The Capacitated Matroid Median Problem

by

Sanchit Kalhan

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Combinatorics and Optimization

Waterloo, Ontario, Canada, 2018

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

In this thesis, we study the capacitated generalization of the Matroid Median Problem which is a generalization of the classical clustering problem called the k-Median problem. In the capacitated matroid median problem, we are given a set $\mathcal{F}$ of facilities, a set $\mathcal{D}$ of clients and a common metric defined on $\mathcal{F} \cup \mathcal{D}$, where the cost of connecting client j to facility i is denoted as c_{ij}. Each client $j \in \mathcal{D}$ has a demand of d_j, and each facility $i \in \mathcal{F}$ has an opening cost of f_i and a capacity u_i which limits the amount of demand that can be assigned to facility i. Moreover, there is a matroid $\mathcal{M} = (\mathcal{F}, \mathcal{I})$ defined on the set of facilities. A solution to the capacitated matroid median involves opening a set of facilities $F \subseteq \mathcal{F}$ such that $F \in \mathcal{I}$, and figuring out an assignment $i(j) \in F$ for every $j \in \mathcal{D}$ such that each facility $i \in F$ is assigned at most u_i demand. The cost associated with such a solution is : $\sum_{i \in F} f_i + \sum_{j \in \mathcal{D}} d_j c_{i(j)j}$. Our goal is to find a solution of minimum cost.

As the Matroid Median Problem generalizes the classical NP-Hard problem called k-median, it also is NP-Hard. We provide a bi-criteria approximation algorithm for the capacitated Matroid Median Problem with uniform capacities based on rounding the natural LP for the problem. Our algorithm achieves an approximation guarantee of 76 and violates the capacities by a factor of at most 6. We complement this result by providing two integrality gap results for the natural LP for capacitated matroid median.

Acknowledgements

I would like to thank my supervisor, Professor Chaitanya Swamy. His insight and support during our weekly meetings have made this thesis possible and his detailed feedback during the writing process has taught me a great deal about how to communicate theoretical concepts and the intuition behind them. I have been extremely lucky to have had Professor Swamy as my advisor and thank him for his guidance and patience as they have helped mature me as a researcher.

I would like to thank my office mates Matthew, Maxwell and Ramin for creating a fun and productive environment in the office.

I want to thank Sharat for helping me with grading deadlines, to Abhinav for driving me around for random errands (albeit reluctantly), to Jimit, who was always up for a trip to Kismet and to Cedric for agreeing to drive me to Toronto at the last minute without any hesitation. My graduate experience at Waterloo would have been lacking without all of you and your friendship.

I would like to thank my family for their support and encouragement.

Table of Contents

List of Tables

List of Figures

List of Algorithms

Chapter 1

Introduction

Consider the following scenario, the multinational furniture company UKEA has decided to open stores in the Greater Toronto Area (GTA) and is interested in finding the best locations to open their stores. They have done their research on the ground and know the demands of their products in all the GTA neighborhoods. They are interested in minimizing the total distance people have to travel to come to their stores while being constrained by having a budget to only open k stores.

This problem, and its variants involving a variety of different objective functions and constraints are clustering problems which have been intensively studied in the past couple of decades. The problem described above is an example of the k-median problem, wherein we have a set of candidate centers (facilities) and a set of clients with demands, both located at points in a metric space. In the simplest version of the problem, called the *uncapacitated k-median problem (UKM)*, the goal is to open at most k-centers and assign every client to a center so as to minimize the sum of the distances between clients and their assigned facilities. A more general version of the problem is the *capacitated k-median problem (CKM)*, wherein each facility has a capacity (possibly different for different facilities) that limits the total demand that may be assigned to the facility. A related problem that is also widely popular in literature is called the *uncapacitated facility location problem (UFL)*, wherein each center has an opening cost but there is no constraint on the total number of facilities that can be opened. Here the goal is to open a subset of facilities and assign the clients to open facilities so as to minimize the sum of the facility-opening costs and the sum of the distances between clients and their assigned centers. Similar to k-median, one can consider a capacitated generalization of the problem call *capacitated facility location (CFL)*.

Even the simplest versions of these problems (UKM and UFL) turn out to be computationally intractable and much of the literature in theoretical computer science has focused on the design of approximation algorithms for these problems [Shm00].

Consider the following problem that is similar to the k-median problem. Suppose that the set of facilities is partitioned into red facilities and blue facilities; and the solution is supposed to assign clients to open facilities while being constrained to open at most k_r red facilities and at most k_b blue facilities. This problem is called the *red-blue median problem* [HKK10] and can be viewed of as a generalization of the k-median problem in which the cardinality constraint on the set of facilities is replaced by a partition matroid constraint. More generally, one can think of having an arbitrary matroid on the set of facilities and constraining the solution so that the set of open facilities forms an independent set of this matroid. This problem is called the *matroid median problem* [KKN+11, Swa16] and it turns out to be a powerful generalization. The matroid median problem with facility opening costs captures and unifies various facility location type problems that had previously been investigated separately (eg. the data placement problem [BRS08], mobile facility location [FS11a, AFS13], etc). Capacities arise naturally in these applications and incorporating them leads to models that better capture the underlying application.

In this thesis, we investigate the matroid median problem with capacities. We define the problem precisely in Section 1.1 and state the results in section 1.2. We conclude this chapter with an overview of related work in Section 1.3.

1.1 Problem definition and special cases

In the *capacitated matroid median (CMM) problem*, we are given a set of facilities $\mathcal{F}$ and a set of clients $\mathcal{D}$ and a common metric defined on $\mathcal{F} \cup \mathcal{D}$. Each facility $i \in \mathcal{F}$ is associated with an opening cost f_i and a capacity U_i, and each client $j \in \mathcal{D}$ is associated with a demand d_j. Moreover, there is a matroid $\mathcal{M} = (\mathcal{F}, \mathcal{I})$ defined on the set of facilities. The goal is to open a set of facilities $F \subseteq \mathcal{F}$ such that $F \in \mathcal{I}$ and figure out an assignment $i(j) \in F$ for each client j such that for each $i \in F$, the total demand assigned to i is at most U_i. The assignment cost incurred by a client j under such an assignment is given by $d_j c_{i(j)j}$. The goal is to minimize the total cost - sum of the facility opening costs and assignment costs which is given by: $\sum_{i \in F} f_i + \sum_{j \in \mathcal{D}} d_j c_{j(i)j}$.

There are two variants of CMM that can be considered : the case when the capacities are uniform ($U_i = U$ for each $i \in \mathcal{F}$) which we denote by CMM-UC, and the case with non-uniform capacities (in which the facilities are allowed to have different capacities) which we denote by CMM-NC.

The CMM problem is the capacitated generalization of the matroid median problem (UMM), that is in UMM we have $U_i = \infty$ for each $i \in \mathcal{F}$. It is easy to see how CMM captures CKM and CFL. In CKM, there are no facility opening costs and the matroid on the set of facilities simply captures the cardinality constraint, i.e., every set of size at most k is independent. In CFL, there are facility opening costs and there is no constraint on the set of open facilities (so the matroid is the trivial one where every set is independent).

In the next section, we describe our results for CMM. To better understand the nature of our results, we briefly give an overview of the nature of results known for CKM and UMM. A more detailed discussion of these results can be found in Section 1.3.

Prior work on CKM has led to bi-criteria approximation algorithms relative to the natural LP, where algorithms that achieve a constant factor approximation guarantee with respect to the natural LP are known to violate either, the cardinality constraint on the total number of facilities that can be opened (the bound k) or the capacity constraints on the facilities, by an $O(1)$ factor [AvdBGL15, CGTS99, BFRS15]. For UMM, again constant factor approximation algorithms are known relative to the natural LP [KKN+11, Swa16]. In this work, we investigate whether we can leverage the techniques utilized in these strands of work to obtain similar guarantees for capacitated matroid median. More specifically, we investigate if the natural LP leads to bi-criteria constant-factor approximation guarantees.

