CMPUT 675: Approximation Algorithms **Winter 2005**

Lecture 17: Mar. 16

Lecturer: Mohammad R. Salavatipour *Scribe: Eddie Rafols*

17.1 Uncapacitated Facility Location (UFL)

Input:

- F, a set of n facilities;
- C, a set of cities/clients/users, $(|C| = m)$;
- f_i, the cost of opening facility i;
- For all $1 \leq i \leq n, 1 \leq j \leq m$: C_{ij} is the cost of connecting city j to facility i

Goal:

- Open a subset of the facilities and connect each city to an open facility such that the total cost of opening facilities and connecting cities to open facilities is minimized

We assume that $F \cap C = \emptyset$. However this is not always the case, in the *complete* case, the intersection of F and C may be non-empty. It turns out that any approximation ratio in the bipartite case carries over to the complete case. Therefore, we only focus on the bipartite case.

Metric UFL: In the metric UFL problem, the connection costs satisfy the triangle inequality, i.e. for every $1 \leq i \leq n$ and $1 \leq j \leq m$: $C_{ij} \leq C_{i'j} + C_{i'j'} + C_{ij'}$.

Uncapacitated Facility Location can be viewed as a set cover problem. In fact, the first approximation algorithm for UFL was based on the greedy algorithm for set cover and has a factor $O(\log n)$-approximation [Hochbaum82]. This is the best factor for non-metric UFL.

Several approximation algorithms since then have appeared since then. Some of which are the following:

- The first constant factor approximation was developed by Shmoys/Trdös/Agrarawal in 1997 using LP-rounding
- A 3−approximation using the Primal-dual technique was developed in 1999 by Jain and Vazirani (this algorithm is presented in the text book).
- We see a greedy algorithm due to Jain/Mahdian/Markakis/Saberi/Vazirani (J. ACM 2003) which has a factor 1.861−approximation. With some careful tuning, the factor can be reduced to 1.61
- The best known approximation factor is 1.52 by Mahdian/Ye/Zhang. (The lower bound for UFL is 1.462.)

17.2 UFL as a Set Cover

In order to frame UFL as a Set Cover instance, we define the notion of stars (corresponding to sets) and cost-effectiveness.

Star: A star is one facility and the cities that are connected to it. The cost of a star is the equal to the cost of opening the facility and the sum of the connection costs. Formally, for a star (i, C') which has facility i and cities $C' \subseteq C$, the cost is $f_i + \sum_{j \in C'} C_{ij}$. The cost-effectiveness of (i, C') is:

$$\frac{f_i + \sum_{j \in C'} C_{ij}}{|C'|}$$

Although the number of stars is exponentially large, we can find the most cost-effective star in polynomial time. For each facility i, sort the cities in a non-decreasing order based on the connection cost to i. The most cost-effective star containing facility i will consist of the first k cities for some value of k. As k is increases, the cost-effectiveness improves until we reach an optimal cost-effectiveness, at which point cost-effectiveness will begin to get worse. We can find this optimal cost-effectiveness for each facility i, then pick the best.

Here is the algorithm for general (non-metric) UFL:

Algorithm 0

1. Consider the stars as sets and the cities as elements
2. Apply the greedy set cover algorithm

Algorithm 0 is an $O(\log n)$-approximation, but we can do better. Once a facility has been opened, the facility can be used later for only the cost of connection (We only pay to open a facility once). This intuition can be used to improve the algorithm. To take advantage of this, we redefine cost-effectiveness.

New definiton of cost-effectiveness: Given a star (i, C'), assume that the r uncovered cities of C' have costs $C_{ij_1} \leq C_{ij_2} \leq \ldots \leq C_{ij_r}$. The cost-effectiveness of star $(i, C')$is now:

$$\frac{g_i + \sum_{s=1}^{r} C_{ij_s}}{r}$$

where

$$g_i = \begin{cases} f_i & \text{if facility } i \text{ is open;} \\ 0 & \text{otherwise} \end{cases}$$

Algorithm 1

U is the set of uncovered cities
initially $U \leftarrow C$ and all facilities are unopened
while $U \neq \emptyset$ do:
    Find the most cost-effective star (i, C')
    Open i if it is not already open, connect all of C' to i
    $U \leftarrow U - C'$
    $f_i = 0$

We are going to analyse this algorithm using dual fitting. For this purpose, we formulate the UFL problem as an IP/LP. We will only use this LP formulation for the purpose of the analysis of the algorithm. Let x_s be an indicator variable for a star s. Let $\mathcal{S}$ be the set of all stars and c_s be the cost of star s.

Primal LP		**Dual LP**	
minimize	$\sum c_s x_s$	maximize	$\sum y_j$
such that $\forall j \in C$:	$\sum_{s:j\in s} x_s \geq 1$	such that $\forall s \in \mathcal{S}$:	$\sum_{j\in s} y_j \leq c_s$

For the dual, we can think of the y_j's as the contribution of city j toward the total expense. We have to maximize the total prices (contributions) such that the amount for every star is not more than its cost. The constraint for the dual is $\sum_{j\in s} \max(0, y_j - C_{ij}) \leq f_i$. This equation is derived by rewriting the constraint on the dual using the formula for the cost of a star.

If we start raising the dual variables y_j for uncovered cities (simultaneously), the most cost-effective star will be the first star for which $\sum_{j\in s} \max(0, y_j - C_{ij}) \leq f_i$ becomes tight. This suggests that we can rewrite Algorithm 1 in the following way. We have a notion of time (which starts at zero) and will increase.

Restatement of Algorithm 1
U is the set of uncovered cities
initially time is zero, $U \leftarrow C$, and all facilities are unopened
while $U \neq \emptyset$ do:
Increase time and simlutaneously increase y_j for every city $j \in U$ until either:
For some unopened facility, $\sum_{j\in s}\max(0, y_j - C_{ij}) = f_i$
In this case, open facility i and for every unconnected city j with $y_j \geq C_{ij}$, connect j to facility i and remove j from U
For some uncovered city j and opened facility i, $y_j = C_{ij}$
In this case, connect city j to facility i and remove j from U.

We know that Algorithm 1 returns a feasible primal from the condition on the while loop. The cost of the solution is $\sum y_j$ because the total amount each city pays goes toward opening a facility and for connection costs or is just exactly the connection cost. This means that in this case, the cost of the primal solution is the same as the cost of the dual solution. This is only possible when the primal and dual are both optimal, or one of primal or dual is unfeasible.

In this case, the dual is not necessarily feasible because at each iteration we exclude some cities from U, so at the end, for some facility we may have $\sum_{j\in s} \max(0, y_j - C_{ij}) > f_i$.

However, it can be shown that there is a dual feasible solution that differs from y by only a small factor. More specifically it can be shown that:

Lemma 17.1 (JMMSV03) *For some $\gamma \leq 1.861$, y_j/γ is a feasible dual solution*

This implies that $\frac{1}{\gamma}\sum_j y_j$ is a lower bound for the cost of optimal primal solution. Since our primal solution has cost $\sum_j y_j$ we have:

Theorem 17.2 *Algorithm 1 is a 1.861−approximation algorithm for uncapacitated facility location.*

References

JMMSV03 K. Jain; M. Mahdian; E. Markakis; A. Saberi; V. Vazirani, Greedy Facility Location Algorithms Analyzed Using Dual Fitting with Factor-Revealing LP, *J. ACM*, 2003.