Algorithmik
Fachbereich IV – Informatik
Universität Trier

Trier, Juni 2023

Prof. Dr. Stefan Näher

Exercise sheet 4

Visualization of Graphs

Exercise 1 – Canonical Orders for Outerplanar Graphs

A graph is *outerplanar* if it has a planar embedding such that all vertices are on the same face, usually the outer face. It is a *maximal outerplanar graph* if it is internally triangulated.

Describe a special canonical order built precisely for maximal outerplanar graphs.

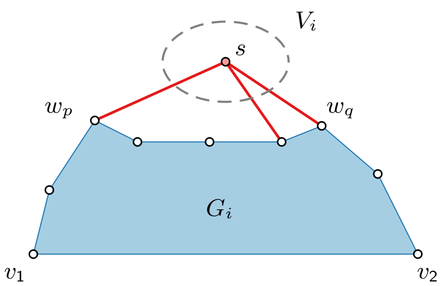
a) Reformulate the conditions (C1)–(C3) for maximal outerplanar graphs. Can we enforce a bound on the degree of v_{k+1}? **2 Points**

b) How can we use the algorithm for maximal planar graphs to obtain a canonical order for maximal outerplanar graphs? **2 Points**

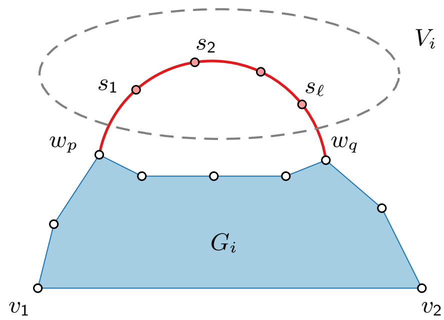
Exercise 2 – Canonical Orders for 3-Connected Planar Graphs

Canonical orders for planar 3-connected graphs are a generalization of canonical orders for plane triangulations. Let G be a 3-connected planar graph. Let $\pi = (V_1, V_2, \ldots, V_K)$ be an ordered partition of $V(G)$. That is, $V_1 \cup V_2 \cup \ldots \cup V_K = V(G)$ and $V_i \cap V_j = \emptyset$ for all $i \neq j$. Define G_i to be the planar subgraph of G induced by $V_1 \cup V_2 \cup \cdots \cup V_i$. Let C_i be the subgraph of G induced by the edges on the boundary of the outer face of G_i. As illustrated below, π is a *canonical order* of G if:

- $V_1 = \{v_1, v_2\}$, where v_1 and v_2 lie on the outer face and $v_1v_2 \in E(G)$.
- $V_K = \{v_n\}$, where v_n lies on the outer face, $v_1v_n \in E(G)$, and $v_n \neq v_2$.
- Each C_i $(i > 1)$ is a cycle containing v_1v_2.
- Each G_i is biconnected and internally 3-connected; that is, removing any two interior vertices of G_i does not disconnect it.
- For each $i \in \{2, 3, \ldots, K-1\}$, one of the following conditions holds:
  - (i) $V_i = \{s\}$ where s is a vertex of C_i with at least two neighbors in C_{i-1}, and s has at least one neighbor in $G \setminus G_i$.
  - (ii) $V_i = (s_1, s_2, \ldots, s_\ell)$, $\ell \geq 2$, is a path in C_i, where each vertex in V_i has at least one neighbor in $G \setminus G_i$. Furthermore, s_1 and s_ℓ have one neighbor in C_{i-1}, and these are the only two edges between V_i and G_{i-1}.



(a)



(b)

a) Suppose that G_i is 3-connected. How can we choose V_i? **1 Point**

b) If G_i contains a vertex of degree two, where is it in G_i? **1 Point**

c) A *separation pair* of a graph G is a pair of vertices $\{a, b\}$ such that $G \setminus \{a, b\}$ consists of two or more components.

Suppose that G_i is 2-connected. Show that for every separation pair $\{a, b\}$ both vertices lie on C_i of G_i. **3 Points**

d) Show that the canonical order described above exists for all planar 3-connected graphs. **5 Points**

Hint: Make a case distinction between whether G_i is 3-connected or 2-connected. In the latter case, consider a minimal separation pair.

Exercise 3 – An Alternative Shift Method

We want to examine an alternative drawing algorithm for planar, embedded, triangulated graphs $G = (V, E)$:

- Let $(v_1, v_2, \ldots, v_n)$ be a canonical order of the vertices.
- Draw v_1 at $(0, 0)$, v_2 at $(2, 0)$, and v_3 at $(1, 1)$.
- Draw the graph incrementally for $k = 4, \ldots, n$. Let $v_1 = w_1, \ldots, w_p, \ldots, w_q, \ldots, w_t = v_2$ be the vertices on the boundary of the outer face of G_{k-1} (in this order), where $w_p, \ldots, w_q$ are the neighbors of v_k in G_k. As the x-coordinate of v_k, choose an integer value $x(v_k)$ with $x(w_p) < x(v_k) < x(w_q)$. If no such value exists, first shift the right part of the drawing to the right by 1; i.e. for $q \leq i \leq t$ move each $L(w_i)$ to the right by 1. Now choose the smallest positive integer y-coordinate for which the drawing stays planar and v_k lies on the outer face.

a) Argue why this algorithm always yields a planar drawing. Why does in step 3 always a suitable y-coordinate exist? **4 Points**

b) Find a good lower bound for the maximum area requirement of the resulting drawing: find an infinite family of graphs where making bad choices for the x-coordinate in step 3 gives huge y-coordinates. **2 Points**