1.2   Our results

We essentially settle the approximability of capacitated matroid median relative to the natural LP (modulo $O(1)$-factors). Our main result is a constant factor approximation algorithm for the capacitated matroid median problem with uniform capacities (CMM-UC) which violates capacities by a factor of at most 6. More formally,

Theorem 1.1. *There is an approximation algorithm for CMM-UC that computes a solution of cost at most* 76OPT *and violates the capacities by a factor of at most 6, where* OPT *denotes the cost of the optimal solution to the natural LP relaxation.*

We give the algorithm and its analysis which proves Theorem 1.1 in Chapter 2.

We complement our main theorem above with a couple of integrality gap examples for the natural LP. Our first integrality gap example shows that for the capacitated matroid median problem with non-uniform capacities (CMM-NC), the natural LP has unbounded integrality gap, even if we are allowed to violate the capacities by any constant factor.

Theorem 1.2. *The natural LP for CMM-NC has unbounded integrality gap even when a solution is allowed to violate the capacity constraint by a factor of β, for any $\beta \geq 0$.*

Our second integrality gap result considers a result complementary in nature to the previous two results. As we discussed in the previous section, for capacitated k-median, two types of bicriteria O(1)-approximation results are known relative to the natural LP, one where capacities are violated, and one where the cardinality bound is violated. Analogous to the case of violating the cardinality constraint, we may consider guarantees relative to the natural LP that violate the matroid constraint on the set of facilities. Here, an α violation of the matroid constraint by a vector $y \in \mathbb{R}^{\mathcal{F}}$ means that for every $S \subseteq \mathcal{F}, \sum_{i \in S} y_i \leq \alpha \cdot \mathrm{r}(S)$, where r(.) is the rank function of the matroid. We were able to show the following result for results of this nature:

Theorem 1.3. *The natural LP for CMM-UC has unbounded integrality gap even when a solution is allowed to violate the matroid constraint by a factor of α, for any $\alpha \geq 0$.*

We can summarize the results of Theorems 1.1, 1.2 and 1.3 using the following table, which summarizes the integrality gap of the natural LP for capacitated matroid median with uniform/non-uniform capacities for the two types of results studied in literature (capacity/cardinality violation). Each cell shows the integrality gap along with maximum violation factor of the integral solution.

Capacities	Cardinality violation	Capacity violation
Uniform	Unbounded, for any constant violation	≤ 76, with violation ≤ 6
Non-uniform	Unbounded, for any constant violation	Unbounded, for any constant violation

Table 1.1: Integrality gap of natural LP for CMM - Contribution of this thesis.

1.3 Related work

There has been a great deal of work on facility location and k-median problems and these problems have been thoroughly examined through various algorithmic techniques.

The uncapacitated facility location problem especially has been attacked using a variety of algorithmic frameworks. The first constant-factor approximation guarantee for UFL was obtained by Shmoys et al [STA97], using LP-rounding. Since then, many LP techniques like filtering [LV92], primal-dual [JV01] and dual-fitting [JMM$^+$03] have been used to

achieve a better understanding of the problem. The current best approximation ratio for UFL is 1.488 by Li [Li11]. On the other hand, the capacitated facility location has not seen the same amount of success as its uncapacitated counter part. Until recently, the only known approximation guarantees for capacitated facility location were achieved through local search techniques. Bansal et al. gave a 5-approximation algorithm for CFL with non-uniform capacities [BGG12] and Aggarwat et al. gave a 3-approximation for CFL with uniform capacities [ALB$^+$13]. It was a big open problem to get an LP-based approximation algorithm for CFL; this was resolved positively by An et al in which they gave a constant factor LP-based approximation algorithm for CFL [ASS14].

On the other hand, for CKM, all known algorithms achieve bi-criteria approximation guarantees. The natural LP for CKM is known to have unbounded integrality gap even if the capacity constraints or the cardinality constraint is allowed to be violated to an extent of $(2 - \epsilon)$. There are two variants of CKM considered in literature - soft CKM and hard CKM. In soft CKM, we're allowed to open multiple facilities at the same location (but these still count towards the cardinality constraint). For soft CKM with non-uniform capacities, Chuzhoy and Rabani gave primal-dual based 40-approximation algorithm that violates capacities by a factor of 50 [CR05]. Byrka et al. gave an $O(\frac{1}{\epsilon^2})$ approximation algorithm for hard CKM with uniform capacities that violates capacities by a factor of at most $2 + \epsilon$ [BFRS15]. Gijswit and Li also gave a $(7 + \epsilon)$-approximation algorithm for hard CKM with non-uniform capacities which opens at most $2k$ facilities [AvdBGL15].

More recently, stronger LP's have been introduced for CKM with uniform capacities and CKM with non-uniform capacities. The Rectangle LP was used by Li to give a $O(\frac{1}{\epsilon^2})$-approximation algorithm for CKM with uniform capacities that opens at most $(1 + \epsilon)k$ facilities [Li17]. Byrka et al. then used the configuration LP to achieve a $O(\frac{1}{\epsilon^2})$-approximation algorithm for CKM with uniform capacities that violates capacities by a factor of $(1 + \epsilon)$ [BRU16]. Finally, Demirci and Li gave a $O(\frac{1}{\epsilon^5})$-approximation algorithm for CKM with non-uniform capacities which violates the capacity constraint by a factor of $(1 + \epsilon)$ [DL16]. Getting a true approximation algorithm for CKM still remains a big open problem.

For the uncapacitated matroid median problem (UMM), there are true constant factor approximation algorithms known based on rounding the natural LP. The paper of Krishnaswamy et al [KKN$^+$11] gave a 16-approximation algorithm for UMM without facility opening costs. Swamy then gave an 8-approximation algorithm was given for the problem with facility opening costs [Swa16]. Recently, Krishnaswamy et al [KLS17] improved upon this ratio by providing an approximation algorithm that achieves a guarantee of 7.081, although this does not quite prove an improved integrality gap for the natural LP of matorid median.

Chapter 2

Our Algorithm

In this chapter, we present our algorithm for CMM-UC and its analysis. This is the first approximation guarantee for the capacitated generalization of the matroid median problem. Our algorithm achieves an approximation guarantee of 76 while violating the capacity constraints by a factor of at most 6. This will prove our main result, which is stated in Theorem 2.1.

In the next section, we describe the natural LP relaxation for CMM-UC. This LP has an exponential number of constraints but can be solved using the ellipsoid method, as we argue that one can construct a separation oracle using standard techniques. In Section 2.2 we give an overview of our algorithm and provide a complete description of our algorithm. Our algorithm involves three main steps, clustering, rounding to obtain a half-integral solution, and converting the half-integral solution to an integral solution. The analysis of this algorithm begins in Section 2.3 in which we discuss the properties that we obtain as a result of our clustering step. In Section 2.4, we discuss in detail and analyze the procedure of getting a half-integral opening vector for the facilities. Finally, in Section 2.5, we analyze the process of obtaining an integral opening vector from this half-integral opening vector.

The techniques used in our rounding algorithm build upon previous work on capacitated k-median and uncapacitated matroid median. There are quite a few similarities to the rounding procedure used for uncapacitated matroid median, but we have to work harder to handle the complications introduced by capacities. Our initial clustering step is also similar to that of capacitated k-median but the presence of the general matroid constraint prohibits us from moving around facility weight locally between clusters and requires a more global approach.

2.1 An LP relaxation for CMM-UC

We will use the variables $\{y_i\}_{i \in \mathcal{F}}$ for each facility $i \in \mathcal{F}$ to indicate whether facility i is open or not. Variables $\{x_{ij}\}_{i \in \mathcal{F}, j \in \mathcal{D}}$ will indicate if client j is assigned to facility i. Using these variables, we can express the matroid median problem as an integer program and then relax the integrality constraints to obtain the following linear program. Throughout, we use i to index facilities in $\mathcal{F}$, and j to index clients in $\mathcal{D}$.

$$\text{minimize} \sum_i f_i y_i + \sum_j d_j \sum_i c_{ij} x_{ij} \tag{P}$$

$$\text{subject to } \sum_i x_{ij} \geq 1 \qquad \forall j \in \mathcal{D} \tag{1}$$

$$\sum_j d_j x_{ij} \leq U y_i \qquad \forall i \in \mathcal{F} \tag{2}$$

$$\sum_{i \in S} y_i \leq \mathrm{r}(S) \qquad \forall S \subseteq \mathcal{F} \tag{3}$$

$$x_{ij} \leq y_i \qquad \forall i \in \mathcal{F}, j \in \mathcal{D} \tag{4}$$

$$0 \leq x_{ij}, y_i \leq 1 \qquad \forall i \in \mathcal{F}, j \in \mathcal{D} \tag{5}$$

Constraint (1) says that every client should be assigned and constraint (4) ensures that a client is assigned to a facility only if that facility is open. Constraint (2) ensures that the total number of clients assigned to a facility is at most its capacity. Finally, constraint (3) ensures that the open set of facilities is independent. Note that any solution to CMM-UC translates to an integer solution to (P).

