Institut for Matematik og Datalogi
Syddansk Universitet

February 25, 2022
JBJ

DM867 – Spring 2022 Weekly Note 5

Stuff covered in week 8

- Non-bipartite Matching. PS Chapter 10.4-10.5 and SCH 5.1-5.2 (We will not cover weighted non-bipartite matching, but you should know that this problem is solvable in polynomial time).
- We also covered the more general f-factors, where we have a graph $G = (V, E)$ and a specification $f(v) \leq d(v)$ at every vertex and we want to select a subset E' of E so that these induce a spanning graph $F = (V, E')$ with $d_F(v) = f(v)$ for each $v \in V$. Such an F is called an f-factor of G. In particular, when $f(v) \equiv k$ for all $v \in V$ we call F a **k-factor** of G.

  To see that this problem can be solved using a matching algorithm, lets create a new graph H from $G = (V, E)$ and f as follows: Replace each vertex v of G with two sets $A(v)$ and $B(v)$ of vertices with $|A(v)| = d(v)$ and $|B(v)| = d(v) - f(v)$. The edges of H consist of all possible edges between $A(v), B(v)$ for all $v \in V$ and for each edge $uw \in E$, put a single edge between $A(u)$ and $A(w)$ such that each vertex of $A(v)$, $v \in V$ belongs to exactly one such edge. Now it is easy to show that H has a perfect matching if and only if G has an f-factor.
  This is a polynomial reduction (remind yourself why!) so we get a polynomial algorithm for checking the existence of an f-factor in a given graph from the polynomial algorithm for the maximum matching problem.

Lectures in Week 9

- Lovász's splitting theorem and augmenting the edge-connectivity of a graph. This is covered by notes at the end this weekly note.
- Arc-disjoint branchings. This is BJG Section 9.5.
- Orientations with degree bounds. BJG Section 8.7 (we will cover pages 446 to 447 top as well as Theorem 8.7.3 and its proof).
- Finding subdigraphs with prescribed in- and out-degrees. BJG Section 3.11.3.
- 2-processor scheduling Schrijver Application 5.2

Problems and applications to discuss on Friday in Week 9

- BJG 3.33, 3.34, 3.35
- Consider Section 3.11.3 in BJG and use this to show that in the case when G is a bipartite graph we can solve the f-factor problem by transforming the problem into a maximum flow problem.
- SCH 5.1, 5.4
- SCH 5.7 page 84.
- PS Problem 5. page 243.
- PS Problem 11. page 245.
- 2-processor scheduling: Suppose we are given a task consisting of 8 jobs all of which take unit time. The jobs are called a, b, c, d, e, f, g, h and have the following precedence relations, where we only list the one that do not follow by transitivity (if x is before y and y before z, then automatically x is before z so we don't write it in the list):

  $$\{a < c, a < f, b < d, c < e, c < g, d < f, f < e, f < g, f < h\}$$

  Find an optimal schedule for processing on 2 processors and prove that it is optimal.
- A matroid $M = (S, \mathcal{F})$ is *connected* if for any non-trivial partition $S = S_1 \cup S_2$ there exist a circuit C of M with $C \cap S_i \neq \emptyset$ for $i = 1, 2$.
  Now consider a graph $G = (V, E)$ and the **graphic** matroid $M = (E, \mathcal{I})$ where $E' \in \mathcal{I}$ if and only if $H = (V, E')$ (the subgraph induced by E') has no cycle. Show that this matroid is connected if and only if G is 2-connected.
- PS Problem 10. page 303.

First set of exam problems
These will probably be posed in Week 11 and must be handed in again Friday April 8th. You are allowed to work in groups of up to 3 persons and different groups may not communicate about the problems.

Notes on Edge-connectivity augmentation

Recall that for a graph $G = (V, E)$ and vertices x, y of G we denote by $\lambda(x, y)$ the maximum number of edge-disjoint (x, y)-paths in G. By Menger's theorem we know that $\lambda(x, y)$ is equal to the minimum degree $d(X)$ of a set $X \subset V$ which contains x but not y.

By a **splitting** off a pair of edges su, sv incident with the same vertex s in a graph G we mean the operation that deletes these two edges and adds a new edge uv (possibly parallel to one or more already existing edges between u and v).

Let $k \geq 2$ be an integer and let $G = (V + s, E)$ be a graph with a special vertex s so that

$$\lambda(x, y) \geq k \text{ for every choice of } x, y \in V \tag{1}$$

We say that a splitting (su, sv) is **feasible** if (1) holds for all $x, y \in V$, where G' is the graph we obtain by the splitting opereation for su, sv.

