Institut for Matematik og Datalogi
Syddansk Universitet

February 4, 2022
JBJ

DM867– Spring 2022 – Weekly Note 2

Bring material to class:

It is important that you bring the book and those sections of extra material that I have announced for the class with you. That way we can save time by referring to figures etc.

Stuff covered in week 5

I gave an overview of the topics that will be addressed in the course, including a quick recap of the complexity class NP as well as the notion of NPC-completeness. I also lectured on basic stuff on matroids, including the greedy algorithm. I showed how the greedy algorithm for minimum (maximum) spanning trees is just a special case of the greedy algorithm for matroids. The notes I used for the lecture on matroids are available on the homepage and and on itslearning.

It is important that you have some basic knowledge of flows since we will use flows several times in the course. If you feel you need some extra information about this topic, please read Sections 3.1-3.5 in BJG. Especially the flow decomposition property and section 3.4 are very useful. You may also read SCH sections 4.2-4.5. Besides this I have put videos of several lectures (from DM817) where I lecture on exactly Sections 3.1-3.5 in BJG. Please watch these when you have time (soon).

Material on the homepage :

Among (many other things you will find Chapter 12 from the book Approximation algorithms by V.V.Vazirani, Springer Verlag 2003. Here you can recapitulate basic things about LP-duality.

Lecture February 7, 2022

I will

- recall important parts of Sections 3.1-3.5 in BJG. I will not cover everything so you should watch the videos if this material is new to you!
- talk about matchings in bipartite graphs and show how to solve this problem using flows.
- I will also talk about connectivity of (di)graphs (Menger's theorem). Again flows are a useful tool here.
- I will give a very useful example of an NP-completeness proof (see notes below)

Material:

- BJG Section 3.1-3.5 and 3.11.1

- SCH sections 3.1-3.4
- SCH section 4.1
- BJG 7.3

Exercises for February 10, 2022:

- SCH Application 1.2 page 8 (read and understand so that you can explain it at the blackboard).
- SCH section 1.3 This describes the important extension of the shortest path problem where we can have negative weights on the arcs. (read and understand so that you can explain it at the blackboard).
- SCH application 1.3.
- Prove the following claims:
  - Let $D = (V, A)$ be a directed graph and let $\mathcal{F}_1$ denote those subsets A' of D with the property that no vertex v in D has more than one arc from A' which ends in v (an arc ends in v if it is of the form uv for some $u \in V$). Then $(A, \mathcal{F}_1)$ is a matroid on A.
  - Let $M = (S, \mathcal{F})$ be a matroid on S and let $X \subseteq S$. Define $\mathcal{F}_X$ as follows: $\mathcal{F}_X = \{Y \cap X | Y \in \mathcal{F}\}$. Then $(S, \mathcal{F}_X)$ is a matroid on S.
  - Let $M = (S, \mathcal{F})$ be a matroid on S and let $X \subseteq S$ with $X \in \mathcal{F}$ be given. Define $\mathcal{F}'_X$ as follows: $\mathcal{F}'_X = \{Y \subset S - X | X \cup Y \in \mathcal{F}\}$. Then $(S, \mathcal{F}'_X)$ is a matroid on S.
  - Let $M = (S, \mathcal{F})$ be a matroid on S and let $\mathcal{F}^* = \{X | X \cap B = \emptyset \text{ for some base } B \text{ of } M\}$. Then $(S, \mathcal{F}^*)$ is a matroid on S (called the **dual matroid** of M).
  - Given a connected undirected graph $G = (V, E)$, a non-negative real-valued weight function ω on the edges and a subset $E' \subset E$ which forms a forrest. Then, using the greedy algorithm, one can find a cheapest spanning tree T (with respect to ω) which contains all edges of E'.
- SCH application 1.4 Project scheduling
- SCH application 1.7.

