

# Visualisierung von vernetztem Wissen als Graph im virtuellen Raum

Zwischenpräsentation

Bearbeiter:	Sonja Stefani
Aufgabensteller:	Prof. Dr. Helmut Krcmar
Betreuer:	Dimitri Vorona, Dr. Matthias Baume
Abgabedatum:	15.03.2018

Technische Universität München

Garching bei München, 05. Februar 2018

# Gliederung

1. Thema und Motivation
2. Aufbau und Komplexität von Wissensnetzwerken
3. Übersichtliche Darstellung von Graphen
4. Informationen im virtuellen Raum
5. Gegenüberstellung von Algorithmen zur Visualisierung von Graphen

# Thema und Motivation

Erstellen einer VR-Software zur Darstellung von Wissensplattformen

Unterstützen des Lernens

Modellierung nach Konstruktivismus

Idee des selbstinitiierten Lernprozesses [Terhart, 2009]

# Aufbau und Komplexität von Wissensnetzwerken

Dichte Vernetzung vieler Artikel und anderer Seiten

Richtungen der Verlinkungen

Hohe Anzahl an einzelnen Artikeln und daraus resultierender Knoten im Graphen

Aufbau einzelner Seiten aus verschiedenen Elementen, wie

- Texte
- Bilder
- Audiodateien
- Videos

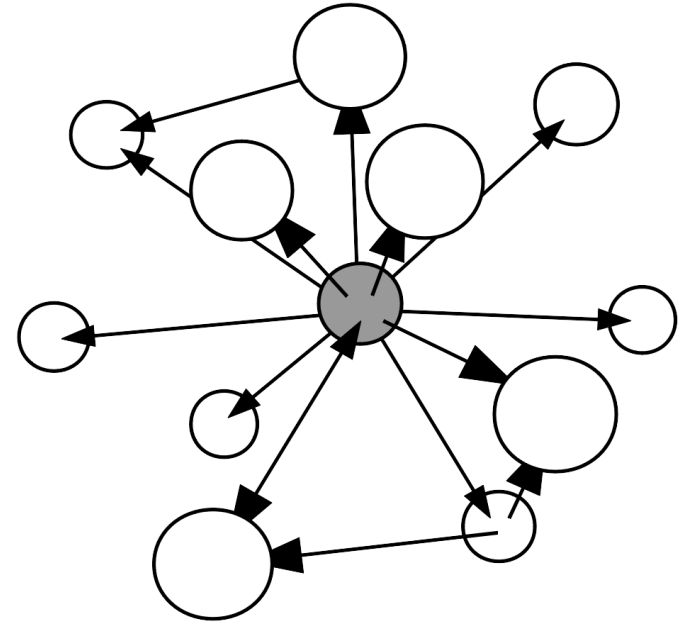
# Übersichtliche Darstellung von Graphen

Reduzierung der Komplexität zur Verbesserung der Übersichtlichkeit [Herman et al., 2000]

Darstellung eines kleineren Ausschnitts des gesamten Graphen mit Start bei einem ausgewählten Knoten

Beschränken der dargestellten Knoten und Kanten durch zwei Parameter:

- Anzahl der maximal dargestellten Knoten
- Wert der maximalen Entfernung vom Startknoten



# Informationen im virtuellen Raum

Sicherstellen der Zugänglichkeit von dargestellten Informationen

Mitbeachtung der Steuerung im virtuellen Raum

Anpassen der einzelnen Teilelemente eines Artikels an die Darstellung in VR

- Steuerung mit dreidimensionalem Cursor
- Behandeln von Elementen wie Text oder Bild als getrennte Einheiten
- Fokus auch vereinfachte Steuerung und Übersichtlichkeit

# Gegenüberstellung von Algorithmen zur Visualisierung von Graphen

Problem der effizienten Berechnung einer akzeptablen Visualisierung eines Graphen

