

(a) **Solution:** Code below.

This modification of Bellman Ford runs the relaxation algorithm k times. Each of these iterations computes the intermediate shortest paths up to length i on the i -th iteration, therefore relaxing k times computes the intermediate shortest paths up to length k on the k -th iteration. This algorithm runs in $O(kE)$ time with linear ($O(kV)$) space. ■

(b) **Solution:** The input for max $s - t$ k flow consists of the directed graph $G = (V, E)$, source and sink s, t respectively, capacity function $c(e)$ for each edge e , and a set of paths of length at most k denotes $P_{s,t}^{(k)}$. The following linear program computes the flow function f .

$$\begin{aligned} & \text{maximize} && \sum_{p \in P_{s,t}^{(k)}} f(p) \\ & \text{subject to} && \sum_{p \in P_{s,t}^{(k)}; e \in p} f(p) \leq c(e) \forall e \in E \\ & && f(p) \geq 0 \end{aligned}$$

The objective function is maximizing the amount of flow going through each one of the paths $p \in P_{s,t}^{(k)}$. The constraints make sure that these flow values through each path are at least 0 and also that they satisfy the capacity constraint, e.g. the amount of flow going through an edge corresponds to the path-wise flow of all paths that include that edge, and that value is bounded by capacity for that edge. It should be noted that the path based construction of flow implicitly guarantees the conservation constraint. The optimization of this linear program will produce precisely the max $s - t$ k flow, since this linear program captures path-based flow for only paths that are in $p \in P_{s,t}^{(k)}$. ■

(c) **Solution:**

$$\begin{aligned} & \text{minimize} && \sum_{(u \rightarrow v) \in E} c(u \rightarrow v) y_{u,v} \\ & \text{subject to} && \sum_{(u \rightarrow v) \in p} y_{u,v} \geq 1 && \forall p \in P_{s,t}^{(k)} \\ & && y_{u,v} \geq 0 && \forall (u \rightarrow v) \in E \end{aligned}$$

The objective function is minimizing a weighted value applied to each edge, subject to the primary constraint that for each path p there must be a collective value of at least one applied to all edges along that path. ■

referenced: <http://theory.stanford.edu/~trevisan/cs261/lecture15.pdf>