Although the above LP has exponential number of constraints, we can separate over it in polynomial time, as follows. Consider an arbitrary fractional solution (x, y). We can easily check if the solution satisfies constraints (1), (2) and (4) in polynomial time. To check if the vector $\{y_i : i \in \mathcal{F}\}$ satisfies constraint (3), it is equivalent to checking the following condition:

$$\min_{S \subseteq \mathcal{F}} \left\{ \mathrm{r}(S) - \sum_{i \in S} y_i \right\} \geq 0$$

The function $f(S) = \mathrm{r}(S) - \sum_{i \in S} y_i$ is submodular as the matroid rank function, $\mathrm{r}(.)$, is submodular. Hence, we can use a polynomial time algorithm for submodular function

minimization to check if constraint (4) is satisfied. Hence, with the use of a polynomial time algorithm for submodular function minimization, we can construct an efficient separation oracle for the above LP and use the ellipsoid method to solve the LP in polynomial time. There are more efficient ways of constructing separation oracles for the matroid polytope.

The following definition will be useful and will be referred to throughout the thesis.
β-**violated solution.** A solution (x', y') is a β-violated solution if (x', y') satisfies constraints (1), (3), (4), (5) and violates constraint (2) by a factor of at most β.

2.2   Overview of the Algorithm

Our algorithm is based on rounding an optimal solution to (P). Hence given an instance of CMM-UC, we solve the natural LP relaxation and obtain the optimal solution (x, y). For a client j, let $C_j = \sum_{i \in \mathcal{F}} c_{ij} x_{ij}$ denote the unit assignment cost of j under the assignment x. Let OPT denote the value of the optimum LP solution (x, y). Note that the solution (x, y) is a 1-violated solution. Thus OPT $= \sum_{i \in \mathcal{F}} f_i y_i + \sum_{j \in \mathcal{D}} d_j \sum_{i \in \mathcal{F}} c_{ij} x_{ij} = \sum_{i \in \mathcal{F}} f_i y_i + \sum_{j \in \mathcal{D}} d_j C_j$. For a set $S \subseteq \mathcal{F}$, we use $y(S)$ to denote $\sum_{i \in S} y_i$.

Our focus will be on rounding to obtain an integral facility-opening vector; given such an opening vector, we can solve a min-cost flow problem to obtain the assignments for the clients $\mathcal{D}$. We prove the following theorem, which is a more formal restatement of Theorem 1.1.

Theorem 2.1. *We can efficiently round (x, y) to a 6-violated solution $(\tilde{x}, \tilde{y})$, where $\tilde{y}$ is integral, of cost at most 76OPT. Thus, we obtain an approximation guarantee of 76 while violating capacities by a factor of at most 6.*

At a high level, the algorithm consists of three components: (1) *Clustering* to move (roughly speaking) to a more structured instance. (2) Leveraging the structure obtained after the clustering to *round to a half-integral solution* and (3) rounding the *half-integral solution* to an *integral solution.*

Clustering the original instance. First, as is standard for rounding algorithms for k-median and its variants, we use the LP solution to transform our instance to a more structured instance. That is, we will use (x, y) to cluster clients and facilities around certain cluster centers (which will also be clients). The properties that we seek from the clustering are (i) that the cluster centers should lie far apart, which will ensure that each cluster will have a certain minimum facility weight, and (ii) every client should be 'near' a cluster center. We use $\mathcal{D}'$ to refer to the set of cluster centers. We then assign every

facility to the cluster center nearest to it. The set of facilities assigned to a cluster center k are referred to as F_k (we will often refer to F_k also as k's cluster) : property (i) will ensure that F_k will have a total facility weight of at least $\frac{1}{2}$, that is $y(F_k) \geq \frac{1}{2}$. We will sparsify our instance by 'moving' demand from the original client locations to the cluster centers, thereby obtaining a more structured instance where all the demand is aggregated at the cluster centers. In this process of moving the demands, a client j might split it's demand across multiple cluster centers, property (ii) will ensure that the total movement cost is at most $6\mathtt{OPT}$ (see Lemma 2.4). Let D_k denote the demand aggregated at cluster center $k \in \mathcal{D}'$ after this movement, the movement process ensures that $D_k \leq Uy(F_k)$ for every $k \in \mathcal{D}'$.

Given this clustering, we would like to treat each cluster independently, but various difficulties arise in doing so. The facility weight $y(F_k)$ for $k \in \mathcal{D}'$ could be less than 1, and so we will not be able to handle the demand D_k using facilities solely in F_k without blowing up $y(F_k)$, but doing so could violate the matroid constraints. This difficulty also arises in uncapacitated k-median (and matroid median) but some rounding algorithms for capacitated k-median follow such an approach to obtain a guarantee that violates the cardinality constraint by an $O(1)$ factor. However, such a cluster-by-cluster approach is not viable for CMM since the matroid constraint is a global constraint and could couple the different clusters. Instead, as in k-median and uncapacitated matroid median, since the clustering step will ensure $y(F_k) \geq \frac{1}{2}$ for every $k \in \mathcal{D}'$, we will first seek to obtain and half-integral solution.

In doing so, one subtle point of departure with respect to UMM is that, there is no simple way of translating the LP solution (x, y) for the original instance to a fractional solution to the clustered instance. In UMM, the assignment x restricted to the cluster centers (and with the new demands $\{D_k\}_{k \in \mathcal{D}'}$) yields a feasible fractional solution of cost at most $\mathtt{OPT}$, which is convenient as it allows one to solely work with the structured instance. Here however, this restriction is no longer a feasible fractional solution as the demand D_k of $k \in \mathcal{D}'$ could be larger than d_k. Another possible fractional assignment for the structured instance is setting $x'_{ik} = \frac{\text{total demand served by } i}{D_k}$; x' completely assigns the D_k demand and satisfies the capacity constraints (i.e. constraints (1) and (2) in (P) are satisfied), but it may no longer be that $x'_{ik} \leq y_i$ [1]. Thus, while the structured instance with the clusters will be quite useful to us, we cannot quite work solely with this instance and forget about the original instance (like in UMM). We will however, round (x, y) to obtain a 3-violated half-integral solution and then round this to a 6-violated integral solution. We proceed to give an overview of these two steps.

---

[1]For example, consider a cluster F_k with only one facility i; an assignment such as x' will ensure that $x'_{ik} = 1$ regardless of the value of y_i

Getting a half-integral solution. An important consequence of property (i) is that in each cluster, the LP has opened a facility weight of at least $\frac{1}{2}$. We will exploit this and obtain a rounded solution that opens a facility weight of at least $\frac{1}{2}$ in each cluster as well. As noted earlier, for clusters with $y(F_k) < 1$, we will not be able to open a facility-weight of at least 1 in F_k, and so will not be able to assign the D_k demand solely to facilities in F_k. If our rounded solution $\hat{y}$ ends up opening a facility weight less than 1 in F_k, even if allocate $\hat{y}_i$ fraction of the demand D_k to facility i, there will be a $1 - \hat{y}(F_k)$ fraction of the demand left to allocate. To handle such clusters, we will assign a $(1 - \hat{y}(F_k))$-fraction of the demand D_k to (facilities opened from) a "neighboring" cluster center (defined suitable later). Since each cluster has a weight of at least $\frac{1}{2}$ in the rounded solution, we can ensure that between the two clusters, we have enough facility weight to distribute the demand D_k completely. We will choose the neighboring cluster center carefully so that (a) the cost of assigning the requisite fraction of the D_k-demand to the neighbor is bounded, and (b) a cluster is designated as a neighbor of at most 2 clusters, so that its facilities are not too overloaded by the demands of too many other clusters. This latter consideration does not arise in UMM, and is another place where we need to diverge from the algorithm and analysis of UMM. For clusters with $y(F_k) \geq 1$, we will ensure that that we can open a $\max(1, \lfloor \frac{D_k}{U} \rfloor)$ facility-weight in F_k, which will ensure that we can assign the D_k demand to the facilities opened from F_k while violating capacities by a factor of at most 2.

