



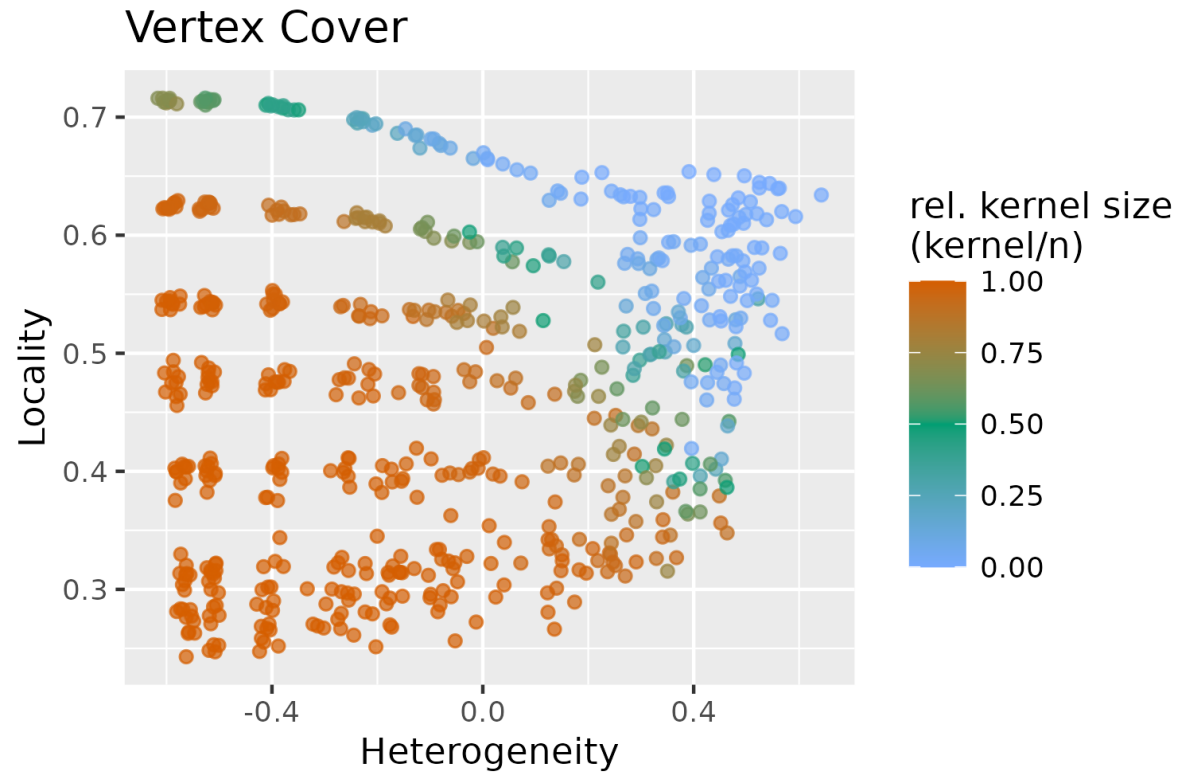
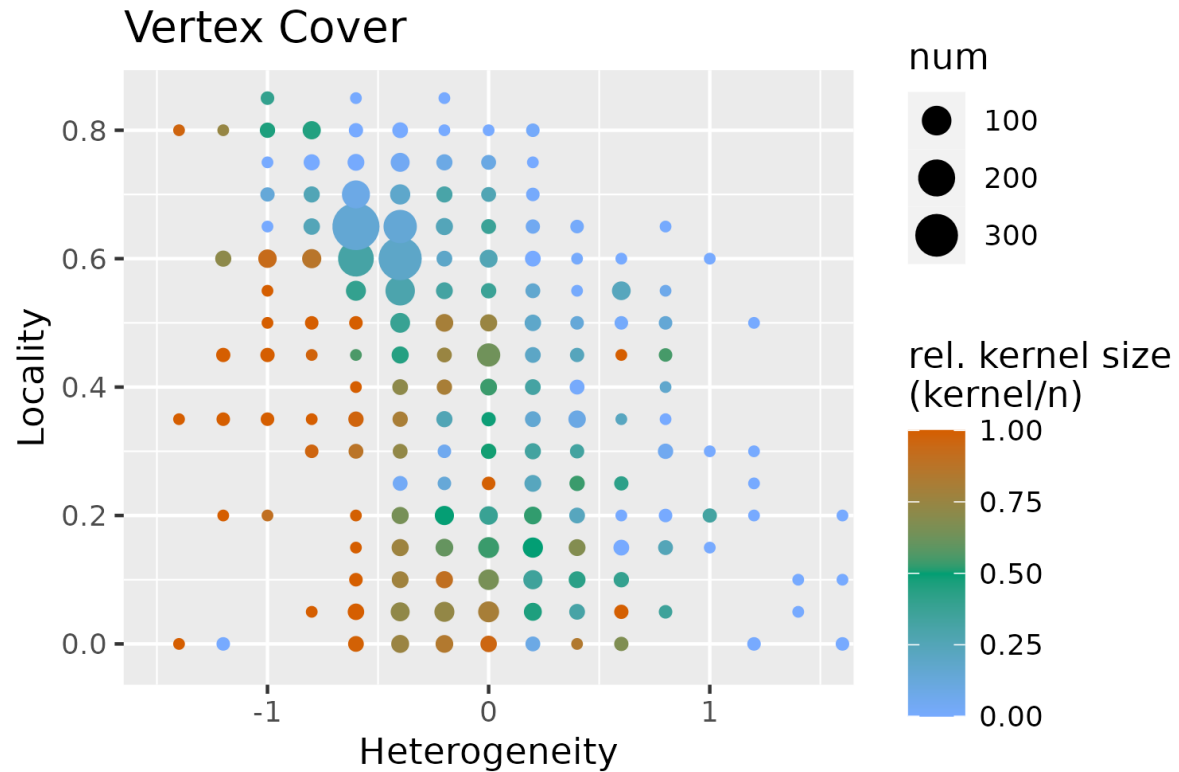
Praktikum – Beating the Worst Case