Recall that following holds for all $X, Y \subseteq V$ (here $d(X, Y)$ denotes the number of edges with one end in $X - Y$ and the other in $Y - X$ and $d(X \cap Y, V + s - (X \cup Y))$ denotes the number of edges between $X \cup Y$ and the complement of $X \cup Y$):

(i) $d(X) + d(Y) = d(X \cap Y) + d(X \cup Y) + 2d(X, Y)$

(ii) $d(X) + d(Y) = d(X - Y) + d(Y - X) + 2d(X \cap Y, V + s - (X \cup Y))$

Theorem 1 *Let $k \geq 2$ be an integer and let $G = (V + s, E)$ be a graph with a special vertex s such that $d(s)$ is possitive and even and (1) holds for G.*

Then for every edge $e = su$ incident with s there exists another edge $f = sv$ incident with s such that the splitting (su, sv) is feasible.

Proof:

Let us call a set $X \subset V$ **dangerous** if $d(X) \leq k + 1$. Observe that if u, v are neighbours of s that lie in the same dangerous set, then the splitting (su, sv) is not feasible since after this, the set X would have degree $k - 1$, implying that $\lambda_{G'}(x, y) \leq k - 1$ for every $x \in X$ and $y \in V - X$. Conversely, if there is no dangerous set containing both u and v then the splitting (su, sv) **is** feasible since (1) still holds after the splitting.

Let $e = su$ be fixed and suppose that there is no other edge sv such that the splitting (su, sv) is feasible. By the remark above, this means that every neighbour $v \neq u$ of s is contained in some dangerous set X_v that also contains u.

Now let $\mathcal{L}$ be a family of maximal dangerous sets (they are not contained in any bigger set which is also dangerous) all of which contain u and so that every other neighbour $v \neq u$ of s is contained in some $X \in \mathcal{L}$. Furthermore assume that $|\mathcal{L}|$ is minimum with that property (so $\mathcal{L}$ contains the minimum number of dangerous sets among all such families). We claim that $\mathcal{L}$ has the following properties:

(a) For all distinct members $X, Y \in \mathcal{L}$ such that $X \cup Y \neq V$ we have $d(X \cap Y) = k$, $d(X \cup Y) = k + 2$ and $d(X, Y) = 0$.

(b) For all distinct members $X, Y \in \mathcal{L}$ we have $d(X - Y) = k = d(Y - X)$ and $d(X \cap Y, V + s - X \cup Y) = 1$.

(c) For every choice of distinct members $X, Y \in \mathcal{L}$ and distinct members $X', Y' \in \mathcal{L}$ we have $X \cap Y = X' \cap Y'$.

It is easy to check that (a) and (b) follow from (i) and (ii). To prove that (c) holds, we argue as follows. Let $W = X \cap Y$ and $W' = X' \cap Y'$ and suppose $W \neq W'$. By (a) both sets have degree k and contain u (all sets in $\mathcal{L}$ contain u) so if both $W - W'$ and $W' - W$ are non-empty, then we would have

$$\begin{aligned} k + k &= d(W) + d(W') \\ &= d(W - W') + d(W' - W) + 2d(W \cap W', V + s - W \cup W') \\ &\geq k + k + 2, \end{aligned}$$

a contradiction. Hence we may assume that $W' \subseteq W$ and we just need to show that also $W \subseteq W'$. If this is not the case, then $\{X', Y'\} \neq \{X, Y\}$ so without loss of generality (and by the minimality fo $\mathcal{L}$ we have $X' - W \neq \emptyset$ and $W - X' \neq \emptyset$ so we can apply (ii) to W and X' and get

$$\begin{aligned} k + (k + 1) &\geq d(W) + d(X') \\ &= d(W - X') + d(X' - W) + 2d(W \cap X', V + s - W \cup X') \\ &\geq k + k + 2, \end{aligned}$$

a contradiction again (here we used that W and X' both contain the vertex u. Thus we have proved (c).

Let $\mathcal{L} = \{X_1, X_2, \ldots, X_r\}$, for some $r \geq 1$. It follows from (c) and (i) that there exists a set $Z \subset V$ such that $u \in Z$, $d(Z) = k$ and $X_i \cap X_j = Z$ for all $1 \leq i < j \leq r$.

