



# Übung zu Automatisches Zeichnen von Graphen

## Übungsblatt 3

Ausgabe: 13.11.2007 — Besprechung: 27.11.2007

Bei Fragen zur Übung wenden Sie sich bitte an Bernd Zey (Email: [Bernd.Zey@uni-dortmund.de](mailto:Bernd.Zey@uni-dortmund.de)) oder an Hoi-Ming Wong (Email: [hoi-ming.wong@cs.uni-dortmund.de](mailto:hoi-ming.wong@cs.uni-dortmund.de)). Bei Fragen zu OGDF eignet sich auch das Forum unter <http://www.ogdf.net/forum>.

### Aufgabe 1: Theoretische Aufgabe zu Barycenter und Median

- (a) Gegeben ist ein bipartiter Graph  $G$ , der auf 2 Schichten kreuzungsfrei (geradlinig) gezeichnet werden kann. Zeigen Sie, dass das Ergebnis der in der Vorlesung besprochenen Heuristiken *Barycenter* und *Median* eine Einbettung ohne Kreuzungen ist.
- (b) Für manche Anwendungen enthält der Graph wichtige und weniger wichtige Kanten. Dies führt zum gewichteten Kreuzungsminimierungsproblem: Gegeben ist ein bipartiter Graph  $G = (V_1, V_2, E)$  mit Kantengewichten  $w_e$  und eine fixe Permutation der Knotenmenge  $V_1$ . Bestimme eine Permutation der Knotenmenge  $V_2$ , so dass die gewichtete Kreuzungszahl minimiert wird. Diskutieren Sie den 3-Approximationsalgorithmus von *Cakiroglu, Erten, Karatas, Sözdinler: Crossing Minimization in Weighted Bipartite Graphs, Workshop on Experimental Algorithms 2007 (WEA), Lecture Notes in Computer Science 4525, 122-135, Springer-Verlag, 2007*. Wie sind die experimentellen Resultate im Vergleich zu anderen Heuristiken?

### Aufgabe 2: Sugiyama-Verfahren für große Graphen

- (a) Evaluieren Sie das in OGDF enthaltene Sugiyama-Verfahren als Eignung für große Graphen (mehr als 1000 Knoten): Wo bleibt die Laufzeit? (Achtung: Welche Optionen sind geeignet?)
- (b) Entwickeln Sie eigene Ideen, wie man das Sugiyama Verfahren abändern könnte, so dass es für große Graphen verwendet werden kann. Welche Qualitätsverluste nehmen Sie dabei in Kauf? Mögliche Ideen finden Sie in der Arbeit von *Eiglsperger, Siebenhaller und Kaufmann: An Efficient Implementation of Sugiyama's Algorithm for Layered Graph Drawing. Journal of Graph Algorithms and Applications (JGAA), vol. 9, nr. 3, 305-325, 2005*.

### Aufgabe 3: Zählen von Kreuzungen

Implementieren Sie die beiden Verfahren basierend auf *Dynamic Programming* und auf *Divide & Conquer* zum Zählen von Kreuzungen in bipartiten Graphen. Evaluieren Sie beide Verfahren experimentell im Vergleich zu den in OGDF enthaltenen Verfahren BJM und Sweep-Line.