To precisely define how we round to a half-integral solution, for each cluster center $k \in \mathcal{D}'$, we will define a function T_k that will act as a proxy for the assignment cost of D_k in the rounded (half-integral solution). Also, each cluster F_k will give rise to certain constraints; we will append these with the matroid rank constraints to obtain a polytope $\widehat{P} \subseteq \mathbb{R}^{\mathcal{F}}$ (see Step A4 in Algorithm 1). In Lemma 2.6 and 2.8 we show that there exists a $z^* \in \widehat{P}$ such that $\sum_{i \in \mathcal{F}} f_i z_i^* + \sum_{k \in \mathcal{D}'} T_k(z^*)$ is $O(\mathtt{OPT})$. The constraints of $\widehat{P}$ are structured enough to conclude half-integrality of $\widehat{P}$ (see Lemma 2.9). Thus by minimizing $T(z) = \sum_{i \in \mathcal{F}} f_i z_i + \sum_{k \in \mathcal{D}'} T_k(z)$ over $\widehat{P}$, we can obtain a half-integral facility-opening vector $\hat{y}$ with $T(\hat{y}) \leq T(z^*) = O(\mathtt{OPT})$. We show how to use this to obtain a half-integral assignment vector $\hat{x}$ (this will involve making copies of some cluster centers; see Algorithm 3). This gives our half-integral solution $(\hat{x}, \hat{y})$. We prove that $T_k(\hat{y})$ gives an upper bound on the cost of assigning D_k demand under the facility-opening vector $\hat{y}$, and that the cost of $(\hat{x}, \hat{y})$ is at most $T_k(\hat{y}) = O(\mathtt{OPT})$ (Lemma 2.11). Also, we show that the capacity violation incurred is at most a factor of 3 (Lemma 2.12), so that $(\hat{x}, \hat{y})$ is a 3-violated solution of cost $O(\mathtt{OPT})$.

Getting an integral solution. A useful consequence of having a half-integral assignment vector $\hat{x}$ is that the solution is filtered − which means that if a cluster center k is assigned to a facility i, then $c_{ik} \leq 2 \sum_{i \in \mathcal{F}} c_{ik} \hat{x}_{ik} = 2 \hat{C}_k$. We use a standard facility location clustering

10

to further prune $\mathcal{D}'$ (the cluster centers obtained via the initial clustering) to a subset $\tilde{\mathcal{D}}$ such that the facilities serving distinct clients in $\tilde{\mathcal{D}}$ are disjoint, and every $k \in \mathcal{D}' \setminus \tilde{\mathcal{D}}$ shares a facility with some nearby $j \in \tilde{\mathcal{D}}$. Given this, it is easy to write down another proxy program, this time involving minimizing a linear function over an *integral polytope*, and solving this yields an integral facility-opening vector which leads to a solution where capacities are (further) violated by a factor of 2 (Lemma 2.14). We now give a detailed description of the algorithm, and then proceed to analyze the algorithm.

Algorithm 1: Algorithm for CMM-UC

A1.   Obtain optimal LP solution (x, y) by solving program (P).
A2.   <u>Clustering.</u> - $\mathcal{D}' := \emptyset$.
- Consider clients j in increasing order of C_j. If for client j,
 there exists no client $k \in \mathcal{D}'$ such that $c_{jk} \leq 4C_j$, then $\mathcal{D}' = \mathcal{D}' \cup \{j\}$.
A3.   - Define $F_k = \{i \in \mathcal{F} : c_{ik} = \min_{j \in \mathcal{D}'} c_{ij}\}$ for each $k \in \mathcal{D}'$.
- For each $k \in \mathcal{D}'$, $D_k = \sum_{j \in \mathcal{D}} \sum_{i \in F_k} d_j x_{ij}$ (Intuitively, D_k is the total
 demand moved from the original client locations to cluster center k).
A4.   <u>Obtaining a half-integral solution.</u>
- Define $\overline{F'_k = \{i \in F_k : c_{ik} \leq 2C_k\}}$ for each $k \in \mathcal{D}'$.
- Define $\gamma_k = \min_{i \notin F_k} c_{ik}$ for each $k \in \mathcal{D}'$.
- Define $G_k = \{i \in F_k : c_{ik} \leq \gamma_k\}$ for each $k \in \mathcal{D}'$.
- Partition the set of cluster centers as follows:

  - $\mathcal{D}_1 = \{k \in \mathcal{D}' : D_k < U \text{ and } y(G_k) > 1\}$

  - $\mathcal{D}_3 = \{k \in \mathcal{D}' : D_k < U \text{ and } y(G_k) \leq 1\}$ r

  - $\mathcal{D}_2 = \{k \in \mathcal{D}' : D_k \geq U\}$

- Obtain half-integral opening vector $\hat{y}$ by solving the following
linear program,

$$\min_{z \in \widehat{P}} f_i z_i + \sum_{k \in \mathcal{D}'} T_k(z) \text{ ,where}$$

$$T_k(z) = \begin{cases} D_k \sum_{i \in F_k} c_{ik} z_i & k \in \mathcal{D}_1 \\ 2U \sum_{i \in F_k} c_{ik} z_i & k \in \mathcal{D}_2 \\ D_k \sum_{i \in G_k} c_{ik} z_i + D_k(1 - z(G_k))5\gamma_k & k \in \mathcal{D}_3 \end{cases}$$

$$\widehat{P} = \left\{ z(F'_k) \geq \frac{1}{2} \quad \forall k \in \mathcal{D}', \quad z(F_k) \geq \max\left(1, \lfloor \frac{D_k}{U} \rfloor\right) \quad \forall k \in \mathcal{D}_1 \cup \mathcal{D}_2, \right.$$

$$\left. z(G_k) \leq 1 \quad \forall k \in \mathcal{D}_3, \quad z(S) \leq r(S) \quad \forall S \subseteq \mathcal{F} \right\}$$

A5.   Obtain assignment $\hat{x}$ for the clustered instance using Algorithm 3
(page 26) on input $\hat{y}$.
A6.   Use Algorithm 4 (page 29) to obtain an integral solution for the
clustered instance on input $(\hat{x}, \hat{y})$.

2.3   Properties of the clustered instance

We now show that the clustering step leads to a structured instance with various useful properties. In particular, we will argue that every cluster F_k has facility-weight at least $\frac{1}{2}$, and that the cost of aggregating demands at the cluster centers is at most $6\mathsf{OPT}$ (Lemma 2.4).

For a subset of facilities $F \subseteq \mathcal{F}$, let $x(F, j) = \sum_{i \in F} x_{ij}$ denote the fraction of j's demand served by facilities in F.

Performing the clustering as in Step A2 of Algorithm 1 gives us the following properties:

Lemma 2.2. *The set $\mathcal{D}'$ of cluster centers satisfies the following two properties:*

(i) if $k, k' \in \mathcal{D}'$, then $c_{kk'} \geq 4\max(C_k, C_{k'})$.

(ii) $\forall j \in \mathcal{D}$, there exists $k \in \mathcal{D}'$ such that $c_{jk} \leq 4C_j$ and $C_k \leq C_j$.

Proof. (i) Without loss of generality, assume that k was considered before k' by the algorithm. k' was selected as a cluster center only because there was no $k'' \in \mathcal{D}'$ such that $c_{kk''} \leq 4C_k$. As k was already selected as a cluster center when considering k', this implies that $c_{kk'} \geq 4C_{k'} = 4\max(C_k, C'_k)$.

