 1$ abzuschwächen, und dann den Einfluss eines Knotens auf einen anderen über alle Kantenfolgen zu diesem anderen zu summieren. Mit dem Lemma über Adjazenzmatrizen $A = A(G)$ und die Anzahlen von Kantenfolgen sind die Zentralitäten dann gerade die Zeilensummen der Einflussmatrix

$$A_\infty = \sum_{k=1}^{\infty} (\alpha \cdot A)^k .$$

Allerdings konvergiert diese Reihe nur für hinreichend kleines α . Wir geben eine solche Abschwächung an.

2.37 Lemma

Die Einflussmatrix ist wohldefiniert, falls $\alpha = \frac{1}{\min \{ \Delta^-(G), \Delta^+(G) \} + 1}$.

■ **Beweis:** Wir zeigen durch Induktion über k , dass die Folgen aller Einträge der Summanden $(\alpha \cdot A)^k$ jeweils von der Folge

$$\min \left\{ \left(\frac{\Delta^-}{\Delta^- + 1} \right)^k, \left(\frac{\Delta^+}{\Delta^+ + 1} \right)^k \right\}$$

majorisiert wird. Dazu reicht es aus, für alle Einträge $a_{s,t}^{(k)}$ von A^k zu zeigen, dass $a_{s,t}^{(k)} \leq \min \{ \Delta^-(G)^k, \Delta^+(G)^k \}$. Aus der Konvergenz von $\sum_{k=1}^{\infty} q^k$ für $q \in (0, 1)$ folgt dann die Behauptung.

Die Einträge von A sind höchstens kleiner als $\Delta^- = \max_{v \in V} \sum_{u \in N^-(v)} a_{u,v}$ und $\Delta^+ = \max_{v \in V} \sum_{w \in N^+(v)} a_{v,w}$. Für beliebige $s, t \in V$ betrachte daher den Eintrag $a_{s,t}^{(k)}$ von A^k mit $k > 1$. Wegen $A^k = A^{k-1} \cdot A$ gilt

$$\begin{aligned} a_{s,t}^{(k)} &= \sum_{v \in V} a_{s,v}^{(k-1)} \cdot a_{v,t} = \sum_{(v,t) \in E} a_{s,v}^{(k-1)} \\ &\leq \Delta^- \cdot \max_{v \in N^-(t)} a_{s,v}^{(k-1)} \end{aligned}$$

und wegen $A^k = A \cdot A^{k-1}$ analog

$$\begin{aligned} a_{s,t}^{(k)} &= \sum_{v \in V} a_{s,v} \cdot a_{v,t}^{(k-1)} = \sum_{(s,v) \in E} a_{v,t}^{(k-1)} \\ &\leq \Delta^+ \cdot \max_{v \in N^-(t)} a_{v,t}^{(k-1)} \end{aligned}$$

Nach Induktionsvoraussetzung ist aber

$$a_{u,w}^{(k-1)} \leq \min \left\{ (\Delta^-)^{k-1}, (\Delta^+)^{k-1} \right\}$$

für alle $u, w \in V$. □

2.38 Definition (Einfluss; Katz 1953)

Die Einfluss-Zentralität c_I ist definiert durch

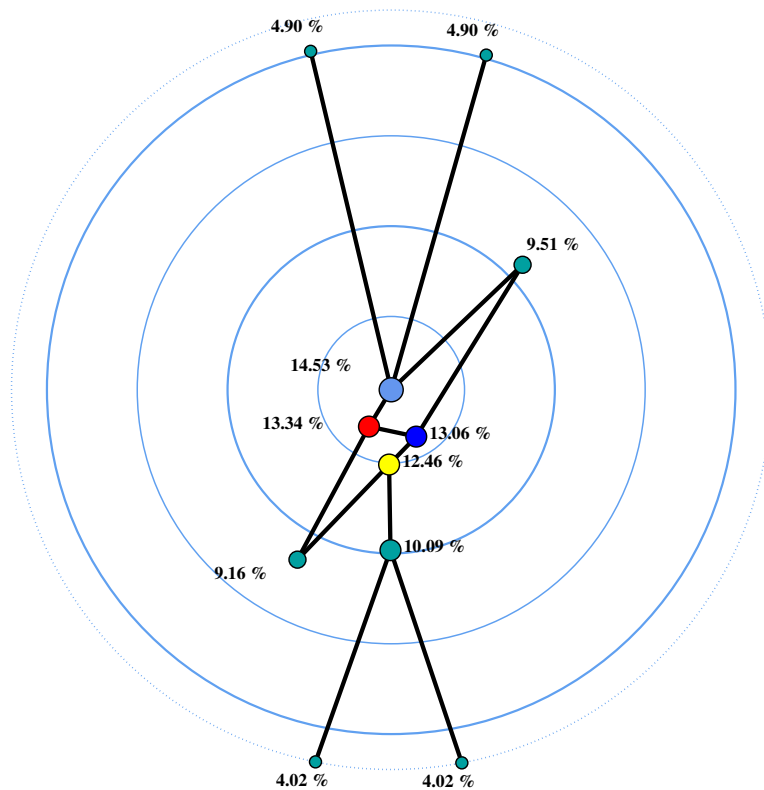
$$c_I(G) = \left(\sum_{k=1}^{\infty} (\alpha \cdot A)^k \right) \cdot \mathbf{1}$$

für alle $G \in \mathcal{G}$ mit Adjazenzmatrix $A = A(G)$ und Abschwächung $\alpha = \alpha(G) = \frac{1}{\min\{\Delta^-(G), \Delta^+(G)\} + 1}$.

2.39 Bemerkung

In der Originalarbeit von Katz (1953) werden nicht Kantenfolgen zu, sondern von anderen Knoten bewertet. Der Zentralitätsindex heißt dort entsprechend auch nicht Einfluss, sondern Status.

Vermutung: $c_I \in \bullet \rightarrow \circ(\mathcal{G}) \cup \bullet \rightarrow \bullet(\mathcal{G})$



normierte Einfluss-Zentralität im Beispielgraphen

Die unendliche Reihe ist für die tatsächliche Berechnung unpraktisch. Aufgrund der folgenden Gleichgewichtsbeziehungen können wir die Einfluss-Zentralitäten allerdings auch durch Lösen eines linearen Gleichungssystems bestimmen. Man beachte die auffällige formale Ähnlichkeit mit der Rekursionsformel für Abhängigkeiten (s. Betweenness-Zentralität).

2.40 Lemma

$$c_I(G)_v = \sum_{(v,w) \in E} \alpha \cdot (1 + c_I(G)_w)$$

■ **Beweis:** Zunächst gilt

$$\begin{aligned} A_\infty &= \sum_{k=1}^{\infty} (\alpha A)^k = (\alpha A) \cdot \sum_{k=0}^{\infty} (\alpha A)^k \\ &= (\alpha A) \cdot \left(I + \sum_{k=1}^{\infty} (\alpha A)^k \right) \\ &= (\alpha A) \cdot (I + A_\infty) \end{aligned}$$

und daher

$$\begin{aligned} c_I(G) &= A_\infty \cdot \mathbf{1} = (\alpha A) \cdot (I + A_\infty) \cdot \mathbf{1} \\ &= (\alpha A) \cdot (\mathbf{1} + c_I(G)) . \end{aligned}$$

□

Nebenbei können wir damit auch formal ausdrücken, inwiefern Einfluss eine Verallgemeinerung der Ausgangsgrad-Zentralität $c_D^+(G)$ ist.

2.41 Folgerung

$$c_I(G) = \alpha \cdot (c_D^+(G) + A \cdot c_I(G))$$

■ **Beweis:** Wie oben gezeigt ist $c_I(G) = \alpha A \cdot \mathbf{1} + \alpha A \cdot c_I(G)$, und es gilt $c_D^+(G) = A \cdot \mathbf{1}$. □

Die Lösung großer, dünn besetzter Gleichungssysteme (und wir können wie üblich davon ausgehen, dass die Adjazenzmatrix dünn ist), wird oft iterativ angenähert. Bei der sogenannten *Jacobi-Iteration* wird mit einer initialen Näherung gestartet und jede Gleichung einzeln unter Verwendung der Näherungen aus dem letzten Schritt gelöst. Hier können wir ganz ähnlich vorgehen durch

$$\begin{aligned} c_v^{(0)} &= 0 \\ c_v^{(i+1)} &= \alpha \sum_{(v,w) \in E} (1 + c_w^{(i)}) . \end{aligned}$$

Die folgende Aussage zeigt, dass sich die resultierende Folge zur Annäherung an die Zentralitätswerte eignet und die Summation wie nach Anwendung des Distributivgesetzes auf die unendliche Summe erfolgt. Jede Iteration benötigt nur $\mathcal{O}(n + m)$ Schritte.

