ADFOCS 2008 - Primal-Dual Algorithms for Online Optimization: Exercise Set 2

1. Given is an undirected graph $G = (V, E)$, a cost function $c : E \to \mathbb{R}^+$, and a requirement function f. The requirement function f is a set of demands of the form $D = (S, T)$, where S and T are subsets of vertices in the graph, such that $S \cap T = \emptyset$. A demand is satisfied by picking a path from a vertex in S to a vertex in T. Edges picked in the graph can be used to satisfy multiple demands. The goal is to find a minimum cost subgraph that satisfies the requirement function f.

   Our model is online; that is, the requirement function is not known in advance and it is given "demand by demand" to the algorithm, while the input graph is known in advance.

   (a) Show that online set cover is a special case of the above problem.

   (b) Formulate the above problem as a linear program and derive its dual.

   (c) Define an online primal-dual algorithm for the problem. What is the competitive factor obtained?

   (d) Suppose the input graph is a set of disjoint trees $T_1, \ldots, T_m$. The edges of the trees have non-negative costs. There is a set of *terminals* and each leaf in a tree is associated with a terminal. A terminal can be associated with several leaves, where each leaf belongs to a different tree. Each request is to a terminal t and a feasible solution is a path from one of the leaves associated with t to the root of the tree. The cost of the solution is the cost of the path picked.

      i. Convince yourself that this problem falls into the model defined in this question.

      ii. How would you apply online randomized rounding to a fractional solution generated by your primal-dual online algorithm for this problem?

2. Same setting as the previous one. Now, a feasible solution is a set of edges that separates for each demand $D = (S, T)$, any two vertices $s \in S$ and $t \in T$. The goal is to find a minimum cost subgraph that satisfies the requirement function f.

   (a) Formulate the above problem as a linear program and derive its dual.

   (b) Define an online primal-dual algorithm for the problem. What is the competitive factor obtained?

3. In the routing algorithms shown in class the requests are pairs (s, t), where for each request a path from s to t needs to be computed. Suppose now that each request is a set of vertices T, and a request is served by computing a Steiner tree on T. How would you adapt the routing algorithms shown in class for this version? Recall that the Steiner tree problem is NP-hard, yet 2-approximation algorithms (and even better) are known for it.

4. Show that there is an instance of the online fractional covering problem with n variables such that any online algorithm is $\Omega(\log n)$-competitive on this instance.

5. Consider the fractional algorithm shown in class for the weighted paging problem. How would you obtain an integral deterministic k-competitive algorithm for the problem along the same lines?