Unsplittable Flow Problem on Paths and Trees

Closing the LP Relaxation Integrality Gap

Adam Jozefiak, Yuchong Pan

November 28, 2019

The Problem

- Given an undirected graph (a tree or path) $G = (V, E)$ with edge capacities $c_e \in \mathbb{R}_+, \forall e \in E$. Where $|V| = n, |E| = m$.
- Given a set of k requests $\{R_1, \dots, R_k\}$ (denoted by $\mathcal{R} = \{1, \dots, k\}$). Where each request R_i is characterized by $((s_i, t_i), d_i, w_i)$.
  - $s_i, t_i \in V$ are the source and destination vertices of request R_i. Let P_i be the unique s_i-t_i path in G
  - $d_i \in \mathbb{R}_+$ is the demand of request R_i.
  - $w_i \in \mathbb{R}_+$ is the weight of request R_i.
- $S \subseteq \mathcal{R}$ is routable if $\forall e \in E, \sum_{i \in S} d_i \leq c_e$
- Goal is to find a routable subset S of requests that maximizes total weight: $\sum_{i \in S} w_i$
- Refer to an instance on a tree as UFP-Tree and an instance on a path as UFP-Path
- Aside: The term unsplittable comes from the general case where requests on a general graph must be routed along a single path.

Integer Program Formulation

We can formulate the UFP-Tree and UFP-Path problems as an integer program where $x_i \in \{0, 1\}$ corresponds to choosing to route request R_i. We will call this integer program UFP-IP:

$$\begin{array}{rll} \text{maximize} & \sum_{i=1}^{k} w_i x_i & \\ \text{s.t.} & \sum_{i:e \in P_i} d_i x_i \leq c_e & \forall e \in E \\ & x_i \in \{0, 1\} & \forall i \in \mathcal{R} \end{array}$$

LP Relaxation

UFP-IP leads to a natural LP relaxation which we will call UFP-LP:

$$\begin{aligned} \text{maximize} \quad & \textstyle\sum_{i=1}^{k} w_i x_i & \\ \text{s.t.} \quad & \textstyle\sum_{i:e\in P_i} d_i x_i \leq c_e & \forall e \in E \\ & x_i \in [0,1] & \forall i \in \mathcal{R} \end{aligned}$$

The papers being surveyed present LP relaxations for UFP-Path and UFP-Tree (which can be solved in polynomial time) with the goal of minimizing the integrality gap between the LP relaxation and UFP-IP.

No Bottleneck Assumption and Natural LP Relaxation

- A UFP instance satisfies the *no bottleneck assumption* (NBA) if:

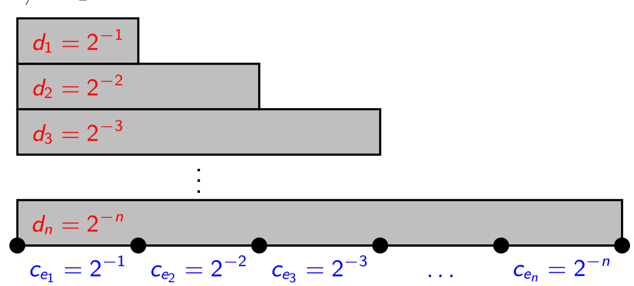
$$\max_{i \in \mathcal{R}} d_i \leq \min_{e \in E} c_e$$

- Chekuri, Mydlarz, and Shepherd 2003 [1] proved two key results on the integrality gap of the natural LP relaxation UFP-LP on UFP-Tree instances with NBA:

  1. For an instance of UFP-Tree with unit demands (i.e $d_i = 1 \forall i \in \mathcal{R}$) an integrality gap of at most 4 is attained.
  2. For a general demand instance of UFP-Tree an integrality gap of at most 48 is attained.

Natural LP Relaxation Without No Bottleneck Assumption

- Chakrabarti, Chekuri, Gupta, and Kumar 2007 [2] give an example on a path with an integrality gap of $\frac{n}{2}$, with $w_i = 1, \forall i \in \mathcal{R}$.



- Any feasible solution can only route at most one request. Hence the maximum weight over all routable sets is 1.
- $x_i = \frac{1}{2}, \forall i \in \mathcal{R}$ is a feasible solution to the natural LP relaxation UFP-LP which attains a total weight of $\frac{n}{2}$.
- Hence the integrality gap of this instance is $\Omega(n)$.

