

Seminar Algorithmen für planare Graphen

Reinhard Bauer, Marcus Krug, Ignaz Rutter, Dorothea Wagner

Universität Karlsruhe (TH)
Institut für Theoretische Informatik
Lehrstuhl Algorithmik I

24. Oktober 2008

Betreuer

- Ignaz: `rutter@iti.uka.de`, Büro 322
- Reinhard: `rbauer@iti.uka.de`, Büro 307
- Marcus: `krug@iti.uka.de`, Büro 317



Betreuer

- Ignaz: `rutter@iti.uka.de`, Büro 322
- Reinhard: `rbauer@iti.uka.de`, Büro 307
- Marcus: `krug@iti.uka.de`, Büro 317

Vorstellung

- Name, Semester, Studiengang, Seminare
- algorithmische Vorkenntnisse,
z.B. Algorithmentechnik,
Vorlesung Algorithmen für planare Graphen



Ziel der Veranstaltung

- Überblick über Algorithmen für planare Graphen
 - algorithmische Besonderheiten planarer Graphen
 - Auswahl **aktueller** Publikationen
- Vertiefung algorithmischer Techniken
 - Tree-Width, SPQR-Tree, Right-First-Search, PTAS, Dynamic Programming ...
- Schulung Methodischer Kompetenzen
 - selbständiges Erarbeiten eines Themas
 - fachliches Wissen präsentieren
- Schein

- **Teilnahme** an Präsenz-Terminen
- **Kurzvortrag**: Thema vorstellen, ca. 5 Minuten
- **Hauptvortrag**: Ergebnisse erläutern, 40 - 45 Minuten
- **Medien**
 - Projektor, \LaTeX , Powerpoint
 - Tafel & Kreide
- schriftliche **Ausarbeitung**
 - 5 – 10 Seiten in \LaTeX
 - Übung für Studien- und Diplomarbeit

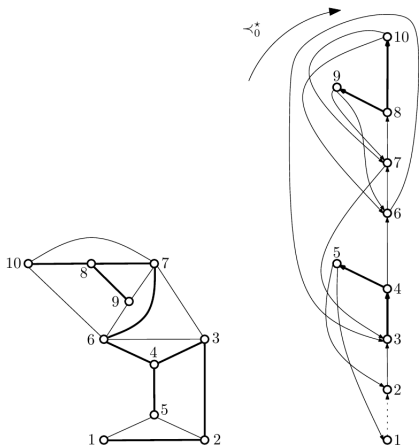
Geplanter Ablauf

Woche	Datum	Termin
43	heute	1. Besprechung
44	Mo 27.10.	15:45-17:15 Einführung Vortragsgestaltung Raum 236
45	Do 06.11.	16:00-19:00 Präsentationsschulung (kww) Raum 131
47	Fr 21.11.	11:30-13:00 Kurzvorträge Raum 301
??	offen	Hauptvorträge
9	01.03.09	Abgabe Ausarbeitungen

- *mindestens zwei Wochen* vor dem Hauptvortrag:
Besprechung der Gliederung mit dem Betreuer
- *mindestens eine Woche* vor dem Hauptvortrag:
Besprechung der Folien mit dem Betreuer
- **Fragen?**

1. Trémaux-Trees and Planarity

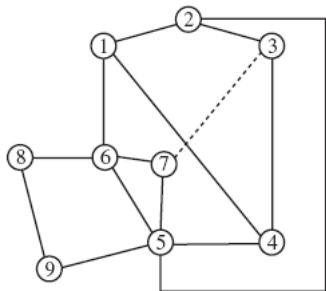
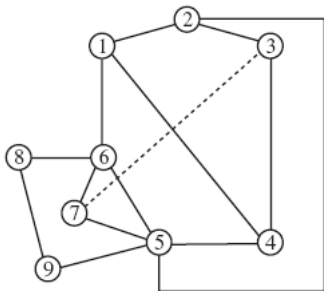
- Linearer Planaritätstest
- Basierend auf Tiefensuche und Färbung



2. Inserting an Edge into a Planar Graph

Gegeben: Planarer Graph $G = (V, E)$ und zusätzliche Kante e .

Gesucht: Kreuzungsminimale Zeichnung von $G' = (V, E \cup \{e\})$ bei der e an allen Kreuzungen beteiligt ist.



3. Edge-Disjoint Paths in Planar Graphs

Problem *Maximum edge-disjoint paths* (MEDP):

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

Menge $\mathcal{T} = \{s_1 t_1, s_2 t_2, \dots, s_k t_k\}$ von Knotenpaaren.

Gesucht: Maximale Anzahl von Paaren in \mathcal{T} , die durch kanten-disjunkte Wege verbunden werden können.