Anpassen eines geeigneten Algorithmus an eine dreidimensionale Umgebung

Grundidee

- Behandeln des Graphen als Kombination aus Ringen (Knoten) und Federn (Kanten)
- Verwenden der simulierten Kräfte an Knoten zur Finden einer übersichtlichen Darstellung
- Berechnen eines Equilibriums

# Gegenüberstellung von Algorithmen zur Visualisierung von Graphen

*Kamada, Kawai (1989)*

- Minimieren des Ungleichgewichts
- Lösen von Differentialgleichungen
- Ursprüngliche Komplexität  $O(n^3)$

*Gajer, Goodrich, Kobourov (2000)*

- Erstellen von Filtrierung
- Iterieren über Ebenen des Graphen
- Gute Performanz für große Graphen

*Davidson, Harel (1996)*

- Gedanke des „simulated annealing“, nicht Federn und Ringe
- „Abkühlen“ der Knoten beim Iterieren
- Schlechte Performanz



# Gegenüberstellung von Algorithmen zur Visualisierung von Graphen

*Fruchterman, Reingold (1991)*

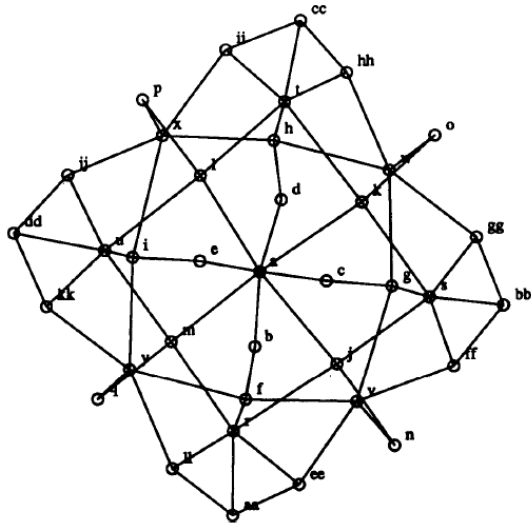
- Basierend auf Grundgedanken von Federn und Ringen
- Nicht physikalisch korrekt
- Ursprüngliche Laufzeit  $O(|V|^2 + |E|)$
- Verbesserung auf  $O(|V| + |E|)$  für gleichmäßig verteilte Knoten

Grober Aufbau des Codes

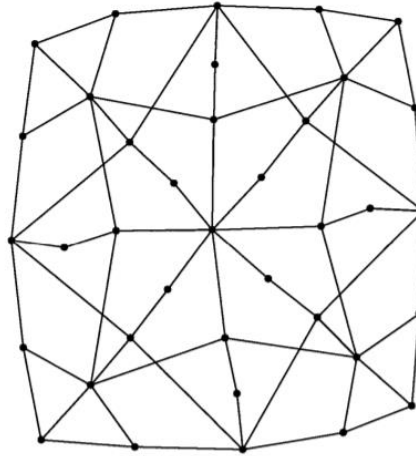
Eingabe: Graph  $G := (V, E)$

- Berechne  $fläche := W * L * H$
- Berechne  $k := \sqrt[3]{fläche/|V|}$
- Lege fest  $f_a(x)$  und  $f_r(x)$  in Abhängigkeit von  $k$
- Für alle  $v \in V$ :
  - Berechne abstoßende Kräfte mit  $f_r$
  - Berechne anziehende Kräfte mit  $f_a$
  - Berechne neue Position in Abhängigkeit von Temperatur  $t$

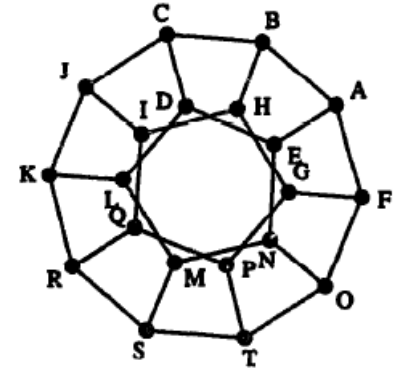
# Gegenüberstellung von Algorithmen zur Visualisierung von Graphen



*Fruchterman, Reingold (1991)*



*Davidson, Harel (1996)*



*Kamada, Kawai (1989)*

# Quellen

Ron Davidson and David Harel. Drawing graphs nicely using simulated annealing. *ACM Trans. Graph.*, 15(4):301-331, October 1996.