2.42 Lemma

Es ist $c^{(i+1)} \geq c^{(i)}$ für alle $i \in \mathbb{N}_0$ und $c^{(i)} \xrightarrow{i \rightarrow \infty} c_I(G)$

■ **Beweis:** Wir zeigen

$$c^{(i)} = \left(\sum_{k=1}^i (\alpha A)^k \right) \cdot \mathbf{1}$$

für alle $i \in \mathbb{N}_0$ per Induktion (d.h. in jeder Iteration wird ein Reihenglied hinzugefügt). Für $i = 0$ ist $c^{(0)} = \mathbf{0} = \left(\sum_{k=1}^0 (\alpha A)^k \right) \cdot \mathbf{1}$, und für $i = 1$ folgt $c^{(1)} = \alpha A \cdot \mathbf{1} = \left(\sum_{k=1}^1 (\alpha A)^k \right) \cdot \mathbf{1}$. Für $i > 0$ gilt

$$c^{(i+1)} = \alpha A \cdot \mathbf{1} + \alpha A \cdot c^{(i)}$$

nach Definition und unter Verwendung der Induktionsvoraussetzung

$$\begin{aligned} c^{(i+1)} &= \alpha A \cdot \mathbf{1} + \alpha A \cdot \left(\sum_{k=1}^i (\alpha A)^k \right) \cdot \mathbf{1} \\ &= \alpha A \cdot \mathbf{1} + \left(\sum_{k=2}^{i+1} (\alpha A)^k \right) \cdot \mathbf{1} \\ &= \left(\sum_{k=1}^{i+1} (\alpha A)^k \right) \cdot \mathbf{1} \end{aligned}$$

□

Das Abbruchkriterium im folgenden Algorithmus ist lediglich ein Beispiel.

Algorithmus 13: Einfluss-Zentralitäten (näherungsweise)

Eingabe: Multigraph $G = (V, E)$, Toleranz ε

Daten: Knotenarray c' (letzte Näherung)

Ausgabe: Knotenarray c (Einfluss bis auf ε , initialisiert mit 0)

$$\alpha \leftarrow \frac{1}{\min\{\Delta^-(G), \Delta^+(G)\} + 1}$$

repeat

$c' \leftarrow c$

foreach $v \in V$ **do**

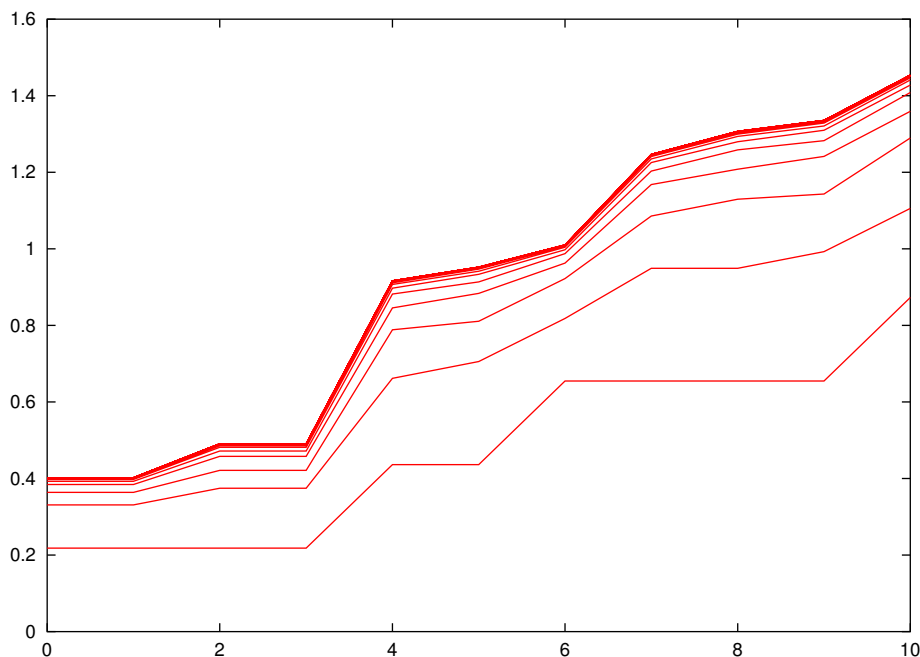
$c_v \leftarrow 0$

foreach $(v, w) \in E$ **do** $c_v \leftarrow c_v + \alpha \cdot (1 + c'_w)$

until $\max_{v \in V} |c_v - c'_v| < \varepsilon$

2.43 Beispiel (und einige Fragen)

Der Algorithmus angewandt auf den Beispielgraphen. Auf der x -Achse sind die Knoten geordnet nach Einfluss-Zentralität aufgeführt. Jede Linie entspricht der Näherung nach einer Iteration. Von 1000 Iterationen sind die ersten zwanzig sowie die letzte gezeigt.



Die Konvergenz ist in diesem Beispiel gut. Ist das immer so? Wann ist es (nicht) so?

Die Einfluss-Zentralitäten sind in diesem Beispiel sogar eine Verfeinerung der Ausgangsgrade (ein Knoten mit höherem Ausgangsgrad hat auch höheren Einfluss). Ist das immer so? Wann ist das (nicht) so?

Es hat auch während der ersten Iterationen keine Rangvertauschungen gegeben. Warum? Ist das immer so? Wann (nicht)?

Wir haben gesehen, dass drei Faktoren den Einfluss eines Knotens bestimmen: die Zahl seiner Nachbarn, deren Einfluss, und eine vorgegebene Abschwächung des Einflusses entlang jeder Kante. Verzichten wir auf den Knotengrad, zählen nur noch die wechselseitigen Abhängigkeiten der Bewertungen. Die Zentralität eines Knotens ist dann proportional zu der seiner Nachfolger, und für einen entsprechenden Index c heißt das

$$c_v = \alpha \cdot \sum_{(v,w) \in E} c_w \quad \text{bzw.} \quad c = \alpha A \cdot c .$$

Ein geeigneter Proportionalitätsfaktor lässt sich hierfür allerdings nicht so leicht angeben wie bei der Einfluss-Zentralität – es muss ihn nicht einmal immer geben.

2.44 Definition (Eigenwert, Eigenvektor)

Gilt für eine quadratische Matrix A und einen Vektor x

$$A \cdot x = \lambda x,$$

dann heißt x Eigenvektor von A . Ist $x \neq \mathbf{0}$, so heißen λ Eigenwert und (λ, x) Eigenpaar von A .

Da wir für $\lambda = \frac{1}{\alpha}$ aus der Zentralitätsgleichung

$$A \cdot c = \lambda c$$

erhalten, bieten sich positive reelle Eigenwerte λ der Adjazenzmatrix für die Wahl der Abschwächung an. Es muss dann allerdings auch einen zugehörigen nicht-negativen Eigenvektor geben. Wir werden zeigen, dass für stark zusammenhängende Multigraphen immer ein entsprechendes Eigenpaar existiert.

Im folgenden sei daher A die Adjazenzmatrix eines stark zusammenhängenden Multigraphen $G = (V, E)$. Ungleichungen, Beträge, usw. von Vektoren und Matrizen sind immer komponentenweise zu verstehen.

2.45 Definition

Ein Vektor $x \in \mathbb{R}_{\geq 0}^V \setminus \{\mathbf{0}\}$ ist eine ρ -pseudoharmonische Bewertung von G , falls

$$A \cdot x \geq \rho x .$$

2.46 Lemma

Ist G ein stark zusammenhängender Multigraph, dann existiert ein größtes $\rho \in \mathbb{R}_{>0}$, für das es eine ρ -pseudoharmonische Bewertung von G gibt.

■ **Beweis:** Sei

$$F(x) = \min_{v \in V: x_v \neq 0} \frac{(Ax)_v}{x_v}$$

definiert auf der Menge der nicht-negativen Knotenbewertungen $\mathbb{R}_{\geq 0}^V \setminus \{\mathbf{0}\}$. Jeder nicht-negative Vektor x ist $F(x)$ -pseudoharmonisch. Da F invariant unter Multiplikation mit einem Skalar ist, wäre die Behauptung also bewiesen, wenn wir einen Vektor y in der Menge

$$P = \{x \in \mathbb{R}^V : x \geq \mathbf{0}, \sum_{v \in V} x_v = 1\} .$$

angeben könnten, für den $F(y)$ maximal ist. Weil P kompakt ist, würde dessen Existenz folgen, wenn F auf P stetig wäre. Dies ist aber an den Rändern nicht der Fall.