Jean-Pierre von der Heydt und Marcus Wilhelm | 13.11.2023



Vorstellung Übungsblatt 2

Lösung Übungsblatt 2



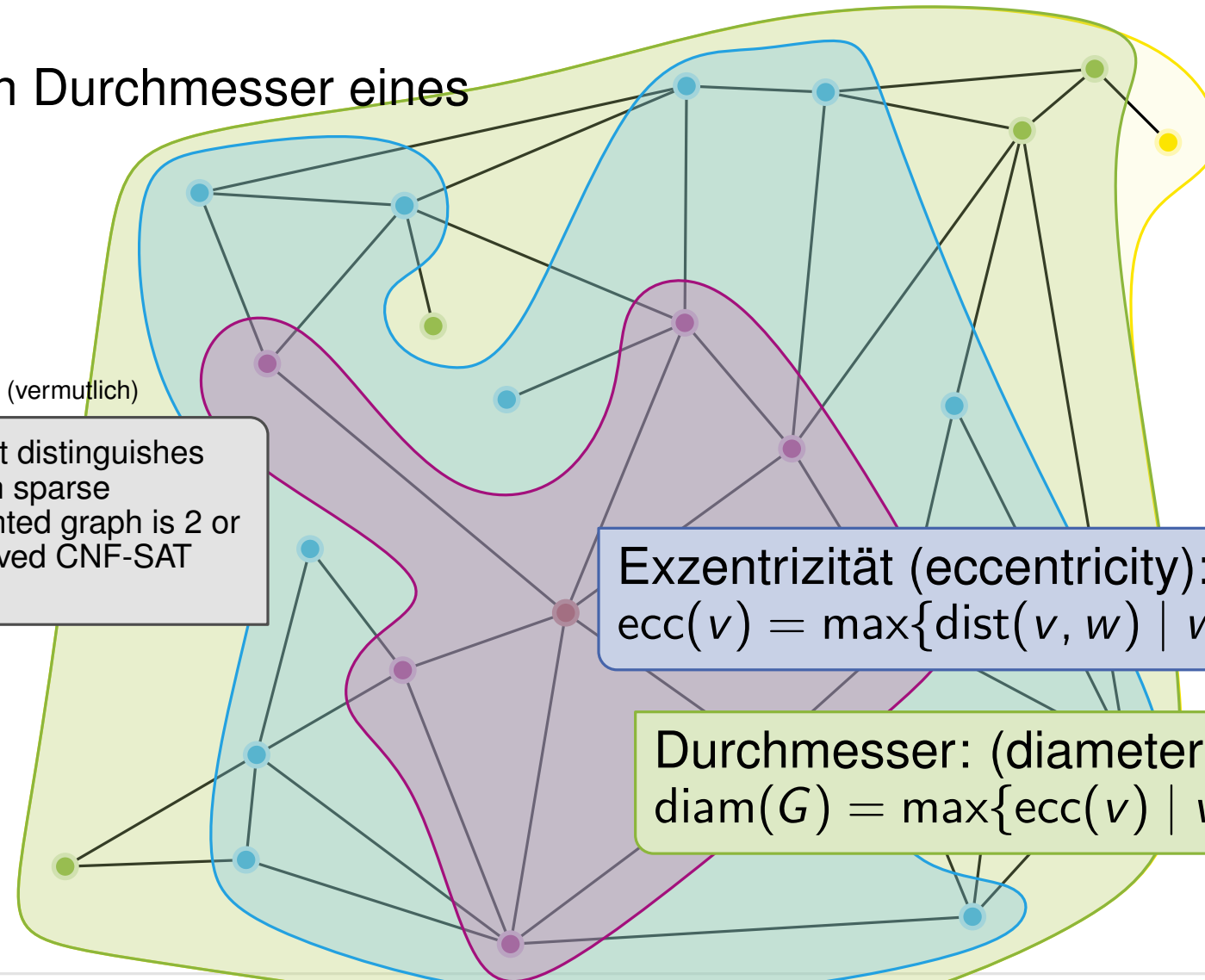
Exzentrizität und Durchmesser

Wie berechnet man den Durchmesser eines Graphen?

- naiv: n mal BFS
- geht es besser?
- nein:

“(vermutlich) any $O(n^{2-\epsilon})$ time algorithm