(ii) Consider a client $j \in \mathcal{D}$. If $j \in \mathcal{D}'$, then its distance to itself is 0 which is at most $4C_j$. Otherwise consider $j \in \mathcal{D} \backslash \mathcal{D}'$. As j was not selected as a cluster center, it means that at the time j was being considered, $\exists k \in \mathcal{D}'$ such that $c_{jk} \leq 4C_j$. As we never remove a center after it has been added to $\mathcal{D}'$, this means that there is a $k \in \mathcal{D}'$ such that $c_{jk} \leq 4C_j$. Moreover, as k was considered before j, it means that $C_k \leq C_j$. ∎

Let $nc : \mathcal{F} \cup \mathcal{D} \to \mathcal{D}'$ (for **n**earest **c**enter) be a function that, for $l \in \mathcal{F} \cup \mathcal{D}$, denotes the closest cluster center to l, i.e., $nc(l) = \arg\min_{k \in \mathcal{D}'} c_{lk}$ (Note that ties are broken arbitrarily, but in a consistent way). Hence, the F_k's defined in Algorithm 1 is simply, $F_k = \{i \in \mathcal{F} : nc(i) = j\}$. The partition defined by the F_k's satisfy the following useful property, which we will repeatedly use.

Claim 2.3. *Let i be a facility such that $i \in F_k$ for $k \in \mathcal{D}'$ and let j be any client in $\mathcal{D}$. Then, $c_{ik} \leq c_{ij} + 4C_j$*

Proof. As $i \in F_k$, the distance $c_{ik} \leq c_{ik'}$ for any $k' \in \mathcal{D}'$. Thus, $c_{ik} \leq c_{nc(j)i} \leq c_{ij} + c_{nc(j)j} \leq c_{ij} + 4C_j$, where the last inequality follows from part (ii) of (Lemma 2.2). ∎

We now show that the cost of moving the demands from the clients to the cluster centers according to step A3 in Algorithm 1 is bounded.

Lemma 2.4. *The movement cost* $\sum_{k \in \mathcal{D}'} (d_j \cdot x(F_k, j)) c_{jk}$ *incurred in moving demand* d_j *from* j *to the cluster centers is at most* $6 d_j C_j$.

Proof. For each cluster center $k \in \mathcal{D}'$, j sends $d_j \cdot x(F_k, j)$ amount of demand to k. Hence, the movement cost is

$$
\begin{aligned}
\sum_{k \in \mathcal{D}'} d_j x(F_k, j) c_{jk} &= \sum_{k \in \mathcal{D}'} d_j \sum_{i \in F_k} x_{ij} c_{jk} \\
&\leq \sum_{k \in \mathcal{D}'} d_j \sum_{i \in F_k} x_{ij} (c_{ik} + c_{ij}) && \text{(triangle inequality)} \\
&\leq \sum_{k \in \mathcal{D}'} d_j \sum_{i \in F_k} x_{ij} (2c_{ij} + 4C_j) && \text{(Claim 1.3)} \\
&= 2 d_j \sum_i c_{ij} x_{ij} + 4 d_j C_j \sum_i x_{ik} \\
&= 6 d_j C_j
\end{aligned}
$$

∎

Summing up across all clients, we see that the total movement cost is at most $\sum_j 6 d_j C_j \leq 6\texttt{OPT}$.

Thus, we can conclude,

Corollary 2.5. *If there is* β-*violated solution to the clustered instance of cost at most* $\alpha\texttt{OPT}$, *then there is a* β-*violated solution to the original instance of cost at most* $(\alpha + 6)\texttt{OPT}$.

Proof. In order to obtain a solution to the original instance, we need to move the demands from the cluster centers back to the original clients. The cost of moving back j's demand is $\sum_{k \in \mathcal{D}'} d_j x(F_k, j) c_{jk}$, which according to Lemma 2.4 is at most $6 d_j C_j$. Summing up across all clients gives us $\sum_j 6 d_j C_j \leq 6\texttt{OPT}$. ∎

In the sequel, we focus on showing that our algorithm returns a 6-violated solution to the clustered instance of cost at most $70\texttt{OPT}$. By the above corollary, this immediately translates to a 6-violated solution to the original instance of cost at most $76\texttt{OPT}$, thereby yielding the approximation guarantee stated in Theorem 2.1.

14

2.4 Getting a half-integral solution

In this section, we explain how the proxy problem solved in step A4 arises, describe Algorithm 3 mentioned in step A5, and show how these lead to a half-integral solution $(\hat{x}, \hat{y})$.

Recall that for $k \in \mathcal{D}'$, we have $F_k' = \{i \in \mathcal{F} : c_{ik} \leq 2C_k\}$, $\gamma_k = \min_{i \notin F_k} c_{ik}$ and $G_k = \{i \in F_k : c_{ik} \leq \gamma_k\}$. Note that $F_k' \subseteq F_k$ because for all facilities i such that $c_{ik} \leq 2C_k$, the closest center to i has to be k, otherwise we would have a contradiction to part (i) of Lemma 2.2. We also defined γ_k as, $\gamma_k = \min_{i \notin F_k} c_{ik}$ as the distance between k and the facility closest to k but not in F_k and using this defined $G_k = \{i \in F_k : c_{ik} \leq \gamma_k\}$. Again, using part (i) of Lemma 2.2, we can see that $\gamma_k \geq 2C_k$; thus, $F_k' \subseteq G_k \subseteq F_k$.

Claim. For $k \in \mathcal{D}'$, $\sum_{i \in F_k'} y_i \geq \frac{1}{2}$.

Proof. This follows as a simple consequence of Markov's inequality. Markov's inequality states that for a non-negative random variable X and any constant $c > 0$, $\Pr[X \geq c] \leq \frac{\mathbb{E}[X]}{c}$. As $C_k = \sum_{i \in \mathcal{F}} c_{ij} x_{ij}$ and $\sum_{i \in \mathcal{F}} x_{ij} = 1$, we can see that $\sum_{i \in \mathcal{F}: c_{ij} \geq 2C_k} x_{ij} \leq \frac{1}{2}$. Thus, $\sum_{i \in \mathcal{F}: c_{ij} \leq 2C_k} x_{ij} = \sum_{i \in F_k'} x_{ij} \geq \frac{1}{2}$. From constraint (4) in (P) we know that $x_{ij} \leq y_i$ for all $i \in \mathcal{F}, j \in \mathcal{D}$. Thus, $\sum_{i \in F_k'} y_i \geq \sum_{i \in F_k'} x_{ij} \geq \frac{1}{2}$. ∎

For a set of $F' \subseteq \mathcal{F}$ of facilities, let $cost_1(F') = \sum_j d_j \sum_{i \in F'} c_{ij} x_{ij}$ be the cost of assigning demands to facilities in F' under the LP solution (x, y). Also, define $cost_2(F') = \sum_j d_j C_j \cdot x(F', j)$ to be the fraction of the total assignment cost proportional to the amount of demand assigned to facilities in F' under the LP solution (x, y). Hence, $\sum_{i \in \mathcal{F}} f_i y_i + cost_1(\mathcal{F}) = \sum_{i \in \mathcal{F}} f_i y_i + cost_2(\mathcal{F}) = \text{OPT}$.

Explanation and analysis of the proxy problem in step A4. We obtain a half-integral solution in step A4 of Algorithm 1 by minimizing the function $T(z)$ given by $\sum_{i \in \mathcal{F}} f_i z_i + \sum_{k \in \mathcal{D}'} T_k(z)$ over the polytope $\widehat{P}$, where $T_k(z)$ is a function that acts as a convenient proxy for the cost incurred in assigning D_k demand to the facilities opened (fractionally) under z (Lemma 2.11). More precisely, we will ensure that: (a) there is a point $z^* \in \widehat{P}$ such that $T(z^*) \leq 35\text{OPT}$ (Lemma 2.8); and (b) extreme points of $\widehat{P}$ are half-integral (Lemma 2.9). Thus optimizing T over $\widehat{P}$ will yield a half integral $\hat{y} \in \widehat{P}$, and the proxy function is set up so that we further have: (c) using $\hat{y}$, we can obtain a half-integral solution of cost at most $T(\hat{y})y) \leq T(z^*) \leq 35\text{OPT}$.