Polylogarithmische Approximation für MEDP unter folgenden Bedingungen:

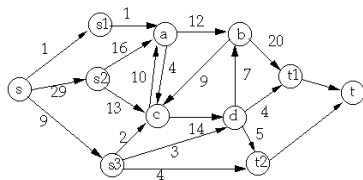
- Graph ist ungerichtet, planar und
- bis zu zwei Pfade dürfen sich eine Kante teilen

4. An $O(n \log n)$ algorithm for maximum st-flow in a directed planar graph

Maximaler st -Fluß in

- gerichteten planaren Graphen
- $O(n \log n)$ Zeit

Basierend auf Right-First-Suche



5. Subgraph Isomorphism in Planar Graphs and Related Problems

Problem SUBGRAPHISOMPRPHISM:

Gegeben Graphen G, H , ist H als Subgraph in G enthalten?

Lineare Laufzeit in $|G|$ für $|H| = O(1)$.

- Partitionierung in Stücke mit kleiner Baumbreite
- dynamische Programmierung

Anwendungen:

- Knoten-/Kantenzusammenhang in planaren Graphen in Linearzeit berechnen
- Lokale kürzeste Wege in planaren Graphen in $O(\log n)$

6. Polynomial-Time Approximation Scheme for Steiner Tree in Planar Graphs

Problem STEINERTREE:

Gegeben: Graph $G = (V, E)$ mit Kantenlängen,
Menge $S \subseteq V$ von Knoten

Gesucht: Zusammenhängender Teilgraph von G , der alle Knoten in S aufspannt und minimale Länge hat.

Resultat:

Approximations-Schema für STEINERTREE auf planaren Graphen mit Laufzeit $O(n \log n)$.

7. A linear-time approximation scheme for planar weighted TSP

TRAVLEING SALES PERSON (TSP)

Gegeben: Menge von Punkten V mit Distanzfunktion

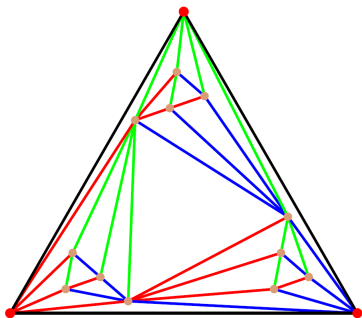
Gesucht: Kürzester Rundweg

ε -optimale Tour in $O(c^{1/\varepsilon^2} n)$ Zeit in der Metrik, die durch die kürzesten Wege in einem planaren Graphen mit nicht-negativen Kantenlängen definiert wird.

8. Greedy drawings of triangulations

Ein *distanz-verringender Pfad* in einem planen Graphen ist ein Pfad $s = v_0, v_1, \dots, v_k = t$ so dass $\|v_i - t\| < \|v_{i-1} - t\|, 2 \leq i \leq k$.

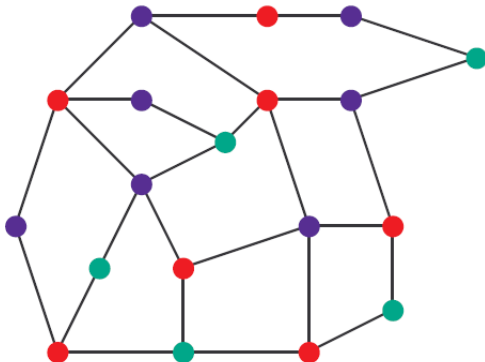
Greedy-Drawing: Zeichnung bei der für jedes Knotenpaar s, t einen distanz-verringenden Pfad besitzt.



9. Fast 3-Coloring of Triangle-Free Planar Graphs

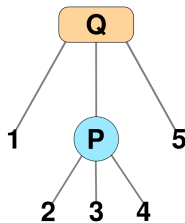
Färbe dreiecksfreie planare Graphen mit 3 Farben in Linearzeit!

- Konstruktiver Existenzbeweis
- Verwendet lokale-kürzeste-Wege-Datenstruktur für Implementierung



10. Planarity Algorithms via PQ-Trees

- Linearer Planaritätstest, basierend auf PQ-Bäumen
- Vereinigung der Ansätze von Shih, Hsu und Boyer, Myrvold



11. Preprocessing an undirected planar network to enable fast approximate distance queries

Vorverarbeitung eines gewichteten planaren Graphen für
 $0 < \varepsilon < 1/10$:

- ε -approximative Distanz zwischen zwei bel. Knoten in $O(\varepsilon^{-1})$ Zeit
- Speicherbedarf $O(\varepsilon^{-1} n \log n)$

- 1 Trémaux Trees and Planarity
 - 2 Inserting an Edge
 - 3 Edge-Disjoint Paths
 - 4 Maximum st-Flow
 - 5 Subgraph Isomorphism
 - 6 PTAS for Steiner Tree
 - 7 PTAS for TSP
 - 8 Greedy Drawings
 - 9 3-coloring
-
- 10 Planarity via PQ-Trees
 - 11 Distance Queries

- **Präsentationsschulung:** Montag, 27.10., 15:45-17:15, SR 236
- **kww-Schulung:** Donnerstag, 06.11., 16:00-19:00, SR 131