Note that we cannot have $r = 1$ because this and the fact that $d(s)$ is even would imply that $d(V - X_1) = d(X_1) - d(s, X_1) + d(s, V - X_1) \leq (k+1) - 2 + 0 = k - 1$, contradicting (1).

Suppose next that $r \geq 3$. We claim that this leads to the conclusion that $d(Z) = 1$, contradicting (1) as $k \geq 2$. First it follows from (b) that there is precisely one edge from

Z to $V+s-(X_1 \cup \ldots \cup X_r)$. Thus it suffices to show that $d(Z, X_i - Z) = 0$ for all $i = 1, 2, \ldots, r$. This follows from (b) applied to two other members X_j, X_h of $\mathcal{L}$, because here (b) implies that $d(X_j \cap X_h, V+s-(X_j \cup X_h)) = 1$ and the edge su is the only edge between $X_j \cup X_h$ and the complement of this set (which contains $X_i - Z$ since Z is the pairwise intersection of every pair of distinct sets in $\mathcal{L}$).

It remains to consider the case $r = 2$. Let $a_i = d(s, X_i - Z)$, $i = 1, 2$, that is, a_i is the number of edges between s and $X_i - Z$. By (b) we see that $d(s, Z) = 1$ so su is the only edge between s and Z. Now it follows from the fact that $d(s)$ is even and $X_1 \cup X_2$ cover all neighbours of s that $a_1 \neq a_2$. Without loss of generality we have $a_1 > a_2$. But then we have that $d(V - X_1) = d(X_1) - d(s, X_1) + d(s, V - X_1) \leq k+1-(a_1+1)+a_2 = k-(a_1-a_2) \leq k-1$, contradicting (1).

Thus in all cases we reached a contradiction to the assumption that there was no neighbour $v \neq u$ of s such that the splitting (su, sv) is feasible. So such a splitting exists and the proof is complete. □

By a **subpartition** of a set S we mean a collection of disjoint subsets of S.

Theorem 2 *Let $k \geq 2$ be an integer and let $H = (V, E')$ be a graph which is not k-edge-connected. Then the minimum number of new edges we need to add to H such that the new graph $H' = (V, E' \cup F)$ is k-edge-connected is $\lceil \frac{\alpha_{H,k}}{2} \rceil$ where*

$$\alpha_{H,k} = \max_{\mathcal{F}} \sum_{X \in \mathcal{F}} (k - d_H(X)) \tag{2}$$

and the maximum is taken over all subpartitions of V.

Proof: Give give a constructive proof due to A. Frank: Let G' be obtained from H by adding a new vertex s and k parallel edges between s and every vertex in V. Clearly G' satisfies (1). Denote the vertices of V by $v_1, v_2, \ldots, v_n$. Let $i = 1$ and delete as many edges between s and v_1 as possible so that (1) still holds. Then let $i = 2$ and do the same with edges between s and v_2. Continuing this way until we have processed all vertices in V we obtain a graph $G = (V+s, E' \cup E'')$ where (1) still holds and every edge incident to s enters a tight set, where we call a subset $X \in V$ **tight** if $d_G(X) = k$. This is true because otherwise we could have deleted the edge and still satisfied (1).

Claim If X and Y are tight sets, then

(I) If $X \cup Y \neq V$, then $X \cap Y$ and $X \cup Y$ are also tight

(II) $X - Y$ and $Y - X$ and both tight and $d(X \cap Y, V+s-(X \cup Y)) = 0$

It is easy to check that (I) follows from (i) and (1) and (II) follows from (ii) and (1).

We first prove that the number of edges between s and V (that is $|E''|$) is exactly $\alpha_{H,k}$.

Let $\mathcal{F}$ be a collection of tight sets that cover all edges incident to s and so that $|\mathcal{F}|$ is minimum over all such families and among families achieving this minimum let $\mathcal{F}$ be such that $\sum_{X\in\mathcal{F}}|X|$ is minimized. If $|\mathcal{F}|\geq 3$ or $\mathcal{F}=\{X,Y\}$ with $X\cup Y\neq V$, then it follows from the minimality of $\mathcal{F}$ and (I) that the sets in $\mathcal{F}$ are disjoint so $\mathcal{F}$ is a subpartition of V. If $\mathcal{F}=\{X,Y\}$ with $X\cup Y=V$, then it follows from (II) and the choice of $\mathcal{F}$ (minimizing the sum of the sizes of the sets) that $Y=V-X$ so $\mathcal{F}$ is again a subpartition. Now we have

$$\begin{aligned} d(s) &= \sum_{X\in\mathcal{F}} d(s,X) \\ &= \sum_{X\in\mathcal{F}} (k-d_H(X)) \\ &\leq \alpha_{H,k} \end{aligned}$$

On the other hand we also have $d(s)\geq\alpha_{H,k}$ since we must add at least $(k-d(X)$ edges from s to any set X belonging to the subpartition $\mathcal{F}'$ that achieves equality in (2). Thus we have $d(s)=\alpha_{H,k}$.

