Graphs & Algorithms: Advanced Topics

Things

Prerequisite: basic graph theory and graph algorithms. In particular the material of the course Graphs and Algorithms (Spring 2007)

In graph theoretic notation we mostly follow the book “Introduction to Graph Theory” by Doug West.

Separators — Recap

Let G be a graph. A set $S \subseteq V(G)$ is called an $(f(n), \alpha)$-separator if

- $|S| \leq f(|V(G)|)$ and
- components of $G - S$ are of order $\leq \alpha n$.

Theorem Every tree contains a $(1, \frac{1}{2})$-separator, which can be found in $O(n)$ time.

Divide-and-Conquer method

1. Solve the problem on “very-very small” sets with brute force.
2. Otherwise DIVIDE: Find a “very small” vertex set C “fast” such that $G - C$ falls into two “small” pieces A and B with no edges in between.
3. CONQUER: Explore all solutions restricted to C (brute force) and solve the corresponding subproblems on A and B recursively. Put together the partial solutions.

Here:

“Small” means $< \beta n$, where $\beta < 1$ is a constant.
“very small” means $o(n)$.
“very-very small” means constant.

Outcome: Algorithm with subexponential running time $2^{\text{very small}}$

Separator for planar graphs

Theorem (Lipton-Tarjan, 1979) G is planar with n vertices. Then G has a $(\sqrt{8n}, \frac{2}{3})$-separator, which can be found in $O(n)$-time.

Remark The order $\sqrt{n}$ is best possible for the order $f(n)$ of a separator $(f(n), \alpha)$ with constant $\alpha < 1$.

Remark Sparsity alone is not enough for the existence of a good separator. Most graphs of linear size would not allow a good separator with $f(n) = o(n)$

A spanning tree could help

Lemma Let G be planar graph, and let T be a spanning tree of G with diameter s.
Then a $(s+1, \frac{2}{3})$-separator of G can be found in $O(n)$-time.

Proof of Theorem using Lemma
WLOG G is connected.

Fix $v_0 \in V(G)$ arbitrarily.
Define levels: $L_i := \{v \in V(G) : dist(v, v_0) = i\}$.
Let $l := \max\{i : L_i \neq \emptyset\}$.

If $2l + 1 \leq \sqrt{8n}$, then we are done by Lemma.

Otherwise, let $s := \lceil \sqrt{\frac{n}{2}} \rceil$ and
$S_j := \bigcup \{L_i : i \equiv j \pmod{s}\}$

Remove S_{j_0} with $|S_{j_0}| \leq \lfloor \frac{n}{s} \rfloor \approx \sqrt{2n}$.

Case 1. All components of $G - S_{j_0}$ are of order $\leq \frac{2}{3}n$.
Be happy, you are done.

Case 2. There is one component K, $|K| > \frac{2}{3}n$.

$$K \subseteq \bigcup_{i=j+1}^{j+s-1} L_i \text{ for some } j \equiv j_0 \pmod{s}.$$

Then contract L_j into a vertex in $G[K \cup L_j]$.

The resulting graph H has a spanning tree with diameter $2(s-1)$ so by Lemma we have a $(2s-1, \frac{2}{3})$-separator S_H in H.

Then $S_{j_0} \cup S_H$ is the appropriate separator of G.

Proof of Lemma

Lemma G is a planar graph, T is a spanning tree of G with diameter s. Then a $(s+1, \frac{2}{3})$-separator of G can be found in $O(n)$-time.

Proof. WLOG G is a triangulation. (linear time!)

For $e \in E(G) \setminus E(T)$ there is a unique cycle $C(e)$ in $T + e$.

$n_{int}(C(e))$ is the number of vertices in $\mathtt{Int}(C(e))$.

$n_{ext}(C(e))$ is the number of vertices in $\mathtt{Ext}(C(e))$.

We are looking for an edge e such that both $n_{int}(C(e))$ and $n_{ext}(C(e))$ are $\leq \frac{2}{3}n$.

Proof of Lemma cont'd

Let $e = xy \in E \setminus E(T)$ be arbitrary
Suppose $n_{int}(C(e)) > \frac{2}{3}n$.
Find $e' \in E \setminus E(T)$, s. t.

