Problem Set 4, Written Part

Network Flows and String Search

Due: 11:55pm Thursday, November 29, 2018

---

1 Tragic Comedy and Network Flows (8 marks)

A ship full of k comedians and $2k$ ordinary (non-comedian) people is capsizing. The hope is to evacuate all people onboard via rescue boats. Fortunately, each comedian has one rescue boat (in case they had to escape a tough crowd), and each boat can carry two additional passengers (for a total of one comedian and two non-comedians). Therefore, if the non-comedians could be arbitrarily paired up and each pair could go on one comedian's boat, everyone could be evacuated.

However, certain ordinary people so strongly dislike the jokes of certain comedians that they would rather swim than be passengers on those comedians' boats. More formally, for each ordinary person p_j (for $j \in \{1, 2, \ldots, 2k\}$), there is some subset S_j of the total set of comedians $\{c_1, \ldots, c_k\}$ with which they are willing to ride. Under the constraint that each of the $2k$ ordinary people needs to be assigned to a boat operated by a comedian whose jokes they will tolerate, and given that each boat can hold two ordinary people, the ship passengers wonder whether it is possible to evacuate everyone aboard.

Show how efficiently answering this question can be reduced to a Max-Flow problem.

2 Max-Flow with Node Capacities (8 marks)

In a standard Max-Flow problem, we assume that edges have capacities, and there is no explicit limit on how much flow is allowed to pass through a node. We now consider a variant of the standard problem where nodes, not edges, have capacities.

Let $G = (V, E)$ be a directed graph, with source $s \in V$, sink $t \in V$, and nonnegative node capacities c_v for each $v \in V$. Given a flow f in this graph, the flow through a node v is defined as $f^{\text{in}}(v)$. We call a flow feasible if it satisfies the usual flow-conservation constraints and the node-capacity constraints: $f^{\text{in}}(v) \leq c_v$ for all nodes.

Show how the Ford-Fulkerson algorithm can be used to find a maximum s-t flow in such a node-capacitated network.

3 Min-Cut and increasing all edge capacities by 1 (8 marks)

A friend makes the following claim: "Let G be an arbitrary flow network and let (A^*, B^*) be a minimum s-t cut. Now, let's increase the capacity of each edge by 1. Then (A^*, B^*) must still be minimum s-t cut."

Are they right? Explain why or why not.

4 Knuth-Morris-Pratt (8 marks)

Let the alphabet be {A, B, C}. Draw the DFA constructed by the Knuth-Morris-Pratt algorithm for the pattern CACABCA. In addition, show the corresponding transition function (by drawing a table like we saw in class).

5 Rabin-Karp and Wildcards (8 marks)

Consider the Rabin-Karp algorithm. A wildcard token is a token that matches any character. Suppose that the length-m pattern has a wildcard token at a given index $j \in \{0, 1, \ldots m-1\}$. Describe how to modify the algorithm to find a match for this pattern.