Statt P betrachten wir daher zunächst die Menge

$$P' = (I + A)^{n-1} \cdot P .$$

Die Matrix $A' = (I + A)$ ist die Adjazenzmatrix des Multigraphen G' , den man aus G erhält, indem man an jedem Knoten eine Schleife hinzufügt. Da im stark zusammenhängenden Multigraphen G keine zwei Knoten einen Abstand größer $n - 1$ haben können, ist die Zahl der Kantenfolgen der Länge $n - 1$ von irgendeinem Knoten zu irgendeinem anderen in G' echt größer als Null. Folglich sind A' und damit auch alle Vektoren in P' positiv. Außerdem ist auch P' kompakt. Weil aber F stetig auf ganz P' ist, nimmt F einen maximalen Wert ρ für einen Vektor $z \in P'$ an. Durch Setzen von

$$y_v = \frac{z_v}{\sum_{w \in V} z_w}$$

erhalten wir auch ein $y \in P$ mit $F(y) = F(z) = \rho$. Da aber außerdem

$$F((I + A)^{n-1}x) \geq F(x),$$

gibt es kein $x \in P$ mit $F(x) > \rho$. □

2.47 Lemma

Sei x eine ρ -pseudoharmonische Bewertung eines stark zusammenhängenden Multigraphen G mit maximalem $\rho \in \mathbb{R}$. Dann ist (ρ, x) ein Eigenpaar von $A(G)$.

■ **Beweis:** Sei

$$\sigma(x) = \{v : (Ax)_v > \rho x_v\}$$

die Menge der nicht im ρ -Gleichgewicht befindlichen Knoten, dann ist x genau dann ein Eigenvektor von A , wenn $\sigma(x) = \emptyset$. Nehmen wir also an, $\sigma(x)$ wäre nicht leer.

Der Träger einer Knotenbewertung x ist die Menge der Knoten mit Bewertung ungleich Null. Sei $h \in \mathbb{R}_{\geq 0}^V \setminus \{\mathbf{0}\}$ ein nicht-negativer Vektor mit Träger $\sigma(x)$. Für den Vektor $y = x + \varepsilon h$ gilt

$$(Ay)_v - \rho y_v = (Ax)_v - \rho x_v + \varepsilon(Ah)_v - \varepsilon \rho h_v .$$

Für $v \in \sigma(x)$ ist $(Ax)_v > \rho x_v$, und für hinreichend kleines ε damit auch die rechte Seite der obigen Gleichung. Folglich $(Ay)_v > \rho y_v$.

Für $v \notin \sigma(x)$ sind $(Ax)_v = \rho x_v$ und $h_v = 0$. Folglich $(Ay)_v - \rho y_v = \varepsilon(Ah)_v$. Für $\varepsilon > 0$ ist die rechte Seite nicht-negativ, und weil G stark zusammenhängend ist, existiert mindestens ein $v \notin \sigma(x)$ so, dass $(Ah)_v > 0$.

Zusammengenommen haben wir $\sigma(x) \subset \sigma(y)$. Gilt $|\sigma(y)| = n$, dann ist y auch ρ' -pseudoharmonisch für ein $\rho' > \rho$, was ein Widerspruch ist. Andernfalls ist y aber immerhin ρ -pseudoharmonisch und $|\sigma(y)| > |\sigma(x)|$, sodass wir das Argument mit y anstelle von x wiederholen können. Nach einer endlichen Anzahl von Wiederholungen ergibt sich der Widerspruch zur Maximalität von ρ . \square

Der *Spektralradius* $\rho(G)$ eines Multigraphen ist der größte Betrag eines Eigenwertes seiner Adjazenzmatrix. Dieser muss selbst nicht Eigenwert sein, und der zugehörige Eigenwert muss auch nicht reell sein.

2.48 Lemma

Ist G ein stark zusammenhängender Multigraph, dann gibt es eine $\rho(G)$ -pseudoharmonische Bewertung von G .

■ **Beweis:** Sei ρ der maximale Wert, sodass G eine ρ -pseudoharmonische Bewertung hat. Für jede $n \times n$ -Matrix B mit $|B| \leq A = A(G)$ und Eigenpaar (θ, x) gilt

$$|\theta||x| = |\theta x| = |Bx| \leq |B||x| \leq A|x| ,$$

d.h. $|x|$ ist ein $|\theta|$ -pseudoharmonischer Vektor von G und daher $|\theta| \leq \rho$. Da insbesondere A selbst die Voraussetzungen erfüllt, ist $\rho = \rho(G)$. \square

2.49 Lemma

Die $\rho(G)$ -pseudoharmonische Bewertung eines stark zusammenhängenden Multigraphen G ist eindeutig bis auf Skalarmultiplikation und hat nur positive Einträge.

■ **Beweis:** Wir zeigen zunächst, dass jede $\rho(G)$ -pseudoharmonische Bewertung nur positive Einträge hat. Angenommen, $x_v = 0$ für ein $v \in V$ (pseudoharmonische Bewertungen sind nicht-negativ). Da $(\rho(G), x)$ Eigenpaar von A ist, gilt auch $(Ax)_v = \rho(G)x_v = 0$. Andererseits gilt

$$(Ax)_v = \sum_{w \in N^+(v)} a_{v,w} \cdot x_w ,$$

woraus wegen $A \geq 0$

$$a_{v,w} \neq 0 \implies x_w = 0$$

folgt. Da G aber stark zusammenhängend ist, liefert ein einfaches Induktionsargument, dass $x = \mathbf{0}$. Dann ist x aber keine pseudoharmonische Bewertung.

Nehmen wir also noch an, es gäbe zwei linear unabhängige $\rho(G)$ -pseudoharmonische Bewertungen x und y . Für jede Linearkombination gilt dann

$$A(ax + by) = aAx + bAy \geq a\rho(G)x + b\rho(G)y = \rho(G)(ax + by)$$

und wir können $a, b \in \mathbb{R}$ so wählen, dass $ax + by$ nicht-negativ und an einer Stelle Null ist. Dies ist ein Widerspruch zur bereits bewiesenen Teilaussage. \square

Die hergeleiteten Eigenschaften sind im zentralen Satz des Abschnitts zusammengefasst, der uns die Definition einer Zentralität erlaubt, in der die Bewertungen nur von den Bewertungen der jeweiligen Nachbarn abhängen.

2.50 Satz (Perron 1907 und Frobenius 1912)

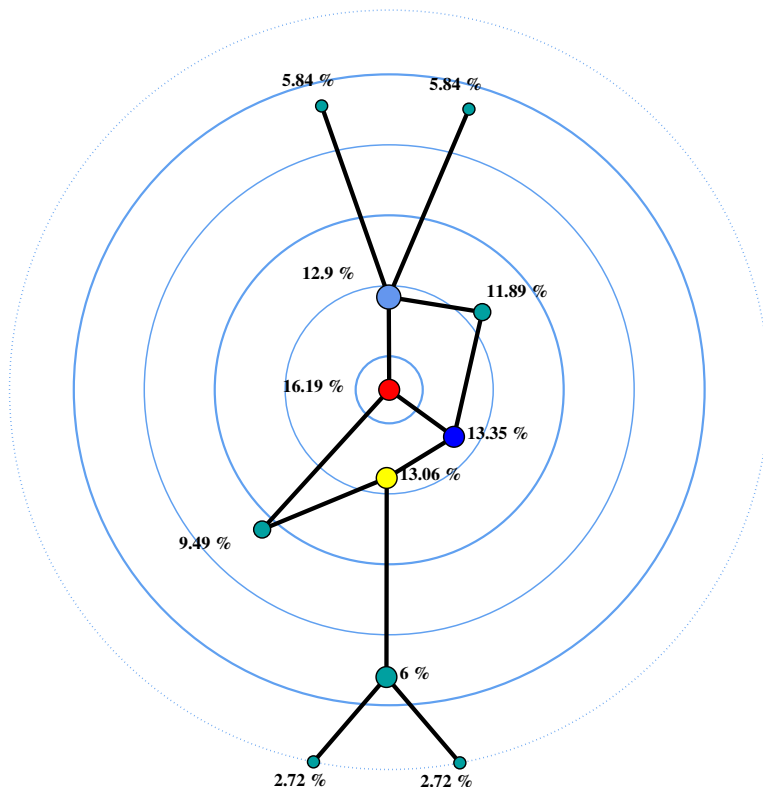
Der Spektralradius $\rho(G)$ der Adjazenzmatrix eines stark zusammenhängenden Multigraphen G ist ein Eigenwert. Der zugehörige Eigenvektor ist eindeutig bis auf Skalarmultiplikation und alle seine Einträge haben das gleiche Vorzeichen.