Adding Rank Constraints to UFP-Path without NBA LP Relaxation

- "Strengthening LP relaxations by adding valid inequalities is a standard methodology in mathematical programming." – Chekuri, Ene, and Korula (2009) [3].
- Accordingly, [3] deals with UFP-Path without NBA by adding new rank constraints to the natural LP relaxation in order to derive two new LP relaxations that attain a $O(\log(n))$ integrality gap.
- A rank constraint is as follows:
  - Let $S \subseteq \mathcal{R}$
  - Let $rank(S) =$ maximum number of requests in S that can be routed simultaneously.
  - Then the constraint is: $\sum_{i \in S} x_i \leq rank(S)$
- In particular [3] utilizes rank constraints for "big" requests (requests whose demands are at least $\frac{3}{4}$ of their bottleneck edge's capacity). Big requests are what make the lack of NBA difficult.

Generalizing Rank Constraints to UFP-Tree without NBA LP Relaxation

- Friggstad and Gao 2015 [4] generzlize [3]'s blocking rank constraints to UFP-Tree without NBA and attain a $O(\log(n) \cdot \min\{\log(n), \log(k)\})$ integrality gap for their two LP relaxations.
- It is interesting to note that this integrality gap result matches [3]'s $O(\log(n) \cdot \min\{\log(n), \log(k)\})$-approximation algorithm. Part of the design of the LP relaxations with rank constraints in [3] was motivated by [3]'s approximation algorithm on UFP-Trees without NBA.
- Additionally, [4] demonstrates that even with all of the rank constraints there is a $\Omega(\sqrt{\log(n)})$ integrality gap for UFP-Tree without NBA through an explicit UFP-Tree instance, similar in spirit to the example in [2].

Generalizing Rank Constraints to UFP-Tree without NBA LP Relaxation

- $\forall i \in \mathcal{R}, \forall v$ in the span of $P_i, \forall a \in \{s_i, t_i\}$, define a *blocking set* $C(i, v, a)$ that includes i and all other $j \in \mathcal{R}$ s.t.
  - v is in the span of P_j,
  - $d_j \geq d_i$,
  - $d_i + d_j > c_e$ for some $e \in P(a, v) \cap P_j$.
- $C(i, v, a)$ generalizes $\text{LeftBlock}(i, e)$ and $\text{RightBlock}(i, e)$ from [3].
- [4] shows that $\text{rank}(C(i, v, a)) = 1$ for all blocking sets.

$$
\begin{array}{llcll}
\text{maximize} & \displaystyle\sum_{i=1}^{k} w_i x_i & & & \\
\text{s.t.} & \displaystyle\sum_{i:e\in P_i} d_i x_i & \leq & c_e & \forall e \in E \\
& \displaystyle\sum_{i\in C(i,v,a)} x_i & \leq & 1 & \forall \text{ blocking sets } C(i, v, a) \\
& x_i & \in & [0, 1] & \forall i \in \mathcal{R}
\end{array}
$$

Constant Integrality Gap for UFP-Path without NBA

- Anagnostopoulos, Grandoni, Leonardi, and Wiese 2013 [5] formulate an LP relaxation for UFP-Path without NBA that has a constant factor integrality gap.
- This result improves the then tightest LP relaxation integrality gap of $O(\log(n))$ by [3] (for UFP-Path without NBA).
- The authors of [5] are able to attain this result by using dynamic programming embeddings into linear programs.

References

1. C. Chekuri, M. Mydlarz, F. B. Shepherd. Multicommodity Demand Flow in a Tree and Packing Integer Programs. *ACM Trans. on Algorithms*, 3(3), 2007. Preliminary version in *Proc. of ICALP*, 410-425, 2003.
2. A. Chakrabarti, C. Chekuri, A. Gupta, A. Kumar. Approximation Algorithms for the Unsplittable Flow Problem. *Algorithmica*, 47(1):53-78, 2007.
3. C. Chekuri, A. Ene, and N. Korula. Unsplittable flow on paths, trees, and column-restricted packing integer programs. In *proceedings of APPROX*, 2009.
4. Z. Friggstad, Z. Gao. On linear programming relaxations for unsplittable flow in trees. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2015)*.
5. A. Anagnostopoulos, F. Grandoni, S. Leonardi, A. Wiese. Constant integrality gap LP formulations of unsplittable flow on a path. In *International Conference on Integer Programming and Combinatorial Optimization* (pp. 25-36), 2013.