Multicommodity Flow

- Given a directed network with edge capacities $u$ and possibly costs $c$.
- Give a set $K$ of $k$ commodities, where a commodity $i$ is defined by a triple $(s_i, t_i, d_i)$ – source, sink and demand.
- For each commodity, you want to find a feasible flow, subject to joint capacity constraints.
Formulation

- $f_i(v, w)$ is the flow of commodity $i$ on edge $(v, w)$.

\[
\sum_{w} f_i(v, w) - \sum_{w} f_i(w, v) = \begin{cases} 
0 & \text{if } v \neq s \text{ and } v \neq t \\
d_i & \text{if } v = s \\
-d_i & \text{if } v = t
\end{cases} \quad \forall v \in V, i \in K
\]

- $\sum_{i \in K} f_i(v, w) \leq u(v, w) \quad \forall (v, w) \in E$

- $f_i(v, w) \geq 0 \quad \forall (v, w) \in E$

- Single commodity flow: $m$ variables, $m + n$ constraints

- Multicommodity flow: $km$ variables, $kn + m$ constraints, $km$ non-negativity constraints

**Size of A matrix:** $km(kn + m) = k^2nm + km^2$

A computationally challenging problem
Facts About Multicommodity Flow

- LP is big
- A matrix is not Totally Unimodular.
- Optimal solution to a multicommodity flow LP might be fractional.
- All feasible solutions might be fractional.

Optimization Variants

- Given costs on edges, \( c(v, w) \), find a feasible flow minimizing \( \sum_i \sum_{vw} c(v, w)f_i(v, w) \)
- No given demands, maximize total flow
- No given demands maximize total flow cost
- Send at least \( z \) percent of each demand, maximize \( z \). (concurrent flow)
- Send demands, find minimum \( \alpha \) such that the flow is still feasible with capacities \( \alpha c(v, w) \). (equivalent to previous problem)
Solutions

- Optimal fractional solution is solvable by LP in polynomial time.
- Polynomial is large (degree 6 or so).
- No known polynomial time algorithms for multicommodity flow that do not use LP ("easiest" such problem without a combinatorial algorithm).
- There are combinatorial algorithms that find a \((1 + \epsilon)\)-optimal solution to concurrent flow in polynomial time (many algorithms e.g. \(O(\epsilon^{-2} knm)\) time).
- Finding a feasible integer solution is NP-complete. Even the disjoint paths version is NP-complete.
Approximation for Concurrent Flow

- Consider integer problem, $u = 1, d = 1$.
- Objective is to maximize fraction of demand sent.
- Equivalent problem: Send one unit of each demand, allow capacity constraints to be violated, but minimize $\lambda = \max_{(v, w)} \sum_i f_i(v, w)$
- Assume wlog, that a fractional flow exists

Algorithm
- Find the optimal fractional flow, via LP
- “Round” the fractions (carefully...)

How to Round
- Decompose the flow for commodity $i$ into a set of $\beta_i$ $s_i - t_i$ paths, $P_1^i, \ldots, P_{\beta_i}^i$ with values $f_1^i, \ldots, f_{\beta_i}^i$.
- Interpret the flow values as probabilities and choose a path for commodity $i$ according to the probability distribution defined by the flows.
Analysis

Use a Chernoff Bound.

Let $x_i$ be $0-1$ random variables, where $x_i = 1$ with probability $p_i$. Let $M = E(\sum x_i) = \sum p_i$. Then, for $0 < \beta < 1$, we have

$$\Pr(\sum x_i > (1 + \beta)M) \leq e^{-\beta^2 M/2}$$