2.51 Definition (Eigenvektor-Zentralität; Bonacich 1972)

Die Eigenvektor-Zentralität c_E ist definiert für alle $G = (V, E) \in \mathcal{S}$ als die eindeutige Lösung von

$$c_E(G) = \frac{1}{\rho(G)} \cdot A(G) \cdot c_E(G)$$

mit $c_E(G)_v > 0, v \in V$, und $\sum_{v \in V} c_E(G)_v = 1$ (d.h. die Eigenvektor-Zentralität wird grundsätzlich normiert).



normierte Eigenvektor-Zentralität im Beispielgraphen

Vermutung: $c_E \in \bullet \rightarrow \circ(\mathcal{S})$ (ohne Beweis)

Die exakte Berechnung von Eigenvektoren ist schwierig. Von den zahlreichen numerischen Methoden zu ihrer Annäherung betrachten wir nur die einfachste: die sogenannte *Potenziteration*.

Wie bei der iterativen Bestimmung der Einfluss-Zentralitäten wird die punktweise Gleichgewichtsgleichung genutzt, um eine Folge von Näherungen zu erzeugen. Ausgehend von einem beliebigen Startvektor $c^{(0)}$, etwa $c^{(0)} = \mathbf{1}$, besteht die durch

$$c^{(i+1)} = Ac^{(i)}$$

definierte Folge von Schätzwerten aus den Vektoren $A^i c^{(0)}$, die – abgesehen von einem problematischen Sonderfall – eine Näherung für die Eigenvektor-Zentralität liefern.

2.52 Lemma

Für einen stark zusammenhängenden, nicht bipartiten Multigraphen G gilt für geeignetes $c^{(0)}$

$$\frac{c^{(i)}}{\|c^{(i)}\|} \xrightarrow{i \rightarrow \infty} \frac{c_E(G)}{\|c_E(G)\|} .$$

■ **Beweis:** Seien $\lambda_1, \dots, \lambda_n$ die Eigenwerte von $A(G)$. Da G stark zusammenhängend und nicht bipartit ist, können wir $\rho(G) = \lambda_1 > |\lambda_2| \geq \dots \geq |\lambda_n|$ annehmen (dabei wird benutzt, dass $-\rho(G)$ genau dann Eigenwert von G ist, wenn G bipartit ist; s. Übung). Ferner seien x_1, \dots, x_n zugehörige Eigenvektoren, insbesondere also x_1 proportional zum gesuchten. Lässt sich der Startvektor mit geeigneten Koeffizienten b_1, \dots, b_n als

$$c^{(0)} = b_1 x_1 + \dots + b_n x_n$$

schreiben, dann folgt daraus mit Induktion über i

$$A^i c^{(0)} = \sum_{j=1}^n b_j \lambda_j^i x_j$$

und, falls $b_1 \neq 0$,

$$A^i c^{(0)} = b_1 \lambda_1^i \left(x_1 + \sum_{j=2}^n \frac{b_j}{b_1} \left(\frac{\lambda_j}{\lambda_1} \right)^i x_j \right) .$$

Wegen $\lambda_1 > |\lambda_j|$ für alle $j = 2, \dots, n$ folgt die Behauptung. □

2.53 Beispiel

Betrachte den zusammenhängenden bipartiten Graphen C_4 . Die Eigenwerte sind $(\lambda_1, \lambda_2, \lambda_3, \lambda_4) = (2, 0, 0, -2)$ (**s. Übung**) mit Eigenvektoren $x_1 = (1, 1, 1, 1)^T$, $x_2 = (1, 0, -1, 0)^T$, $x_3 = (0, 1, 0, -1)^T$ und $x_4 = (-1, 1, -1, 1)^T$. Bei Startvektor $c^{(0)} = 2x_1 + x_4$ (also $a_1 \neq 0$) ergibt sich jedoch die Folge

$$c^{(i)} = \begin{cases} (2^i, 3 \cdot 2^i, 2^i, 3 \cdot 2^i)^T & \text{falls } i \text{ gerade} \\ (3 \cdot 2^i, 2^i, 3 \cdot 2^i, 2^i)^T & \text{falls } i \text{ ungerade,} \end{cases}$$

die sich keinem Eigenvektor nähert.

Da die Folgenglieder stark wachsen, normieren wir in jeder Iteration. Die Konvergenzaussage wird dadurch nicht beeinflusst.

Algorithmus 14: Eigenvektor-Zentralitäten (näherungsweise)

Eingabe: Multigraph $G = (V, E) \in \mathcal{S}$, nicht bipartit

Abbruchkriterium ε

Daten: Knotenarray c' (letzte Näherung)

Ausgabe: Knotenarray c (Eigenvektor-Zentralität)

foreach $v \in V$ **do** $c_v \leftarrow \frac{1}{n}$

repeat

$c' \leftarrow c$

foreach $v \in V$ **do**

$c_v \leftarrow 0$

foreach $w \in N_G^+(v)$ **do** $c_v \leftarrow c_v + a_{v,w} \cdot c'_w$

$\rho \leftarrow \|c\|$; $c \leftarrow \frac{c}{\rho}$

until ρ während der letzten Iterationen nur um ε verändert

2.54 Bemerkung

Die Laufzeit des Verfahrens ist wieder $\mathcal{O}(n + m)$ pro Iteration. Allerdings hängt die Konvergenzgeschwindigkeit der Potenziteration vom Verhältnis $\frac{|\lambda_2|}{|\lambda_1|}$ ab und kann daher sehr schlecht sein. Das Abbruchkriterium kann deutlich geschickter gewählt werden. Für mittelgroße Graphen ist der obige Algorithmus jedoch durchaus praktikabel.

Hubs & Authorities

2.55 Beispiel (Bibliographische Netzwerke, WWW)

Bei der Analyse wissenschaftlicher Publikationen interessiert man sich für einflussreiche Publikationen, bedeutende Autoren, wichtige Zeitschriften, usw. Die Datenbasis besteht normalerweise aus mindestens einem von zwei Netzwerktypen:

	Knoten	Kanten
Zitiernetzwerk	Publikationen	v zitiert w
Autorenschaft	Autoren & Publikationen	v ist Autor von w

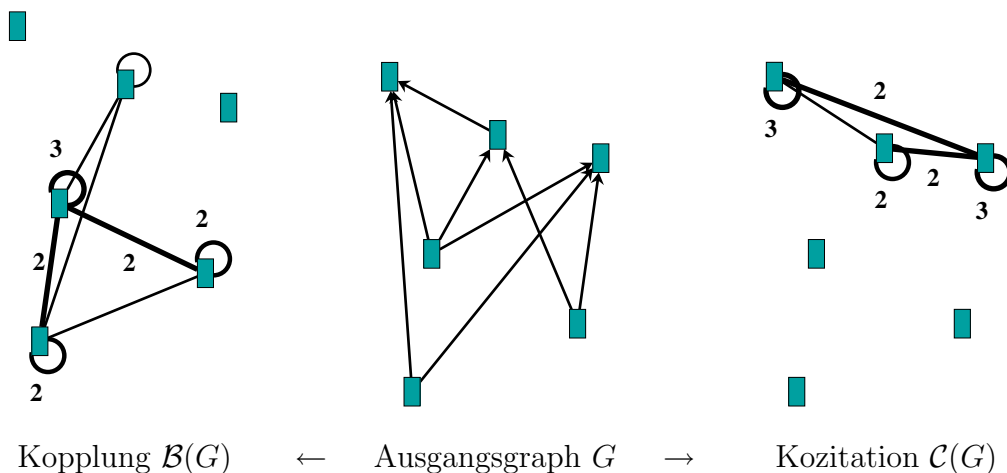
Bei wissenschaftlichen Veröffentlichungen ist der zweite Typ wesentlich einfacher erhältlich, da sich diese Netzwerke leicht aus bibliographischen Datenbanken automatisch erzeugen lassen. Zitate sind aus Aufsätzen und Büchern sehr aufwändig zu extrahieren, sodass entsprechende Datensätze (z.B. Science Citation Index) sehr teuer sind.

