Problems on Network Flows Over Time

Martin Skutella

MDS Summer School on Discrete Optimization and Network Flows

June 24-26, 2011

Problem 1. A simple upper bound on the value of a maximum s-t-flow over time with time horizon T can be obtained by multiplying the minimum capacity of a (static) s-t-cut by T. Prove that this upper bound can diverge from the maximum value of a feasible s-t-flow over time by an arbitrarily large factor.

Problem 2. Consider the network G depicted in Figure 1.

(a) Use the Ford-Fulkerson Algorithm in order to determine a maximum s-t-flow over time with time horizon $T = 18$.

(b) Determine a minimum capacity s-t-cut over time with time horizon $T = 18$.

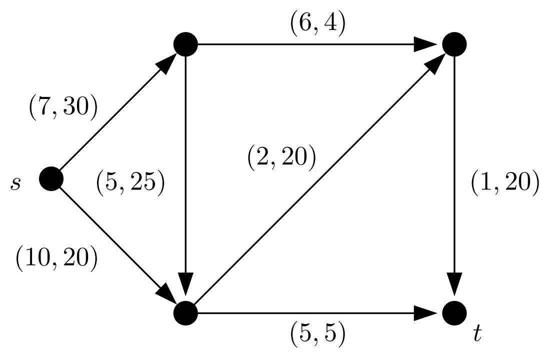

Figure 1: An instance of the Maximum Flow Over Time Problem. The arc labels indicate transit times and capacities, i.e., the label at arc e is (τ_e, u_e). The time horizon is $T = 18$.

Problem 3. Find a network such that no temporally repeated flow has the earliest arrival property.
Hint: Look for a network with unit capacities and a short path that blocks several longer paths.

Problem 4. The definition of an earliest arrival flow can be generalized straightforwardly to a setting with multiple sources, each with a given supply. It turns out that an earliest arrival flow still exists in this case. On the other hand, we run into problems when there are multiple sinks $t_1, \ldots, t_k$ with given demands $d_1, \ldots, d_k$ that bound the excess of the sinks. That is, $\mathrm{ex}_f(t_i, \theta) \leq d_i$ must hold for all i and θ. Construct an example with two sinks and one source for which no flow over time f maximizes $\mathrm{ex}_f(t_1, \theta) + \mathrm{ex}_f(t_2, \theta)$ for all θ simultaneously.

Problem 5. An s-t-flow over time f with time horizon T is a *latest departure flow*, if it maximizes the amount of flow leaving the source after time θ for all $\theta \in [0, T]$ (subject to the constraint that all flow must arrive at t before time T). Prove that the flow over time computed by the Earliest Arrival Algorithm is a latest departure flow.
Hint: Consider the network where the direction of every arc is reversed.

Problem 6. We consider temporally repeated flows in the context of min-cost flows over time.

(a) Construct a network with two temporally repeated flows f and f' corresponding to two different path decompositions of the same static s-t-flow x such that $c(f) \neq c(f')$.

(b) Construct an example in which the cost of any temporally repeated flow with time horizon T and value d is larger than the minimum cost of an s-t-flow over time with time horizon T and value d.

Problem 7. Show that an arbitrary s_0-$t_{(T-1)}$-cut with capacity $\delta < \infty$ in the time-expanded network G^T naturally induces an s-t-cut over time with time horizon T and capacity δ in G.

Problem 8. It follows from the work of Ford and Fulkerson that there always exists a maximum s-t-flow over time obeying the strict flow conservation constraints (i. e., no holdover at intermediate nodes). The purpose of this exercise is to show that this result does not hold in the more general setting with multiple commodities.

(a) Find an instance of the Multi-Commodity Flow Over Time Problem such that any multi-commodity flow over time with time horizon T satisfying all demands uses storage of flow at intermediate nodes for at least one commodity.
Hint: Consider the instance given in Figure 2.

(b) Try to come up with an instance of the Quickest Multi-Commodity Flow Problem such that the ratio of the minimal possible time horizon with and without storage of flow at intermediate nodes, respectively, is as large as possible.
Remark: This is an open research question. There is no instance known which achieves a larger ratio than the one depicted in Figure 2. On the positive side, it is known that the ratio is always upper bounded by 2.

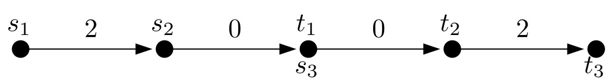

Figure 2: An example with $k = 3$ commodities. Commodities 1 and 3 each have demand 1, commodity 2 has demand 2. The numbers at the arcs indicate transit times; all arcs have unit capacity.