Single Source Multi-Commodity Network Flow

November 23, 2011

1 Motivation

Suppose you are attending a conference and have driven to the venue of this conference in your car. Suppose also, that all the other attendees have also driven to the conference venue in their cars (it might be worthwhile to note that this problem can be trivially extended to fit simple solutions such as carpools).

At the end of the conference, everybody decides to leave the venue to go home. They have all printed driving directions for the fastest route to get back home. Note that people living close to each other will share the majority of their routes (such as highways), with minor differences only towards the end.

There is however a problem there was a natural calamity during the conference, which caused the major roads to start sinking (note that this affects only the major roads). As a result, each road can only support a certain number of cars before it becomes unusable. Now, the new objective is to get everybody home, safely in minimal time.

2 Formalization

- Let G = (V,E) be a directed graph
- Let s $\in$ V be the source the conference venue
- Let $A = \bigcup_{0<i\leq k} a_i$ be the k attendees of the conference
- $\forall a_i \in A$, let $t_i \in V$ be the destination of a_i
- $\forall e \in E$, w(e) is the weight/cost of the edge. This models the time it takes to travel that edge (road)
- $\forall e \in E$, c(e) is the capacity of the edge
- $\forall e \in E$, f(e) is the total flow through that edge

  Let us now define $N = \{n_1, n_2, ..., n_j\}$, disjoint subsets of V, where each $n_i \in N$ is a neighborhood and each a_i lives in exactly one neighborhood (note that N is a partitioning of T = $t_1, ..., t_k$). Note that $\bigcup_i n_i$ does not have to be equal to V or $V \setminus \{s\}$. We can now partition A into disjoint subsets c_1, c_2, ..., c_j j commodities.

  Let d_i be the demand for c_i at it's terminal n_i. Note that $d_i = |\{c_i\}|$.

The objective is to find a network flow such that $\forall a \in A$ in each c_i is routed through the same path ending at n_i (an unsplittable flow [1]) and the total cost is minimized (note that the cost of an edge is dependent only on whether or not it is used, not on the amount of flow through it. This has the effect of modelling travel time by edge costs).

I propose to tackle an extension of a smaller problem: I will make the assumption that c(e) is at least $max_i(d_i) \forall e \in E$. I will then add a component to this reduced problem: Find the optimal partitioning N, of A such that the total cost of the network flow is minimized.

In order to do this, I will use the algorithm suggested in [1] to determine if an unsplittable flow for all c_i exists. I will then use a shortest path algorithm (say, Dijkstra's algorithm) to determine the actual flow graph. I will do this for multiple different partitions of $t_1, t_2, ..., t_k$ to find a minimal total cost routing. I propose to generate these partitions using a set partition algorithm from the text book. If there is time after I finish this, I will use multiple set partitioning algorithms from the text book (or elsewhere) and compare their runtimes.

3 References

[1] Dinitz, Yefim, Naveen Garg, and Michel Goemans. "On the Single-Source Unsplittable Flow Problem ." (1998): n. page. Web. 29 Oct. 2011. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.34.5780&rep=rep1&type=pdf>.