As mentioned in Algorithm 1, $\widehat{P}$ contains some constraints for each cluster F_k, as well as the matroid constraints. The definition of T_k, and the constraints that we include for cluster k, will depend on the type of cluster k. The cluster centers $\mathcal{D}'$ were partitioned into the following three types of clusters:

- $\mathcal{D}_1 = \{k \in \mathcal{D}' : D_k < U \text{ and } y(G_k) > 1\}$

- $\mathcal{D}_3 = \{k \in \mathcal{D}' : D_k < U \text{ and } y(G_k) \leq 1\}$

- $\mathcal{D}_2 = \{k \in \mathcal{D}' : D_k \geq U\}$

As we have $y(F_k') \geq \frac{1}{2}$ for every $k \in \mathcal{D}'$, we want to ensure that this is true for all clusters in the rounded solution as well and hence, we will add this constraint to $\widehat{P}$. For cluster centers $\mathcal{D}_1 \cup \mathcal{D}_2$ we also have that $y(F_k) \geq \max\left(1, \lfloor \frac{D_k}{U} \rfloor\right)$ and therefore we incorporate these constraints into $\widehat{P}$ as well. And finally, for $k \in \mathcal{D}_3$, we know that $y(G_k) \leq 1$. Thus, $\widehat{P}$ is defined as:

$$\widehat{P} = \left\{ z(F_k') \geq \frac{1}{2} \quad \forall k \in \mathcal{D}', \quad z(F_k) \geq \max\left(1, \lfloor \frac{D_k}{U} \rfloor\right) \quad \forall k \in \mathcal{D}_1 \cup \mathcal{D}_2, \right.$$

$$\left. z(G_k) \leq 1 \quad \forall k \in \mathcal{D}_3, \quad z(S) \leq r(S) \quad \forall S \subseteq \mathcal{F} \right\}$$

For $k \in \mathcal{D}_1 \cup \mathcal{D}_2$, we will be able to use the facilities opened from F_k to assign the D_k demand, and $T_k(.)$ will encode the cost incurred in doing so. For $k \in \mathcal{D}_3$, we will need to assign some fraction of the D_k-demand outside of k's cluster, but we will show that we can ensure that this portion is assigned to a facility i with $c_{ik} \leq 5\gamma_k$. Recall that, the proxy function $T_k(.)$ for cluster k is defined as:

$$T_k(z) = \begin{cases} D_k \sum_{i \in F_k} c_{ik} z_i & k \in \mathcal{D}_1 \\ 2U \sum_{i \in F_k} c_{ik} z_i & k \in \mathcal{D}_2 \\ D_k \sum_{i \in G_k} c_{ik} z_i + D_k(1 - z(G_k))5\gamma_k & k \in \mathcal{D}_3 \end{cases}$$

We now provide some intuition for the above definition. For clusters defined by centers $k \in \mathcal{D}_1$, we have ensured that a facility weight of at least 1 is always opened within the cluster, that is, for these clusters we know that $z(F_k) \geq 1$. Under this constraint, the term $D_k \sum_{i \in F_k} c_{ik} z_i$ provides an upper bound on k's assignment cost.

For centers $k \in \mathcal{D}_2$, we have ensured that the facilities in F_k are open to an extent of at least $\lfloor \frac{D_k}{U} \rfloor$. That is, for these clusters we know that, $z(F_k) \geq \lfloor \frac{D_k}{U} \rfloor$. As $D_k \geq U$ for for $k \in \mathcal{D}_2$, opening a facility weight of $\lfloor \frac{D_k}{U} \rfloor$ will (1) allow us to handle all the demand within the cluster (as $z(F_k) \geq 1$), (2) with at most a factor 2 violation in capacities, there

is enough capacity in the cluster F_k to handle all the demand within the cluster. Hence, under this constraint, the term $2U \sum_{i \in F_k} c_{ik} z_i$ provides an upper bound on the assignment cost of demand D_k as $\left(2U \cdot \lfloor \frac{D_k}{U} \rfloor\right) \geq D_k$.

For clusters defined by centers $k \in \mathcal{D}_3$, we have ensured that the facilities in F'_k are open to an extent of at least $\frac{1}{2}$ and that facilities in G_k are open to an extent of at most 1. That is, for $z \in \mathbb{R}^{\mathcal{F}}_+$, we want $z(F'_k) \geq \frac{1}{2}$ and $z(G_k) \leq 1$. Given such an opening vector, only a $z(G_k) \geq \frac{1}{2}$ fraction of the demand can be distributed within the cluster F_k. We will show that the residual demand can be assigned to a facility $i' \in F'_{k'}$ where $k' \in \mathcal{D}' \setminus \{k\}$ and $c_{i'k} \leq 5\gamma_k$. The constraint $z(F_l) \geq \frac{1}{2}$ for all $l \in \mathcal{D}'$ ensures that there is always a facility open to an extent of at least $1/2$ in F'_l for every $l \in \mathcal{D}'$. Hence, under these constraints, the term $D_k \sum_{i \in G_k} c_{ik} z_i + D_k (1 - z(G_k)) 5\gamma_k$ is an upper bound on the assignment cost of demand D_k located at k.

The opening vector z^*. We now define an opening vector $z^* \in \mathbb{R}^{\mathcal{F}}$. We will show that $z^* \in \widehat{P}$ and that $T(z^*)$ is bounded in terms of $\mathtt{OPT}$. The openings for the facilities depend on the type of cluster the facility is present in.

For facilities $i \in F_k$ where $k \in \mathcal{D}_1$. Order the facilities in F_k in increasing order of distance from the center k. Let $y_{i_1}, y_{i_2}, \ldots, y_{i_{|F_k|}}$ be such an ordering and let i_l be the nearest facility to k such that $y_{i_1} + \ldots + y_{i_{l-1}} + y_{i_l} \geq 1$. Define the opening as follows:

$$
z^*_{i_r} = \begin{cases} y_{i_r} & \text{if } r < l \\ 1 - (y_{i_1} + \ldots + y_{i_{l-1}}) & \text{if } r = l \\ 0 & \text{otherwise} \end{cases}
$$

For facilities $i \in F_k$ where $k \in \mathcal{D}_2$,

$$
z^*_i = \begin{cases} \max\left(x_{ik}, \frac{\sum_j d_j x_{ij}}{U}\right) & \text{if } i \in F'_k \\ \frac{\sum_j d_j x_{ij}}{U} & \text{if } i \in F_k \setminus F'_k \end{cases}
$$

Finally, for facilities $i \in F_k$ where $k \in \mathcal{D}_3$,

$$
z^*_i = \begin{cases} y_i & \text{if } i \in G_k \\ 0 & \text{otherwise} \end{cases}
$$

It is easy to see that z^* satisfies the constraints in $\widehat{P}$.

Lemma 2.6. $z^* \in \widehat{P}$.

17

Proof. For each cluster center $k \in \mathcal{D}'$, $z^*(F_k') \geq \sum_{i \in F_k'} x_{ik}$ which, from Markov's inequality, we know is at least $\frac{1}{2}$. For cluster centers $k \in \mathcal{D}_3$, $z^*(G_k) = y(G_k) \leq y(F_k) \leq 1$. For $k \in \mathcal{D}_1$, according to the procedure of opening facilities, $z^*(G_k) = 1$. Finally for $k \in \mathcal{D}_2$, as $z_i^*(F_k) \geq \frac{\sum_j d_j x_{ij}}{U}$ for each $i \in F_k$. Thus, $z^*(F_k) \geq \frac{D_k}{U}$. As $D_k \geq U$ for centers in $\mathcal{D}_2$, this finishes the proof. ∎

We bound the cost of $T(z^*)$ by showing that $T_k(z^*)$ for each center $k \in \mathcal{D}'$ can be bound in terms of $cost_1(F_k)$ and $cost_2(F_k)$, the cost that the LP pays for assigning demand to the facilities in F_k. Summing up $T_k(z^*)$ across all k hence will bound $T(z)$ in terms of OPT.

Lemma 2.7. *Consider a cluster center $k \in \mathcal{D}'$ with demand D_k. The optimal value of the program,*

$$min \quad \sum_{i \in \mathcal{F}} D_k c_{ik} z_i \qquad s.t. \qquad z(\mathcal{F}) \geq 1, \quad 0 \leq z_i \leq y_i \quad \forall i \in \mathcal{F} \qquad (R_k)$$

is at most $2cost_1(F_k) + 5cost_2(F_k)$.

Proof. Note that the solution to the above linear program is simple. Sort the facilities in increasing order of distance from k. Let $i_1, i_2, \ldots, i_{|\mathcal{F}|}$ be such an ordering. Let l be the smallest integer such that $\sum_{p=1}^l y_i \geq 1$. The optimal solution to the above linear program is setting $z_{i_p}^* = y_{i_p}$ for $1 \leq p \leq l-1$, setting $z_{i_l}^* = 1 - \sum_{p=1}^{l-1} y_i$ and setting $z_i^* = 0$ otherwise.