Now we are ready to find a set of $\lceil\frac{\alpha_{H,k}}{2}\rceil$ new edges whose addition to H makes the resulting graph k-edge-connected.

If $d(s)$ is odd in G, then we add an new edge from s to v_1 and call the resulting graph G. Hence we can assume below that $d(s)$ is even in $G=(V+s,E)$, where E is either $E'\cup E''$ or $E=E'\cup E''\cup\{sv_1\}$. Now we can apply Theorem 1 $\frac{d(s)}{2}$ times, each time choosing a neighbour u of s in the current graph and finding another neighbour v of s such that the splitting (su,sv) is feasible, that is, (1) still holds after deleting su,sv and adding the edge uv. The resulting graph $H'=(V,E'\cup F)$ is k-edge-connected and we added $\frac{d(s)}{2}=\lceil\frac{\alpha_{H,k}}{2}\rceil$ new edges. □

We now show that the proof above can be turned into an efficient algorithm for finding a minimum set of new edges whose addition to a given graph H which is not k-edge-connected makes the resulting graph k-edge-connected.

The two steps we need to show how to do efficiently are:

(A) Given an edge su incident with the special vertex s; find another edge sv such that the splitting (su,sv) is feasible.

(B) In the deletion process starting from the graph where s is joined to each vertex by k parallel edges, we need to be able to determine how many edges we can delete between the current vertex v_i and s in step i of the deletion process.

We first deal with (A): Let u be a fixed neighbour of s. As we have seen above, in order to check whether (su, sv) is a feasible splitting, we have to check whether $d(X) \geq k+2$ for every proper subset X of V which contains u, v. By Menger's theorem this is equivalent to checking whether $\lambda(z,t) \geq k+2$ for every $t \in V - \{u,v\}$ in the graph $G_{u,v}$ which we obtain by contracting the set $\{u,v\}$ to one vertex z. To calculate $\lambda(z,t)$ for a fixed $t \in V - \{u,v\}$ we make the flow network: $\mathcal{N}$ that we obtain from $G_{u,v}$ by replacing each edge pq by a two directed arcs $p \to q, q \to p$ each with capacity 1 and adding a new arc from s to t with capacity ∞. The last arc ensures that every finite (z,t)-cut $(S, \bar{S})$ in $\mathcal{N}$ will have $s \in \bar{S}$. By construction, the capacity of a finite (z,t)-cut $(S, \bar{S})$ is exactly $d(S)$ so the capacity of a minimum (z,t)-cut $(S, \bar{S})$ is exactly $\lambda(z,t)$. Hence we can check whether (su, sv) is a feasible splitting by performing at most $|V| - 2$ maximum flow calculations (one for each $t \in V - \{u,v\}$).

Now to (B): For each $i = 0, 1, \ldots, n-1$ we let G_i be the graph in the beginning of the i'th deletion step, that is, we are ready to find out how many edges of the kind sv_i we can delete without violating (1). So $G_0 = G$. After the deletion we must have $d(X) \geq k$ for every proper subset of V that contains v_i so we can delete p edges between v_i and s if and only if $d(X) \geq k+p$ for every proper subset of V that contains v_i. As above we can find the minimum degree of a proper subset X containing v_i by maxflow calculations, this time $|V| - 1$ of these, corresponding to the $|V| - 1$ choices of a vertex t to be outside X and fining a maximum (v_i, t)-flow. Let ρ be the minimum value of the $|V| - 1$ maxflows that we found. Then we can delete $\min\{k, \rho - k\}$ edges between s and v_i.

Corollary 1 *Let $k \geq 2$ be a fixed integer. There exists a polynomial algorithm that takes as input a graph H on n vertices and m edges which is not k-edge-connected and produces an optimal augmentation of H to a k-edge-connected graph H^* in time $O(nMF(n,m))$, where $MF(n,m)$ is the running time of the fastest algorithm for finding a maximum flow between two vertices p, q in a network.*

It is known that $MF(n,m)$ is $O(nm)$ so the complexity of the algorithm above is $O(n^2m)$.