Eine andere Form der Publikation von Wissen sind WWW-Seiten. Hier ist das Aufwandsverhältnis genau umgekehrt, da sich Zitate (Links) aus HTML-Seiten sehr leicht extrahieren lassen, Autoren jedoch selten zuverlässig feststellbar sind.

Wir betrachten zwei Operatoren, mit denen aus gerichteten Multigraphen (wie etwa Zitier- und Autorenschaftsnetzwerken) symmetrische (ungerichtete) Multigraphen für Teilaspekte einer Analyse gewonnen werden.

2.56 Definition (Bibliographische Operatoren; Kessler 1963 und Small 1973)

Ist G ein Multigraph, dann heißt der symmetrische Multigraph $\mathcal{B}(G)$ mit Adjazenzmatrix $B(G) = A(\mathcal{B}(G)) = A(G)A(G)^T$ (bibliographische) Kopplung von G . Der symmetrische Multigraph $\mathcal{C}(G)$ mit Adjazenzmatrix $C(G) = A(\mathcal{C}(G)) = A(G)^T A(G)$ heißt Kozitation von G .



2.57 Bemerkung

Enthält G mindestens eine Kante, so sind weder $\mathcal{B}(G)$ noch $\mathcal{C}(G)$ bipartit (s. Übung).

2.58 Beispiel (Erdős-Graph)

Angewandt auf ein Autorenschaftsnetzwerk ergibt die bibliographische Kopplung den Kollaborationsgraph. Der wohl berühmteste Kollaborationsgraph ist der Erdős-Graph. Dabei handelt es sich um den von allen wissenschaftlichen Publikationen induzierten Kollaborationsgraph zusammen mit einer Knotenbewertung (der Erdős-Zahl), die den Abstand vom Paul Erdős entsprechenden Knoten angibt. Erdős selbst hat die Erdős-Zahl 0, seine Koautoren die Erdős-Zahl 1, deren Koautoren, die nicht Erdős oder Koautor Erdős' sind, die Erdős-Zahl 2, usw.

Siehe auch <http://www.oakland.edu/~grossman/erdoshp.html>.

Im Rahmen des CLEVER-Projekts zur Verbesserung von WWW-Suchmaschinen wurde bei IBM das Verfahren HITS entwickelt, das wesentlich auf dem folgenden Zentralitätsindex beruht.

2.59 Definition (Hubs & Authorities; Kleinberg 1999)

Für $G \in \mathcal{G}$ ist die Hub-Zentralität c_H definiert durch

$$c_H(G) = c_E(\mathcal{B}(G))$$

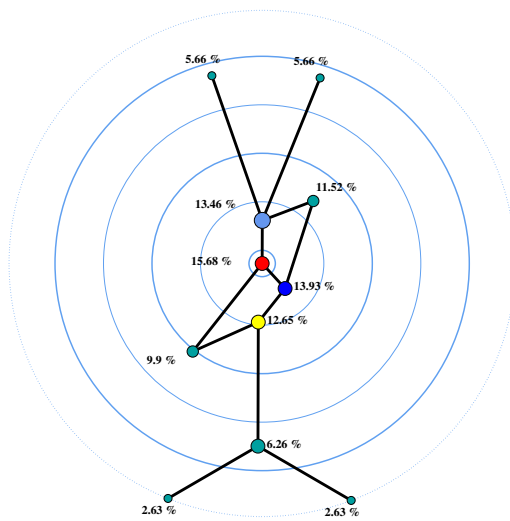
und die Authority-Zentralität c_A durch

$$c_A(G) = c_E(\mathcal{C}(G)) .$$

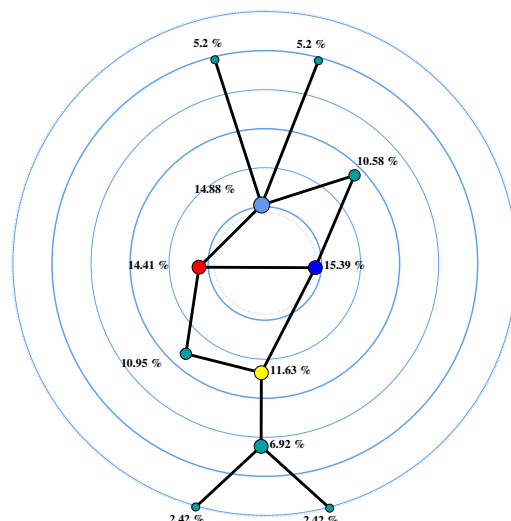
2.60 Bemerkung

Da die symmetrischen Multigraphen $\mathcal{B}(G)$ und $\mathcal{C}(G)$ nicht notwendig zusammenhängend sind, gehen wir in der obigen Definition von einer Verallgemeinerung der Eigenvektor-Zentralität auf Vereinigungen stark zusammenhängender Multigraphen aus. Die Zentralitäten in jeder Zusammenhangskomponente summieren sich dabei zur relativen Größe der Komponente auf.

ACHTUNG:
G ist bi-partit



normierte Hub-Zentralität
im Beispielgraphen
(falsch normiert)



normierte Authority-Zentralität
im Beispielgraphen
(falsch normiert)

Vermutung: $c_H \in \bullet \rightarrow \circ(\mathcal{G})$ und $c_A \in \circ \rightarrow \bullet(\mathcal{G})$

Der naheliegende Ansatz zur Berechnung von Hubs & Authorities, zunächst $\mathcal{B}(G)$ und $\mathcal{C}(G)$ zu berechnen und auf die Ergebnisse dann jeweils Potenziteration anzuwenden, ist im allgemeinen ungeeignet, da die Eigenschaft geringer Dichte verloren gehen kann.

2.61 Lemma

$n(G) = n(\mathcal{B}(G)) = n(\mathcal{C}(G))$, aber

$$\begin{aligned} m(\mathcal{B}(G)) &= \sum_{v \in V} d_G^-(v)^2 \\ m(\mathcal{C}(G)) &= \sum_{v \in V} d_G^+(v)^2 \end{aligned}$$

■ **Beweis:** Nach Definition haben die Adjazenzmatrizen der beiden abgeleiteten Multigraphen die gleichen Dimensionen, sodass die Anzahl der Knoten unverändert ist.

Für die Kardinalität der Kantenmenge von $\mathcal{B}(G)$ mit Adjazenzmatrix $B = (b_{u,w})_{u,w \in V}$ ist

$$\begin{aligned} m(\mathcal{B}(G)) &= \sum_{u,w \in V} b_{u,w} = \sum_{u,w \in V} (AA^T)_{u,w} \\ &= \sum_{u,w \in V} \sum_{v \in V} a_{u,v} \cdot a_{w,v} = \sum_{v \in V} \sum_{u \in V} \sum_{w \in V} a_{u,v} \cdot a_{w,v} \\ &= \sum_{v \in V} \sum_{u \in V} a_{u,v} \sum_{w \in V} a_{w,v} = \sum_{v \in V} \sum_{u \in V} a_{u,v} \cdot d_G^-(v) \\ &= \sum_{v \in V} d_G^-(v)^2 \end{aligned}$$

und analog für $m(\mathcal{C}(G))$. □

Der folgende Zusammenhang erlaubt es uns, die Potenziterationen für beide Zentralitätsindizes ineinander zu schachteln.

2.62 Lemma

$$c_H(G) = A(G) \cdot c_A(G) \text{ und } c_A(G) = A(G)^T \cdot c_H(G).$$

■ **Beweis:** Wir können annehmen, dass $\mathcal{B}(G)$ und $\mathcal{C}(G)$ zusammenhängend sind, denn andernfalls übertragen sich alle Argumente entsprechend auf die Zusammenhangskomponenten.

Im Beweis zur Konvergenz der Potenziteration wurde gezeigt, dass bei stark zusammenhängenden, nicht-bipartiten Multigraphen für einen geeigneten Startvektor x die Folge $A^i x$ gegen einen Eigenvektor des betragsgrößten Eigenwertes von $A = A(G)$ konvergiert.

Daher konvergiert die Folge $(AA^T)^i x$ für die meisten x gegen ein Vielfaches von $c_H(G)$, und die Folge $(A^T A)^i y$ für die meisten y gegen ein Vielfaches von $c_A(G)$. Wegen $(AA^T)^i x = A(A^T A)^{i-1}(A^T x)$ folgt der erste Teil der Behauptung mit Wahl von $y = A^T x$. Der zweite folgt analog. \square

Aus dem obigen Beweis können wir unmittelbar die Korrektheit des folgenden Näherungsalgorithmus folgern.