A key step in upper bounding the cost of this solution is to notice that for each $j \in \mathcal{D}$, setting $z_i = x_{ij}$ also produces a feasible solution to R_k and hence the cost of this solution is at least the cost of z^*.

We analyze the cost of solution z^* by expanding out D_k as $\sum_{j \in \mathcal{D}} d_j \cdot x(F_k, j)$, and then bounding the contribution of client j by considering the assignment $\{x_{ij}\}_{i \in \mathcal{F}}$ instead.

$$D_k \sum_{i \in \mathcal{F}} c_{ik} z_i^* = \sum_{j \in \mathcal{D}} d_j x(F_k, j) \sum_{i \in \mathcal{F}} c_{ik} z_i^*$$

$$\leq \sum_{j \in \mathcal{D}} d_j x(F_k, j) \sum_i c_{ik} x_{ij}$$

$$\leq \sum_{j \in \mathcal{D}} d_j x(F_k, j) \sum_i (c_{ij} + c_{jk}) x_{ij} \qquad \text{(triangle inequality)}$$

$$= \sum_{j \in \mathcal{D}} d_j x(F_k, j) \sum_i c_{ij} x_{ij} + \sum_{j \in \mathcal{D}} d_j x(F_k, j) \sum_i c_{jk} x_{ij}$$

Notice that the first term is just $\sum_{j \in \mathcal{D}} d_j x(F_k, j) C_j$ from the definition of C_j, and so $\sum_{j \in \mathcal{D}} d_j x(F_k, j) C_j = cost_2(F_k)$.

For the second term,

$$\sum_{j \in \mathcal{D}} d_j c_{jk} x(j, F_k) \leq \sum_{j \in \mathcal{D}} d_j \sum_{i \in F_k} (c_{ij} + c_{ik}) x_{ij} \qquad \text{(triangle inequality)}$$

$$\leq cost_1(F_k) + d_j \sum_{i \in F_k} c_{ik} x_{ij}$$

$$\leq cost_1(F_k) + d_j \sum_{i \in F_k} (c_{ij} + 4C_j) x_{ij} \qquad \text{(Claim 2.3)}$$

$$= cost_1(F_k) + 5 cost_2(F_k)$$

Thus, $D_k \sum_{i \in \mathcal{F}} c_{ik} z_i^* \leq 2 cost_1(F_k) + 5 cost_2(F_k)$. ∎

We will use the above lemma to bound the cost of $T(z^*)$ in terms of OPT.

Lemma 2.8. $T(z^*) \leq 35\text{OPT}$

Proof. **For centers defined by** $k \in \mathcal{D}_1$**.** Notice that $T_k(z^*)$ is exactly the optimum value of R_k from Lemma 2.7. So, $T_k(z^*) \leq 2 cost_1(F_k) + 5 cost_2(F_k)$ for $k \in \mathcal{D}_1$.

For centers defined by $k \in \mathcal{D}_3$. According to the definition of γ_k, for every facility $i \notin G_k$, $c_{ik} \geq \gamma_k$. Thus, $D_k \sum_{i \in G_k} c_{ik} z_i^* + D_k(1 - z(G_k))\gamma_k$ is a *lower bound* on the optimum value of R_k. So, by Lemma 2.7, we have Hence,

$$T_k(z^*) = D_k \sum_{i \in G_k} c_{ik} z_i^* + D_k(1 - z^*(G_k))(5\gamma_k) \leq 5(D_k \sum_{i \in G_k} c_{ik} z_i^* + \sum_j d_j x(j, F_k)(1 - z^*(G_k))\gamma_k)$$
$$\leq 5(2cost_1(F_k) + 5cost_2(F_k))$$
$$= 10cost_1(F_k) + 25cost_2(F_k)$$

Thus, for clusters defined by centers $k \in \mathcal{D}_3$, $T_k(z^*) \leq 12cost_1(F_k) + 30cost_2(F_k)$.

Finally, for clusters defined by centers $k \in \mathcal{D}_2$, we will show that $T_k(z^*) \leq 4cost_1(F_k) + 16cost_2(F_k)$.

For the analysis, we partition the facilities in F_k based on our choice of z^* as follows,

Let $F_k^1 = \{i \in F_k \mid z_i^* = x_{ik}\}$ and $F_k^2 = F_k \setminus F_k^1$.

Thus,

$$T_K(z^*) = 2U \sum_{i \in F_k} c_{ik} z_i^*$$
$$= 2U \underbrace{\sum_{i \in F_k^1} c_{ik} x_{ik}}_{a} + 2U \underbrace{\sum_{i \in F_k^2} c_{ik} \cdot \frac{\sum_j d_j x_{ij}}{U}}_{b}$$

We bound the above expression in a term-by-term fashion,

Consider term 'a'. Note that $F_k^1 \subseteq F_k'$, and hence, for every $i \in F_k^1, c_{ik} \leq 2C_k$.

Thus, $2U \sum_{i \in F_k^1} c_{ik} z_i^* = 2U \sum_{i \in F_k^1} c_{ik} x_{ik} \leq 4UC_k \sum_{i \in F_k^1} x_{ik} \leq 4D_k C_k$, where the last inequality follows from $D_k \geq U$ for these clusters and the fact that $\sum_{i \in F_k^1} x_{ik} \leq 1$.

For client in $k \in \mathcal{D}'$, we define $nbr(k) = \arg\min_{j \in \mathcal{D}' \setminus \{k\}} c_{jk}$.

Expanding D_k as $\sum_j d_j x(F_k, j)$,

20

$$4D_k C_k = 4 \sum_{j \in \mathcal{D}} d_j x(F_k, j) C_k$$

$$= 4 \sum_{j \in \mathcal{D}: C_j \geq C_k} d_j x(F_k, j) C_k + 4 \sum_{j \in \mathcal{D}: C_j < C_k} d_j x(F_k, j) C_k \qquad (2.1)$$

$$\leq 4 \sum_{j \in \mathcal{D}: C_j \geq C_k} d_j C_j x(F_k, j) + \sum_{j \in \mathcal{D}: C_j < C_k} d_j x(F_k, j) c_{knbr(k)}$$

where the last inequality follows from part (i) of Lemma 2.2, that $c_{kk'} \geq 4 \max(C_k, C_{k'})$ for any $k, k' \in \mathcal{D}'$. For clients $j \in \mathcal{D}$ such that $C_j < C_k$, we can show that $c_{knbr(k)} \leq 2c_{ij} + 8C_j$ for any facility $i \in F_k$. This is because, as $C_j < C_k$, we know there is another cluster center $k'' \in \mathcal{D}'$ such that $k \neq k''$ and $c_{jk''} \leq 4C_j$. Thus:

$$
\begin{aligned}
c_{knc(j)} &\leq c_{kk''} \\
&\leq c_{kj} + c_{jk''} && \text{(triangle inequality)} \\
&\leq c_{kj} + 4C_j && \text{(Lemma 2.2)} \\
&\leq c_{ij} + c_{ik} + 4C_j && \text{(triangle inequality)} \\
&\leq 2c_{ij} + 8C_j && \text{(Claim 2.3)}
\end{aligned}
$$

Thus,

$$
\begin{aligned}
\sum_{j \in \mathcal{D}: C_j < C_k} d_j x(F_k, j) c_{knbr(k)} &= \sum_{j \in \mathcal{D}: C_j < C_k} d_j \sum_{i \in F_k} c_{knbr(k)} x_{ij} \\
&\leq \sum_{j \in \mathcal{D}: C_j < C_k} d_j \sum_{i \in F_k} (2c_{ij} + 8C_j) x_{ij}
\end{aligned}
$$