that distinguishes whether the diameter of a given sparse ($m = O(n)$) undirected unweighted graph is 2 or at least 3 would imply an improved CNF-SAT algorithm [Liam, Williams – STOC'13]

(vermutlich)

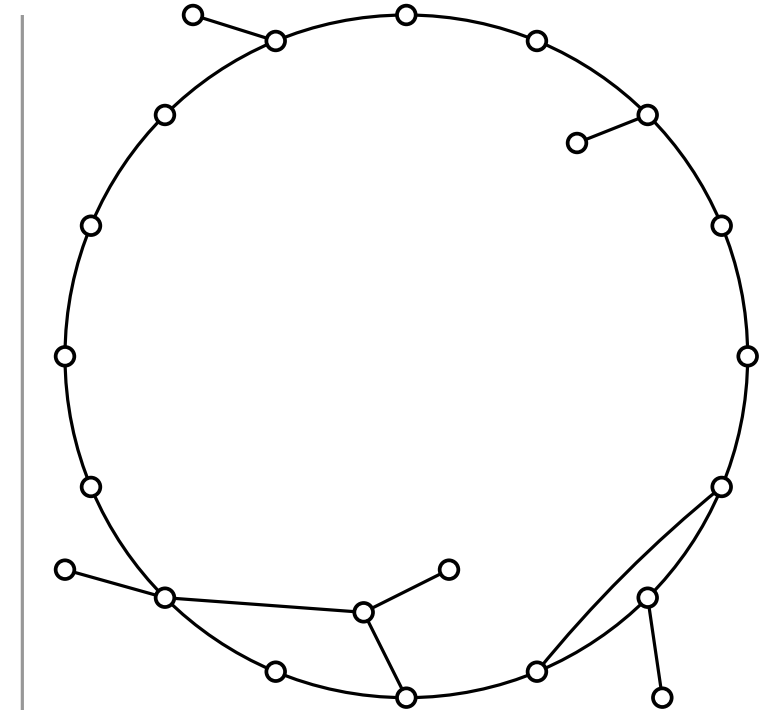
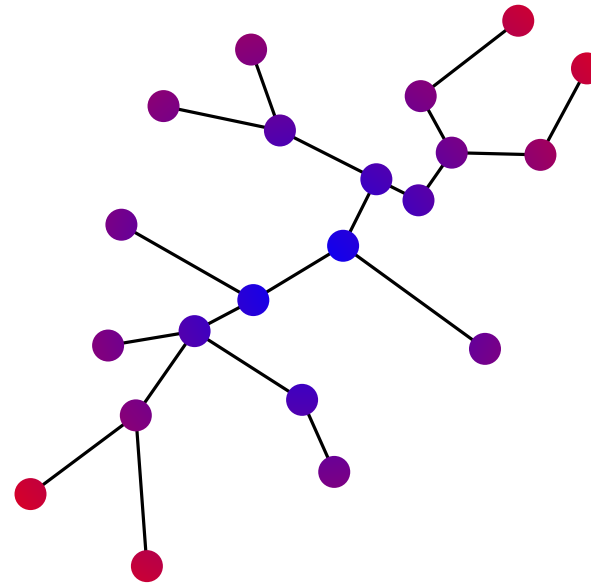
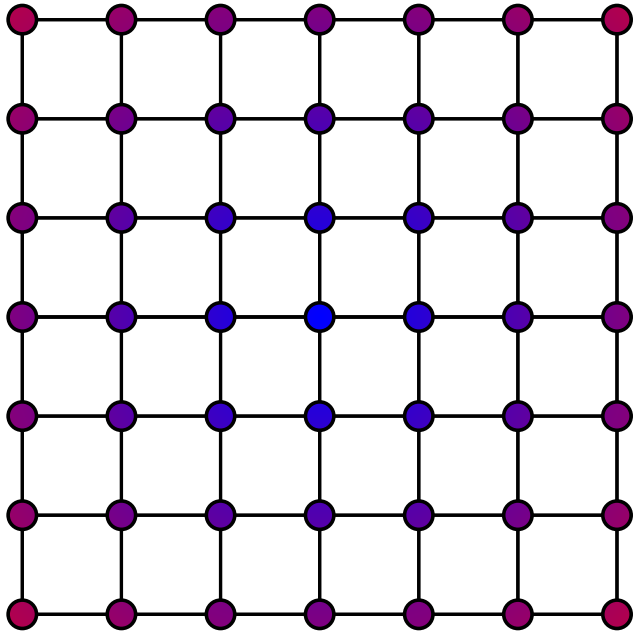