- $n_{ext}(C(e')) \leq \frac{2}{3}n$.
- $\mathrm{Int}(C(e')) \subset \mathrm{Int}(C(e))$

Let $z \in \mathrm{Int}(C(e))$ be the third vertex in the face F containing e.

Case 1. $zx \in E(T)$.
Choose $e' = zy$. (Note that $zy \notin E(T)$.)

Case 2. $zx, zy \notin E(T)$.
Choose $e' = zx$ if $n_{int}(C(zx)) \geq n_{int}(C(zy))$
Otherwise choose $e' = zy$.

The Algorithm

Input: Plane triangulation G, spanning tree $T \subseteq G$

Output: Edge $e \in E \setminus E(T)$; $C(e)$ is a separator with $n_{ext}(C(e)), n_{int}(C(e)) \leq \frac{2}{3}n$.

$e = xy \in E \setminus E(T)$ arbitrary, with direction.
Run `Clockwise-DFS`(y, x, x) to determine $n_{int}(C(e))$ and $n_{int}(C(e))$.
IF $n_{ext}(C(e)) > \frac{2}{3}n$ THEN
  Update $y := x$, $x := y$ (e changes direction)
IF $n_{int}(C(e)) > \frac{2}{3}n$.
  WHILE $n_{int}(C(e)) > \frac{2}{3}n$ DO
    $z \in C(e) \cup \text{Int}(C(e))$ such that $\{z, x, y\}$ is a face.
    Alternately run `Clockwise-DFS`(z, y, x)
    and `Anticlockwise-DFS`(z, x, y).
    IF `Clockwise-DFS` terminates first THEN
      $n_{int}(C(zx)) \leq n_{int}(C(zy))$; Update $e := zy$.
    ELSE
      Update $e := zx$.
Output e.

Algorithm Clockwise-DFS

Clockwise-DFS(z,y,x)
Input: plane triangulation G, spanning tree $T \subseteq G$, cyclic lists L_v of the neighbors of $v \in V$ in G, those which are also neighbors in T are marked;
root vertex z, reference vertex y, target vertex x.

$u := y$, $v := z$.
WHILE $v \neq x$ DO
    $u := v$,
    $v := T$-neighbor of v coming first after u in L_v
        according to the anticlockwise direction.

Remark The tree produced by `Clockwise-DFS` tends to “bend” in the clockwise direction.

For **Anticlockwise-DFS**: Replace “clockwise” with “anticlockwise”.

Finding Planar independent sets

MAXIMUM (PLANAR) INDEPENDENT SET PROBLEM

Input: (Planar) graph G
Output: Independent set $X \subseteq V$ with maximum cardinality, that is, $|X| = \alpha(G)$.

Theorem The MAXIMUM PLANAR INDEPENDENT SET problem is NP-hard.

Theorem The MAXIMUM PLANAR INDEPENDENT SET problem can be solved in time $2^{O(\sqrt{n})}$.

Remark We don't know whether it is possible to solve the MAXIMUM INDEPENDENT SET problem in time $2^{o(n)}$. In fact, we don't expect that happening.

Algorithm `PlanarIndSet`

Input: Plane graph G

Output: Maximum independent set I

IF $|V(G)| \leq 1$ THEN

$\quad I := V(G)$

ELSE

$\quad I := \emptyset$

$\quad$ Find a $(\sqrt{8|V(G)|}, \frac{2}{3})$-separator C for G.

$\quad$ Let $A \cup B = V \setminus C$ a partition of V such that $|A|, |B| \leq \frac{2}{3}n$, $E(A, B) = \emptyset$.

$\quad$ FOR ALL independent set $S \subseteq C$ DO

$\quad\quad I_A := \texttt{PlanarIndSet}(G[A \setminus N(S)])$

$\quad\quad I_B := \texttt{PlanarIndSet}(G[B \setminus N(S)])$

$\quad\quad$ IF $|S| + |I_A| + |I_B| > |I|$ THEN

$\quad\quad\quad I := S \cup I_A \cup I_B$

output I