Algorithmus 15: Hub- & Authority-Zentralitäten (näherungsweise)

Eingabe: Multigraph $G = (V, E)$, Abbruchkriterium ε

Ausgabe: Knotenarray a (Authority-Zentralität)

Knotenarray h (Hub-Zentralität)

```

foreach  $v \in V$  do  $a_v \leftarrow \frac{1}{n}$ 
repeat
  foreach  $v \in V$  do
     $h_v \leftarrow 0$ 
    foreach  $(v, w) \in E$  do  $h_v \leftarrow h_v + a_w$ 
   $\rho_h \leftarrow \|h\|$ ;  $h \leftarrow \frac{h}{\rho_h}$ 
  foreach  $v \in V$  do
     $a_v \leftarrow 0$ 
    foreach  $(u, v) \in E$  do  $a_v \leftarrow a_v + h_u$ 
   $\rho_a \leftarrow \|a\|$ ;  $a \leftarrow \frac{a}{\rho_a}$ 
until  $\rho_a, \rho_h$  während der letzten Iterationen nur um  $\varepsilon$  verändert

```

Trotz der möglicherweise höheren Dichte von $\mathcal{B}(G)$ und $\mathcal{C}(G)$ ist die pro Iteration benötigte Zeit damit wieder nur $\mathcal{O}(n(G) + m(G))$.

PageRank

Die Suchmaschine Google verwendet für die Relevanzbewertung unter anderem einen Zentralitätsindex namens PageRank. Dieser kann als Variante sowohl der Einfluss- als auch der Eigenvektor-Zentralität angesehen werden.

2.63 Definition

Zu einem Multigraph $G = (V, E)$ sei die Ausgangsgradmatrix $D^+(G) = (d_{v,w})_{v,w \in V}$ definiert durch

$$d_{v,w}^+ = \begin{cases} d_G^+(v) & \text{falls } v = w \\ 0 & \text{sonst .} \end{cases}$$

Die (ausgangsgrad-)normalisierte Adjazenzmatrix definieren wir als

$$M^+(G) = (D^+(G))^{-1}A(G)$$

wobei für $(D^+(G))^{-1} = (d_{v,w}^{-1})_{v,w \in V}$ gelte

$$d_{v,w}^{-1} = \begin{cases} d_G^+(v)^{-1} & \text{falls } v = w \text{ und } d_G^+(v) > 0 \\ 0 & \text{sonst .} \end{cases}$$

2.64 Bemerkung

In normalisierten Adjazenzmatrizen ist die Zeilensumme für Knoten mit mindestens einer ausgehenden Kante gleich Eins. Sie können als Adjazenzmatrix eines (kanten-)gewichteten Multigraphen $\mathcal{M}^+(G)$ interpretiert werden. Entsprechende Definitionen und Aussagen für den Eingangsgrad und den Knotengrad sind natürlich ebenfalls möglich, werden aber hier nicht benötigt.

Bei der Eigenvektor-Zentralität werden die Werte an den Nachfolgerknoten aufsummiert. Nimmt man stattdessen den Mittelwert, so ergibt sich

$$c_v = \alpha \sum_{w \in N_G^+(v)} \frac{a_{v,w}}{d_G^+(v)} \cdot c_w \quad \text{bzw.} \quad c = \alpha M^+(G)c .$$

Die Bewertung c heißt dann harmonisch. Die Zentralität eines Knotens ist damit also proportional zur mittleren Zentralität der von ihm direkt beeinflussten Knoten. In WWW-Zitiernetzwerken ist diese Sichtweise sicher wenig plausibel, da Links keine Beeinflussung der Seiten darstellen, auf die verwiesen wird.

Wir hatten bereits festgestellt, dass aus einer $\bullet \rightarrow \circ$ Zentralität eine $\circ \rightarrow \bullet$ Zentralität wird, wenn wir sie im Graphen mit umgedrehten Kanten berechnen. Da PageRank die Relevanz von WWW-Seiten bewerten soll, bietet sich die so umgekehrte Sichtweise an: Die Relevanz einer Seite überträgt sich zu gleichen Teilen auf die durch ausgehende Hyperlinks referenzierten Seiten. Eine entsprechende Bewertung müsste dann

$$c_v = \sum_{u \in N_G^-(v)} \frac{a_{u,v}}{d_G^+(u)} \cdot c_u \quad \text{bzw.} \quad c = M^+(G)^T c .$$

erfüllen. Ein Proportionalitätsfaktor ist hier nicht nötig:

2.65 Lemma

Ist G ein stark zusammenhängender, nicht bipartiter Multigraph, so gilt $\rho(\mathcal{M}^+(G)^T) = 1$.

■ **Beweis:** Aus den Überlegungen zur Potenziteration wissen wir, dass die Folge $(M^+(G)^T)^i x$ für geeignete x und $i \rightarrow \infty$ gegen einen zum Spektralradius gehörigen Eigenvektor mit ausschließlich positiven Einträgen konvergiert.

Außerdem gilt

$$\begin{aligned} \sum_{v \in V} (M^+(G)^T x)_v &= \sum_{v \in V} \sum_{u \in N^-(v)} \frac{a_{u,v}}{d^+(u)} \cdot x_u = \sum_{(u,v) \in E} \frac{1}{d^+(u)} \cdot x_u \\ &= \sum_{u \in V} \sum_{v \in N^+(u)} \frac{a_{u,v}}{d^+(u)} \cdot x_u = \sum_{u \in V} x_u \sum_{v \in N^+(u)} \frac{a_{u,v}}{d^+(u)} \\ &= \sum_{u \in V} x_u , \end{aligned}$$

d.h. die Summe der Einträge eines Vektors ist invariant unter Multiplikation mit der normalisierten Adjazenzmatrix. Dies gilt insbesondere für den positiven Eigenvektor zum Spektralradius, sodass letzterer gleich Eins sein muss. \square

2.66 Bemerkung (Irrfahrten auf Graphen und das WWW)

Die Potenziteration für $(M^+(G))^T$ kann als Irrfahrt auf G interpretiert werden:

- Die Knoten von G beschreiben mögliche Aufenthaltsorte.
- Die ausgehenden Kanten eines Knotens entsprechen den erlaubten Übergängen an Aufenthaltsorte zum nächsten Zeitpunkt.
- Der Wechsel des Aufenthaltsortes geschieht zufällig, d.h. jeder mögliche Übergang findet mit gleicher Wahrscheinlichkeit statt.

Bei Vorgabe einer Anfangsverteilung $x^{(0)}$, etwa der Gleichverteilung $\frac{1}{n} \cdot \mathbf{1}$, beschreibt der Vektor $(M^+(G))^T x^{(0)}$ die Aufenthaltswahrscheinlichkeiten an den Knoten nach genau i Übergängen.

Die Matrix $(M^+(G))^T$ heißt in dieser Interpretation auch Übergangsmatrix. Drücken wir die Aufenthaltswahrscheinlichkeiten zu Zeitpunkt i durch Zufallsvariablen $X^{(i)}$ aus, dann heißt $X^{(0)}, X^{(1)}, \dots$ auch (homogene) Markoff-Kette, und die Konvergenzbedingungen stark zusammenhängend und bipartit entsprechen den Bedingungen irreduzibel und aperiodisch für die Existenz einer stationären Verteilung (einer Verteilung, die sich durch Anwendung der Übergangsmatrix nicht ändert).

Statt als rückgekoppelte Zentralität kann ein Eigenvektor zum größten Eigenwert 1 von $(M^+(G))^T$ daher auch als die stationäre Verteilung einer Irrfahrt auf dem Graphen gedeutet werden. In WWW-Graphen lässt sich diese interpretieren als die Aufenthaltswahrscheinlichkeit eines zufällig umherirrenden Benutzers auf der jeweiligen Seite.

Will man die Aufenthaltswahrscheinlichkeiten zur Relevanzbewertung der Datenbasis einer Suchmaschine verwenden, ist die Einschränkung auf stark zusammenhängende, nicht bipartite Multigraphen unrealistisch. Die folgende Idee beseitigt das Problem: Statt in jedem Schritt einem Link zu einer Seite zu folgen, kann der herumirrende Benutzer mit einer Wahrscheinlichkeit ω auch zu irgendeiner Seite springen. Dadurch wird der Multigraph der möglichen Übergänge vollständig, insbesondere also stark zusammenhängend und nicht bipartit.

Bei gleich wahrscheinlichen Sprungzielen erhalten wir die von Google verwendete Zentralität.

