11.12.22

<u>Special case where $d_i = 1$ for all i and $c(e) = k$ for all edges e.</u>

Since, for every edge e, $c(e) = k \geq \sum_i d_i$, no congestion can occur. Hence we only have to worry about maintaining a valid family of k (s,t)-flows ~~pseudo-flows/s-t paths of demand d_i to for every flow pair.~~

We claim that the following algorithm solves the problem optimally:

1. Perform updates $U_{i,\, b:\, b \text{ is an } i\text{-block},\, v \in F_i^u \cap b - \gamma(b)}(v, i)$ (if any) in a single round.
2. Perform updates $U_{i,\, b:\, b \text{ is an } i\text{-block}}(\gamma(b), i)$ in a single round.
3. Perform updates $U_{i,\, b:\, b \text{ is an } i\text{-block},\, v \in b - F_i^u}(v, i)$ (if any) in a single round.

Proof.

~~Let $i \in [k]$ and b be an i-block.~~ We need to show that our algorithm outputs a feasible schedule. Optimality then follows from Lemma 4.1. Let U_1 be the set of updates performed in step 1 of our algorithm. Define U_2, U_3 analogously. Then we need to show (1) $U_1 \cup U_2 \cup U_3 = \bigcup_i V(F_i^o \cup F_i^u) \times \{i\}$ ~~$U_1 \cup U_2 \cup U_3 = V \times [k]$~~ and (2) ~~(i) U_1 obeys the consistency rule, (ii) $U_1 \cup U_2$ obeys the consistency rule, and (iii) for~~ U_1, U_2, and U_3 are pairwise disjoint, and (3) for every permutation of U_1, for every permutation of U_2, for every permutation of U_3, U_j ~~relation~~ obeys the consistency rule for every $j \in [V \times [k]]$.