Exzentrizität (eccentricity):
 $\text{ecc}(v) = \max\{\text{dist}(v, w) \mid w \in V\}$

Durchmesser: (diameter)
 $\text{diam}(G) = \max\{\text{ecc}(v) \mid v \in V\}$



Algorithmen für nette Eingaben



Idee: mache BFS nur von besonders dezentralen Knoten

Graphen mit Torus Geometrie:
zu schwer ;-)

iFUB Algorithmus

(iterative fringe upper bound, [Crescenzi et al. 2013])

iFUB Algorithmus

- Wähle „zentralen“ Knoten r
- Breitensuche von r , Layer $L_0, L_1, \dots, L_{\text{ecc}(r)}$
- Sei $\langle v_0, \dots, v_{n-1} \rangle$ Reihenfolge der Knoten im Baum von unten nach oben, $V_i = \{v_0, \dots, v_i\}$
- Berechne $\text{ecc}(v_i)$ mittels BFS

- Untere Schranke

$$\text{diam}(G) \geq \max\{\text{ecc}(v) \mid v \in V_i\}$$

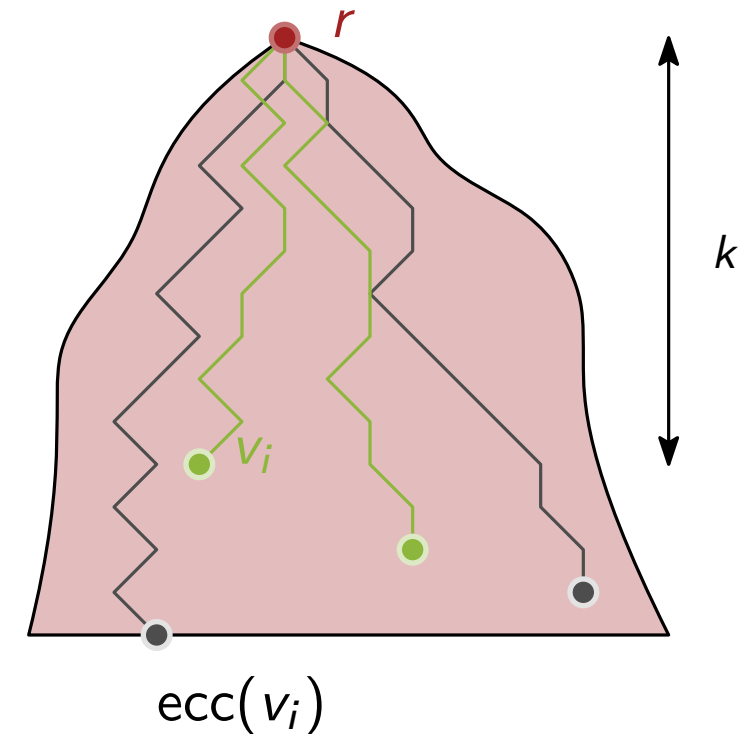
- Obere Schranke an: $\text{ecc}(v_i), \text{diam}(G)$