Example of an NP-completeness proof

Recall that the 3-SAT problem is as follows: we are given a boolean formula $\mathcal{F} = C_1 \wedge C_2 \wedge \ldots \wedge C_m$ where each clause C_i uses exactly 3 literals over the boolean variables $x_1, x_2, \ldots, x_n$ (a literal is either a variable x_i or the negation $\bar{x}_i$ of a variable). For example we could have $C_i = (x_4 \vee \bar{x}_6 \vee x_9)$ and the question is whether there is an assignment $t : \{x_1, x_2, \ldots, x_n\} \rightarrow \{0,1\}^n$ so that each of the m clauses will evaluate to 1 (true). A clause is true under the assignment t if at least one of its literals evaluates to 1. Here $\bar{x}_i = 1 - x_i$ so if x_i is true under t ($t(x_i) = 1$) then $\bar{x}_i$ is false. The 3-SAT problem is one of the core NP-complete problems.

The SPECIAL (s,t)-PATH problem is as follows: Given a graph $G = (V, E)$, two distinct vertices $s, t \in V$ and subsets $W_1, W_2, \ldots, W_m$ of V, all of size 3 (they may share vertices); Question: is there an (s,t)-path in G which avoids at least one vertex from each W_j, $j = 1, 2, \ldots, m$?

We now describe a polynomial reduction from 3-SAT to SPECIAL (s,t)-PATH, which will show that SPECIAL (s,t)-PATH is also NP-complete. To prove this we have to describe a polynomial algorithm for transforming any instance $\mathcal{F}$ of 3-SAT into an instance $[G, s, t, \mathcal{W}]$ of SPECIAL (s,t)-PATH, where $\mathcal{W}$ is the collection of sets $W_1, \ldots, W_m$ for which we need to avoid at least one vertex in each with our (s,t)-path.

Let $\mathcal{F}$ be an instance of 3-SAT with m clauses $C_1, C_2, \ldots, C_m$ and n variables $x_1, x_2, \ldots, x_n$. Let $G = (V, E)$ have the following vertices:
$V = \{z_0, z_1, \ldots, z_n\} \cup \{v_1, v_2, \ldots, v_n\} \cup \{\bar{v}_1, \bar{v}_2, \ldots, \bar{v}_n\}$ and the following edges:
$E = \{z_i v_{i+1}, z_i \bar{v}_{i+1}, v_{i+1} z_{i+1}, \bar{v}_{i+1} z_{i+1} | i \in \{0, 1, \ldots, n-1\}\}$. For each clause C_i in $\mathcal{F}$ we define the set W_i as follows (illustrated by an example). If $C_i = (x_4 \vee \bar{x}_6 \vee x_9)$, then we let $W_i = \{v_4, \bar{v}_6, v_9\}$. Finally let $s = z_0$ and $t = z_n$

First observe that the reduction above (forming $G = G(\mathcal{F})$ from the instance $\mathcal{F}$ of 3-SAT) can be done in polynomial time in the size of $\mathcal{F}$ since G has $3n + 1$ vertices and $4n$ edges and there are m sets of size 3 to form so altogether we spend $O(n + m)$ time.

Suppose that G has an (s,t)-path P which avoids at least one vertex from each W_i, $i \in [n]$. We make a truth assigment t for $\mathcal{F}$ by letting $t(x_i) = 1$ precisely if v_i is not a vertex of P. This is a satisfying truth assigment, because, for each $j \in [m]$ the path P avoids at least one of the three vertices in the set W_j and the corresponding literal of C_j will be set to 1 in t.

Suppose conversely that there is a satisfying truth assigment t' for $\mathcal{F}$, that is, a $0,1$ assigment to each variable $x_1, \ldots, x_n$ so that at least one of the three literals in C_j is set to 1 for each $j \in [m]$. Then we let P be the (s,t)-path constructed as follows: Initially P is just the vertex s. Now perform n addition steps where we add 2 edges in the ith step as follows: If $t'(x_i) = 1$ we add the subpath $z_{i-1} \bar{v}_i z_i$ and otherwise we add the subpath

$z_{i-1}v_iz_i$. Since we constructed P so that it avoids the literal vertex v_i ($\bar{v}_i$) precisely when the corresponding literal x_i ($\bar{x}_i$) is 1 (true), it follows from the fact that t' is a satisfying truth assignment that P avoids at least one vertex in each W_j, $j \in [m]$.