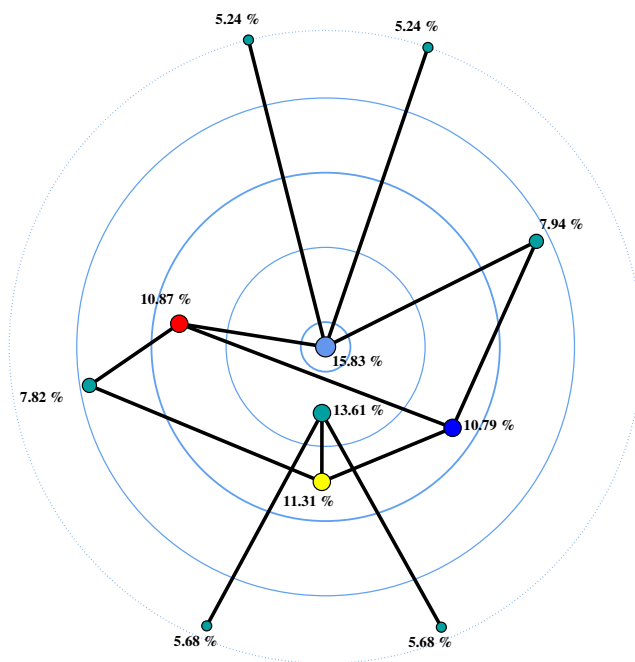
2.67 Definition (PageRank; Brin und Page 1998)

Für $G \in \mathcal{G}$ und ein $0 < \omega < 1$ ist der PageRank c_P definiert als die eindeutige Lösung von

$$c_P(G) = (1 - \omega) \cdot M^+(G)^T c_P(G) + \frac{\omega}{n} \cdot \mathbf{1} .$$

Vermutung: $c_P \in \circ \rightarrow \bullet(\mathcal{G})$

S?



PageRank im Beispielgraphen ($\omega = \frac{1}{4}$)

Offene Frage: Welche Auswirkungen hat ω auf die *Reihung* der Knoten?

Algorithmus 16: PageRank (näherungsweise)

Eingabe: Multigraph $G = (V, E)$, Sprungweite $0 < \omega < 1$

Abbruchkriterium ε

Daten: Knotenarray c' (letzte Näherung)

Ausgabe: Knotenarray c (PageRank)

```

foreach  $v \in V$  do  $c_v \leftarrow \frac{1}{n}$ 
repeat
   $c' \leftarrow c$ 
  foreach  $v \in V$  do
     $c_v \leftarrow \frac{\omega}{n}$ 
    foreach  $u \in N^-(v)$  do  $c_v \leftarrow c_v + (1 - \omega) \frac{a_{u,v}}{d^+(u)} \cdot c'_u$ 
until  $\|c - c'\| < \varepsilon$ 

```

2.3 Kantenzentralitäten

Wir wollen noch kurz anreißen, wie Kantenzentralitäten definiert werden könnten. Die dazugehörige Theorie ist noch weniger weit entwickelt als die für Knotenzentralitäten. Geeignete notwendige Bedingungen an Kantenstrukturindizes sind nicht bekannt.

Zumindest bei der Betweenness-Zentralität lässt sich die Grundidee unmittelbar auf Kanten übertragen.

2.68 Definition (Kanten-Betweenness; Anthonisse 1971)

Die Erweiterung der Betweenness-Zentralität c_B auf Kanten ist definiert durch

$$c_B(G)_e = \sum_{s,t \in V} \frac{\sigma_G(s,t|e)}{\sigma_G(s,t)}$$

für alle $G = (V, E) \in \mathcal{G}$ (mit den Bezeichnungen aus der Knoten-Betweenness-Zentralität).

2.69 Lemma

$$\delta_G(s|(v,w)) = \begin{cases} \frac{\sigma_G(s,v)}{\sigma_G(s,w)} \cdot (1 + \delta_G(s|w)) & \text{falls } (v,w) \in P_G^-(s,w) \\ 0 & \text{sonst.} \end{cases}$$

■ **Beweis:** Falls $(v,w) \notin P_G^-(s,w)$, also auf keinem kürzesten (s,w) -Weg liegt, dann liegt (v,w) auch auf keinem kürzesten (s,t) -Weg für irgend ein $t \in V$, sodass $\delta_G(s|(v,w)) = 0$.

Im Beweis zur Abhängigkeit von Knoten hatten wir bereits gesehen, dass

$$\delta_G(s,t|v,(v,w)) = \begin{cases} \frac{\sigma_G(s,v)}{\sigma_G(s,w)} & \text{falls } t = w \\ \frac{\sigma_G(s,v)}{\sigma_G(s,w)} \cdot \frac{\sigma_G(s,t|w)}{\sigma_G(s,t)} & \text{falls } t \neq w \end{cases}$$

Da aber $\delta_G(s,t|v,(v,w)) = \delta_G(s,t|(v,w))$ folgt die Behauptung durch Summation über alle $t \in V$. □

Zur Berechnung brauchen wir also lediglich den Betweenness-Algorithmus um Abhängigkeiten von Kanten zu erweitern. Für Multigraphen mit Mehrfachkanten ist wieder zu beachten, dass die Vorgängerzählung angepasst werden muss.

Algorithmus 17: Abhängigkeiten eines Knotens $s \in V$

(Spezialisierung der Breitensuche mit Wurzel s)

Daten : Knotenarray σ_s (Anzahl der kürzesten Wege von s)
 Knotenarray P_s (Liste der Vorgänger auf kürzesten Wegen)
 Stack S (Knoten in Reihenfolge ihres Abstands von s)

Ausgabe: Knoten- und Kantenarray δ_s
 (Abhängigkeiten, initialisiert mit 0)

root(vertex s) begin

 | $\sigma_s(s) = 1$

end

traverse(vertex v , edge e , vertex w) begin

 | **if e ist Baumkante then**

 | füge an $P_s(w) \leftarrow v$

 | $\sigma_s(w) \leftarrow \sigma_s(v)$

 | push $w \rightarrow S$

 | **if e ist Vorwärtskante then**

 | füge an $P_s(w) \leftarrow v$

 | $\sigma_s(w) \leftarrow \sigma_s(w) + \sigma_s(v)$

end

done(vertex s) begin

 | **while S nicht leer do**

 | $w \leftarrow pop(S)$

 | **foreach $v \in P_s(w)$ do**

 | $\delta_s(v) \leftarrow \delta_s(v) + \frac{\sigma_s(v)}{\sigma_s(w)} \cdot (1 + \delta_s(w))$

 | $\delta_s((v, w)) \leftarrow \frac{\sigma_s(v)}{\sigma_s(w)} \cdot (1 + \delta_s(w))$

end

2.70 Beispiel (Clustern mit Kanten-Betweenness)

Haben Kanten eine hohe Betweenness-Zentralität, dann sind sie wesentliche Verbindungen zwischen verschiedenen Teilgraphen. Insbesondere haben Brücken eine hohe Betweenness, die umso größer ist, je ähnlicher die Größe der Teilgraphen ist, zu denen die beiden Endknoten gehören.

Eine Idee, durch wiederholtes Entfernen von Kanten den Graphen schrittweise in Cluster zu zerlegen, besteht daher darin, die jeweils am höchsten bewerteten Kanten auszuwählen. Der nachfolgende Algorithmus beschreibt das Vorgehen im Detail.

Algorithmus 18: Betweenness-Clustern (Girvan und Newman 2001)

Eingabe : Multigraph $G = (V, E)$

Ausgabe: Teilgraph $C \subseteq G$

(Zusammenhangskomponenten definieren die Cluster)

$C \leftarrow G$

repeat

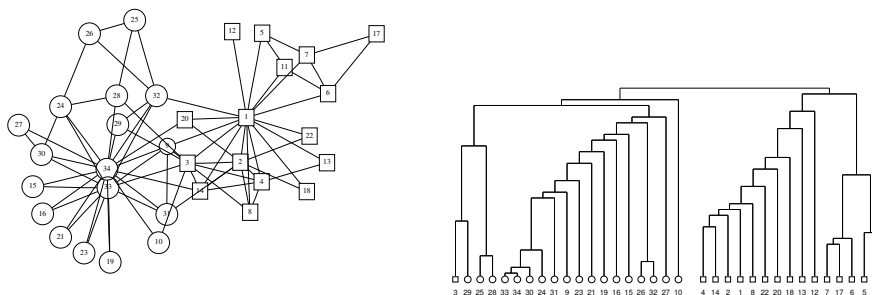
$B \leftarrow \{e \in E(C) : c_B(C)_e \text{ maximal}\}$

 entferne die Kanten in B aus C

until bel. Abbruchbedingung (z.B. Anzahl Cluster, Clustergrößen)

Die Laufzeit ist im schlechtesten Fall $\mathcal{O}(nm^2)$, denn in jeder Iteration muss die Kanten-Betweenness neu berechnet werden. Sie ist in der Regel aber niedriger, wenn z.B. die Kanten-Betweenness in gut aufgeteilten Graphen nur noch in kleinen Komponenten bestimmt werden muss.