- sei v_i in Layer L_k , dann:

$$\text{ecc}(v_i) \leq \max(\underbrace{2k}_{\text{(schrumpft)}}, \underbrace{\max\{\text{ecc}(v) \mid v \in V_{i-1}\}}_{\text{wächst}})$$

$$\text{diam}(G) \leq \max(2k, \max\{\text{ecc}(v) \mid v \in V_i\})$$

- Abbruch sobald $2k \leq \max\{\text{ecc}(v) \mid v \in V_i\}$



Kann $\text{diam}(G) > 2 \cdot \text{ecc}(r)$ sein?

Kann $\text{ecc}(v_i) > 2 \cdot k$ sein?

iFUB Algorithmus

- Wähle „zentralen“ Knoten r
- Breitensuche von r , Layer $L_0, L_1, \dots, L_{\text{ecc}(r)}$
- Sei $\langle v_0, \dots, v_{n-1} \rangle$ Reihenfolge der Knoten im Baum von unten nach oben, $V_i = \{v_0, \dots, v_i\}$

- Berechne $\text{ecc}(v_i)$ mittels BFS

- Untere Schranke

$$\text{diam}(G) \geq \max\{\text{ecc}(v) \mid v \in V_i\}$$

- Obere Schranke an: $\text{ecc}(v_i), \text{diam}(G)$

- sei v_i in Layer L_k , dann:

$$\text{ecc}(v_i) \leq \max(\underbrace{2k}_{\text{(schrumpft)}}, \underbrace{\max\{\text{ecc}(v) \mid v \in V_{i-1}\}}_{\text{wächst}})$$

$$\text{diam}(G) \leq \max(2k, \max\{\text{ecc}(v) \mid v \in V_i\})$$

- Abbruch sobald $2k \leq \max\{\text{ecc}(v) \mid v \in V_i\}$

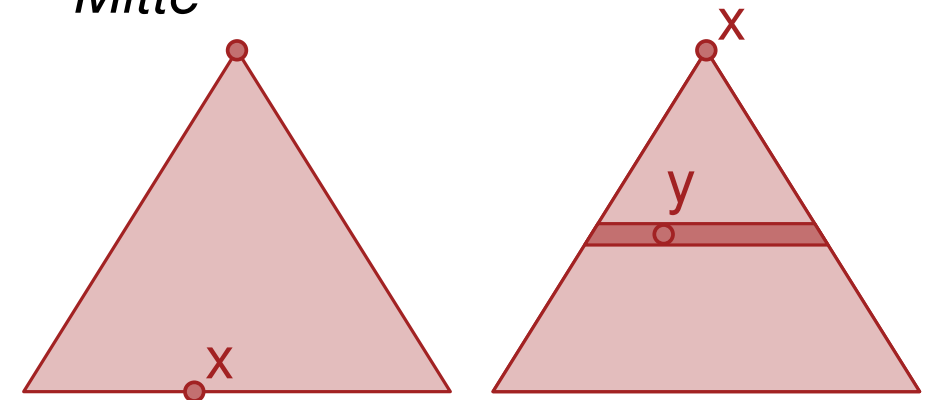
Frage: wie wählen wir r ?

- höchster Grad (Heterogenität)
- mittlerer Knoten (Lokalität)

ggf. iterieren

2-sweep
4-sweep

- BFS: Finde Knoten x am *Rand*
 - BFS von x : Finde Knoten in der *Mitte*



Übungsblatt 3 und Projekt

Übungsblatt 3

- Code und Workflow optimieren / aufräumen
- Performance von iFUB auf realistischen Eingaben untersuchen
- Zeitrahmen: nur *eine* Woche

Anschließend: Projekt (6 Wochen)

- Ziel: Forschungsfrage untersuchen und beantworten
- Inspiration:
 - On the External Validity of Average-Case Analyses of Graph Algorithms [B., F. 2022]
 - Deterministic Performance Guarantees for Bidirectional BFS on Real-World Networks [B., W. 2022]
 - Algorithmen verstehen / verbessern (z.B. diameter, schwere Probleme, ...)