Pawel Gajer, Michael T Goodrich, and Stephen G Kobourov. A fast multidimensional algorithm for drawing large graphs. In *Graph Drawing'00 Conference Proceedings*, pages 211-221, 2000.

Thomas M. J. Fruchterman and Edward M. Reingold. Graph drawing by force-directed placement. *Softw. Pract. Exper.*, 21(11):1129-1164, November 1991.

Tomihisa Kamada and Satoru Kawai. An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31(1):7-15, 1989.

# Quellen

Ivan Herman, Guy Melancon, and M Scott Marshall. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on visualization and computer graphics*, 6(1):24-43, 2000.

Ewald Terhart. *Allgemeine Didaktik: Traditionen, Neuanfänge, Herausforderungen*, pages 13-34. VS Verlag für Sozialwissenschaften, Wiesbaden, 2009.

Gerstenmaier, Jochen, and Heinz Mandl. Wissenserwerb unter konstruktivistischer Perspektive. *Zeitschrift für Pädagogik* 41.6 (1995): 867-888.

Dubs, Rolf. Konstruktivismus: einige Überlegungen aus der Sicht der Unterrichtsgestaltung. *Zeitschrift für Pädagogik* 41.6 (1995): 889-903.

# Quellen

Wikipedia. <https://www.wikipedia.org>. Aufgerufen: 04.02.2018.

# Weiterführend – Konstruktivismus

Konstruktion von Wissen

Wissen wird nicht von außerhalb vermittelt

Kritik an klassischem Erwerb von Wissen (z.B. Vortrag) [Gerstenmaier, Mandl, 1995]

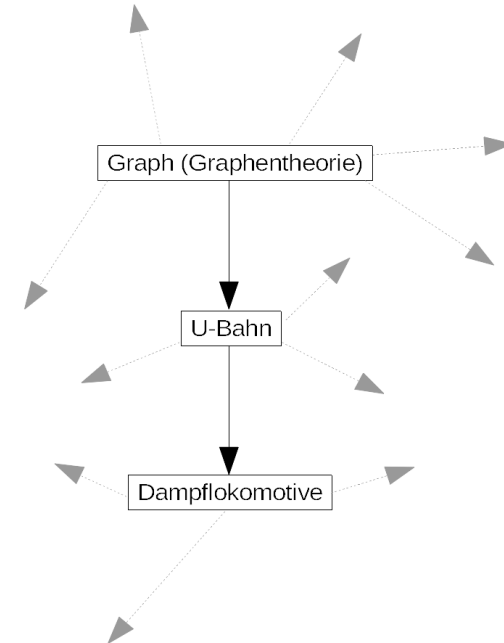
Aktive Beteiligung des Lernenden [Dubs, 1995]

# Weiterführend – Wikipedia als Wissensnetzwerk

Sehr hohe Zahl an Artikeln (2.150.336  
enzyklopädische Artikel, 6.046.787 Seiten  
insgesamt)

Hohe Anzahl an Verlinkungen (Beispiel: Artikel  
„Graph (Graphentheorie)“ mit 72 ausgehenden Links  
und 326 ankommenden Links, davon 26 ein- und  
ausgehend)

Erreichen von irrelevanten Artikeln bereits bei  
geringer Entfernung vom Startknoten (Beispiel:  
Artikel „Graph (Graphentheorie)“ führt in zwei  
Schritten zu Artikel „Dampflokomotive“)



# Weiterführend – Fruchterman Reingold

```

area := W * L; { W and L are the width and length of the frame }
G := (V, E); { the vertices are assigned random initial positions }
k := sqrt(area/|V|);
function fr(z) := begin return x2/k end;
function fa(z) := begin return k2/z end;

for i := 1 to iterations do begin
  { calculate repulsive forces }
  for v in V do begin
    { each vertex has two vectors: .pos and .disp }
    v.disp := 0;
    for u in V do
      if (u # v) then begin
        { Δ is short hand for the difference }
        { vector between the positions of the two vertices }
        Δ := v.pos - u.pos;
        v.disp := v.disp + (Δ / |Δ|) * fr(|Δ|)
      end
    end
  end
end