Das folgende Netzwerk beschreibt die Freundschaftsbeziehungen in einem Karate-Verein. Das anschließende Dendrogramm gibt die Aufteilung der Cluster an (aus: Girvan und Newman 2001).



Offene Frage: Kann Information aus der Kanten-Betweenness-Berechnung für die nächste Iteration wiederverwendet werden (Berechnung auf dynamischen Graphen, bei denen nur Kanten entfernt werden), um die Gesamtlaufzeit zu verringern?

Analog können wir Closeness- und andere Zentralitäten auf Kanten übertragen. Die Definitionen von Wegen und Abständen seien dazu in naheliegender Weise auf Kanten verallgemeinert.

2.71 Definition (Kanten-Closeness-, -Einfluss-, und -Eigenvektor-Zentralität)
Die Erweiterungen der Closeness-, Einfluss- und Eigenvektor-Zentralität auf Kanten sind definiert durch

$$c_C(G)_e = \frac{1}{\sum_{e' \in E} d_G(e, e')}$$

$$c_I(G)_{(u,v)} = \sum_{(v,w) \in E} \alpha \cdot (1 + c_I(G)_{(v,w)})$$

$$c_E(G)_{(u,v)} = \frac{1}{\lambda} \cdot \sum_{(v,w) \in E} c_E(G)_{(v,w)}$$

für alle $G = (V, E) \in \mathcal{S}$ (für $c_I(G)$ sogar $G \in \mathcal{G}$) und geeignete α, λ .

Wir zeigen die Wohldefiniertheit allgemeiner.

2.72 Definition (Kantengraph)

Zu einem Multigraphen $G = (V, E)$ ist der zugehörige Kanten(multi)graph $\mathcal{E}(G) = (E, F)$ definiert durch

$$((u, v), (v, w)) \in_k F \iff (u, v) \in_r E, (v, w) \in_s E \text{ und } k = r \cdot s$$

2.73 Lemma

Seien $G = (V, E)$ ein gerichteter Multigraph und $\mathcal{E}(G) = (E, F)$ der zugehörige Kantengraph. Dann gilt:

- (i) G gewöhnlich $\implies \mathcal{E}(G)$ gewöhnlich
- (ii) G schleifenfrei $\iff \mathcal{E}(G)$ schleifenfrei
- (iii) $\mathcal{E}(G)$ stark zusammenhängend $\iff G$ besteht aus einer starken Zusammenhangskomponente und einer beliebigen Anzahl isolierter Knoten
- (iv) G stark zusammenhängend und nicht bipartit $\implies \mathcal{E}(G)$ stark zusammenhängend und nicht bipartit

■ **Beweis:** Die beiden ersten Eigenschaften sind offensichtlich.

Für die dritte sei G stark zusammenhängend. Es gibt dann zwischen je zwei Knoten einen gerichteten Weg. Insbesondere gibt es für je zwei Kanten $(u, v), (w, x)$ einen gerichteten Weg von v nach w und einen von x nach u . Diese entsprechen in $\mathcal{E}(G)$ gerichteten Wegen von (u, v) nach (w, x) und von (w, x) nach (u, v) , sodass auch $\mathcal{E}(G)$ stark zusammenhängend ist. Bei Hinzufügen von isolierten Knoten zu G bleibt $\mathcal{E}(G)$ unverändert.

Ist umgekehrt $\mathcal{E}(G)$ stark zusammenhängend, so finden wir für jeden nicht isolierten Knoten in G eine inzidente Kante. Zu zwei Knoten in G betrachte daher zwei inzidente Kanten, etwa (u, v) und (w, x) . Mit der gleichen Überlegung wie oben folgt aus dem starken Zusammenhang von $\mathcal{E}(G)$ die Existenz zweier gerichteter Wege in G , von denen der eine mit (u, v) beginnt und mit (w, x) endet, und der andere umgekehrt.

Schließlich sei G stark zusammenhängend und nicht bipartit. Angenommen, $\mathcal{E}(G)$ wäre bipartit, also insbesondere schleifenfrei, und $E = E_1 \cup E_2$ eine entsprechende Partition der Kanten von G . Ist $v \in V$ ein beliebiger Knoten, dann gibt es, weil G stark zusammenhängend ist, mindestens eine eingehende und eine ausgehende inzidente Kante. Gibt es eine eingehende Kante $(u, v) \in E_1$, dann sind alle ausgehenden Kanten $(v, w) \in E_2$ und damit auch alle eingehenden Kanten $(u', v) \in E_1$. Entsprechend für ein $(u, v) \in E_2$. Die Knoten von G können folglich danach aufgeteilt werden, ob ihre eingehenden oder ausgehenden Kanten in E_1 sind. Dann wäre G aber bipartit, denn keine zwei aus einer der beiden Mengen können adjazent sein. \square

Ein Knotenzentralitätsindex angewandt auf den Kantengraph liefert eine Bewertung der Kanten im Ausgangsgraphen, die als Kanten-Zentralität gedeutet werden könnte. Wegen der Eigenschaften (iii) und (iv) im obigen Lemma ist die Erweiterung der nur auf eingeschränkten Klassen definierten Maße unproblematisch.

2.74 Definition (Zentralitäten)

Sei c eine Knotenzentralität und G ein Multigraph so, dass $c(\mathcal{E}(G))$ definiert ist. Dann heißt der Strukturindex \bar{c} mit

$$\bar{c}(G)_v = c(G)_v$$

und

$$\bar{c}(G)_e = c(\mathcal{E}(G))_e$$

die zugehörige (allgemeine) Zentralität.

2.75 Satz

$$c_C = \bar{c}_C, \quad c_I = \bar{c}_I, \quad c_E = \bar{c}_E .$$

■ **Beweis:** Die Definitionen der Kanten-Zentralitäten entsprechen genau denen für Knoten-Zentralitäten angewandt auf den Kantengraphen. \square

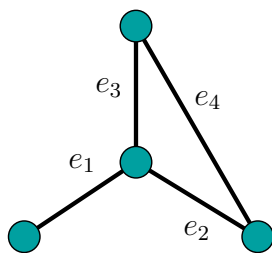
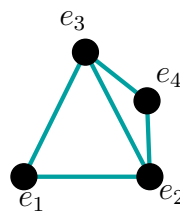
Für Closeness, Einfluss und Eigenvektor-Zentralität können die Algorithmen für die entsprechenden Knoten-Zentralitäten mit wenigen Modifikationen benutzt werden, um die Kanten-Zentralitäten ohne Konstruktion des Kantengraphen zu bestimmen.

Bei Betweenness führt die Definition über den Kantengraphen allerdings zu Inkonsistenzen.

2.76 Satz

c_B und \bar{c}_B sind verschieden.

■ **Beweis:** Es gibt sogar Unterschiede auf zusammenhängenden ungerichteten Graphen:

Graph G Kantengraph $\mathcal{E}(G)$

Es gilt $c_B(e_1) = 6 > c_B(e_2) = c_B(e_3) = 4 > 2 = c_B(e_4)$, aber $\bar{c}_B(e_1) = \bar{c}_B(e_4) = 0 < 1 = \bar{c}_B(e_2) = \bar{c}_B(e_3)$. \square

Man könnte das Problem darin vermuten, dass nur innere Knoten in die Knoten-Betweenness auf dem Kantengraph eingehen, weil die jeweils erste und letzte Kante aller kürzesten Wege im Ausgangsgraphen dadurch nicht berücksichtigt werden. Aus dem vorstehenden Beweis können wir aber ableiten, dass auch eine entsprechende Modifikation der Knoten-Betweenness keine Abhilfe schaffen kann.

2.77 Folgerung

Für jede Knoten-Zentralität c ist \bar{c} verschieden von c_B .

■ **Beweis:** In dem Beispiel im obigen Beweis sind die Knoten e_1 und e_4 von $\mathcal{E}(G)$ strukturell äquivalent, sodass für jeden Knotenstrukturindex c gelten muss $c(e_1) = c(e_4)$ und $c(e_2) = c(e_3)$. Es gilt aber auch $c_B(e_1) \neq c_B(e_4)$. \square