```

```

end
{ calculate attractive forces }
for e in E do begin
  { each edge is an ordered pair of vertices .v and .u }
  Δ := e.v.pos - e.u.pos
  e.v.disp := e.v.disp - (Δ / |Δ|) * fa(|Δ|);
  e.u.disp := e.u.disp + (Δ / |Δ|) * fa(|Δ|)
end

{ limit the maximum displacement to the temperature t }
{ and then prevent from being displaced outside frame }
for v in V do begin
  v.pos := v.pos + (v.disp / |v.disp|) * min(|v.disp|, t);
  v.pos.x := min(W/2, max(-W/2, v.pos.x));
  v.pos.y := min(L/2, max(-L/2, v.pos.y))
end
{ reduce the temperature as the layout approaches a better configuration }
t := cool(t)
end

```



# Weiterführend – Fruchterman Reingold

Notwendigkeit von Anpassungen zur Visualisierung von Graphen in 3D

- $area := W * L * H$
- $k := \sqrt[3]{area/|V|}$
- $v.pos.z := \min\left(\frac{H}{2}, \max\left(-\frac{H}{2}, v.pos.z\right)\right)$

Abkühlen der Knoten bei mehrfacher Iteration durch die Funktion „cool(t)“

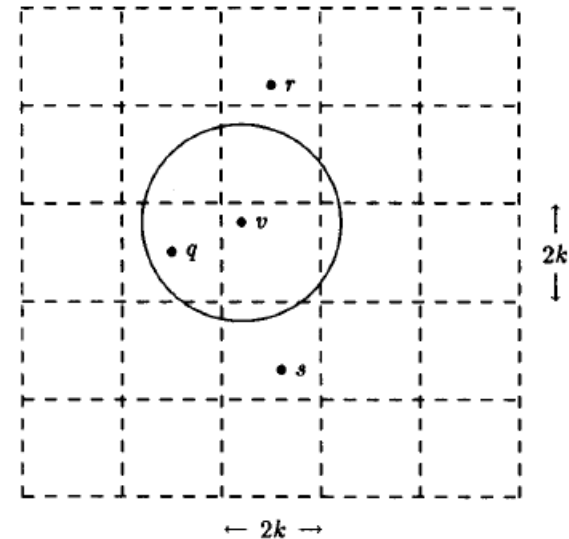
# Weiterführend – Fruchterman Reingold „Grid-Based“

Vermeidung von Berechnen der abstoßenden Kräfte  
von allen gegebenen Knoten

Reduzierung der Berechnung auf Knoten in  
Nachbarschaft des betrachteten Knoten

Verwendung angepasster Formel zur Berechnung  
der abstoßenden Kräfte

- $f_r = \frac{k^2}{d} * u(2k - d)$
- $u = \begin{cases} 1, & \text{falls } x > 0 \\ 0 & \text{sonst} \end{cases}$



# Weiterführend – Programmkonzept

- Starte bei zufälligem oder ausgewähltem Artikel auf Wikipedia
- Zeichne Artikel und Nachbarn je nach Parameter als Graph
- Filtere dabei unwichtige Knoten je nach Parameter heraus
  - Verwende Pageview API um wenig besuchte Artikel herauszufiltern
- Gib dem Benutzer die Möglichkeit, den aktiven Artikel „auszuklappen“ und anzusehen oder einen neuen Artikel auszuwählen
- Gib dem Benutzer die Möglichkeit, den Graphen zu drehen
- Bei Auswahl eines neuen Artikels, wende erneut Visualisierungsalgorithmus auf den neuen Teilgraphen an
- Bei Ausklappen des Artikels, wende Konzepte zur übersichtlichen Darstellung von Informationen an