Finally, plugging this into (2.1),

$$4 \sum_{j \in \mathcal{D}: C_j \geq C_k} d_j C_j x(F_k, j) + \sum_{j \in \mathcal{D}: C_j < C_k} d_j x(F_k, j) c_{knbr(k)}$$

$$\leq 4 \sum_{j \in \mathcal{D}: C_j \geq C_k} d_j C_j x(F_k, j) + 8 \sum_{j \in \mathcal{D}: C_j < C_k} d_j C_j \sum_{i \in F_k} x_{ij} + 2 \sum_{j \in \mathcal{D}: C_j < C_k} d_j \sum_{i \in F_k} c_{ij} x_{ij}$$

$$\leq 8 cost_2(F_k) + 2 cost_1(F_k)$$

21

Now, consider term 'b',

$$2U \sum_{i \in F_k^2} c_{ik} \frac{\sum_j d_j x_{ij}}{U} \leq 2 \sum_{i \in F_k^2} c_{ik} \sum_j d_j x_{ij}$$

$$= 2 \sum_{j \in \mathcal{D}} d_j \sum_{i \in F_k^2} c_{ik} x_{ij}$$

$$\leq 2 \sum_{j \in \mathcal{D}} d_j \sum_{i \in F_k^2} (c_{ij} + 4C_j) x_{ij} \qquad \text{(Claim 2.3)}$$

$$= 2 cost_1(F_k^2) + 8 cost_2(F_k^2)$$

Hence, combining the analysis for terms 'a' and 'b' gives:

$$2U \sum_{i \in F_k} c_{ik} z_i^* \leq 2 cost_1(F_k) + 8 cost_2(F_k) + 2 cost_1(F_k^2) + 8 cost_2(F_k^2)$$

$$\leq 2 cost_1(F_k) + 8 cost_2(F_k) + 2 cost_1(F_k) + 8 cost_2(F_k)$$

$$\leq 4 cost_1(F_k) + 16 cost_2(F_k)$$

As $\{F_k\}_{k \in \mathcal{D}'}$ form a partition of the facility set $\mathcal{F}$, summing up across all cluster centers,

$$T(z^*) = \sum_{i \in \mathcal{F}} f_i z_i^* + \sum_{k \in \mathcal{D}_1} T_k(z^*) + \sum_{k \in \mathcal{D}_2} T_k(z^*) + \sum_{k \in \mathcal{D}_3} T_k(z^*)$$

$$\sum_{i \in \mathcal{F}} f_i y_i + \sum_{k \in \mathcal{D}'} (10 cost_1(F_k) + 25 cost_2(F_k))$$

$$\leq 10 (\sum_{i \in \mathcal{F}} f_i y_i + cost_1(\mathcal{F})) + 25 cost_2(\mathcal{F}) \leq 35 \text{OPT}$$

$\blacksquare$

Lemma 2.9. $\widehat{P}$ *has half-integral extreme points.*

Proof. The proof of Lemma 2.9 is very similar to an analogous result in [Swa16] and we prove it here for completeness. The half-integrality of $\widehat{P}$ follows from the fact that every extreme point is defined by a linearly independent system of tight constraints comprising some $z(S) = \mathrm{r}(S)$ equalities corresponding to a laminar set system, and some $z(F_k') = \frac{1}{2}$

22

and $z(G_k) = 1$ for $k \in \mathcal{D}_3$, $z(F'_k) = \frac{1}{2}$ and $z(F_k) = 1$ for $k \in \mathcal{D}_1$, and some $z(F'_k) = \frac{1}{2}$ and $z(F_k) = \lfloor \frac{D_k}{U} \rfloor$ equalities. The constraint matrix of this system thus corresponds to equations coming from two laminar set systems; such a matrix is known to be totally unimodular. As the right hand side of our system is half-integral, from this we can conclude that $\widehat{P}$ has half-integral extreme points.

■

Thus, we can obtain a half-integral extreme point solution $\hat{y} \in \widehat{P}$ such that $T(\hat{y}) \leq T(z^*)$.

2.4.1 Obtaining a half-integral assignment vector $\hat{x}$ corresponding to $\hat{y}$

For cluster centers in $\mathcal{D}_3$, we know that $\frac{1}{2} \leq \hat{y}(F_k) \leq 1$ according to the constraints in $\widehat{P}$. Thus, we define a relation $\sigma : \mathcal{D}_3 \to \mathcal{D}'$ to handle the residual demand D_k in the case $\hat{y}(F_k) < 1$. We will define $\sigma(.)$ with the following properties in mind. (i) We want $\sigma(k)$ to be close to k as the cost incurred to move the residual demand will be $\sim c_{k\sigma(k)}$. (ii) We don't want a cluster center in $\mathcal{D}'$ to be defined as a neighbour for too many centers in $\mathcal{D}_3$ as we don't want the capacity violation incurred as a result of this extra demand to be too much.

Recall that for $k \in \mathcal{D}'$, we defined $nbr(k) = \min_{j \in \mathcal{D}' \setminus \{k\}} c_{jk}$.

The relation $\sigma : \mathcal{D}_3 \to \mathcal{D}'$ is defined as follows:

Algorithm 2: Assign neighbors

-Create G with $V(G) = \mathcal{D}'$ and $E(G) = \{(k, nbr(k)) : \forall k \in \mathcal{D}_3\}$
for *For each k_0 with an incoming arc* **do**
    Let $k_1, \ldots, k_l$ be children of k_0 ordered in increasing order of distance from k_0.
    (B1) Define $\sigma(k_1) = k_0$.
    (B2) Define $\sigma(k_r) = k_{r-1}$ for $2 \leq r \leq l$.
end

In the algorithm above, every component G is an out-tree rooted at either a node in $\mathcal{D}_1 \cup \mathcal{D}_2$ or a 2-cycle $(k, k')(k', k)$ in $\mathcal{D}_3$. As we break ties consistently, we cannot have cycles with length larger than 2.
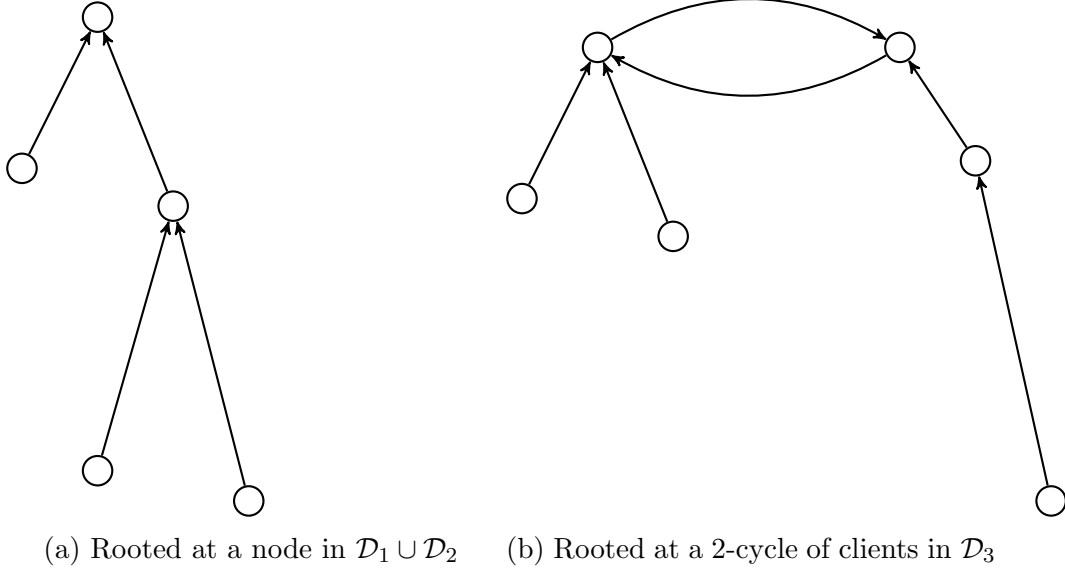
(a) Rooted at a node in $\mathcal{D}_1 \cup \mathcal{D}_2$ (b) Rooted at a 2-cycle of clients in $\mathcal{D}_3$

Figure 2.1: 2 types of components in G from Algorithm 2, an edge $u \to v$ in the diagram implies that $\sigma(u) = v$.

Lemma 2.10. *The relation $\sigma : \mathcal{D}_3 \to \mathcal{D}'$ satisfies the following properties:*

1. *$c_{k\sigma(k)} \leq 4\gamma_k$ for each $k \in \mathcal{D}_3$.*

2. *For any $k \in \mathcal{D}_3$ and $i \in F'_{\sigma(k)}$, $c_{ik} \leq 5\gamma_k$.*

3. *$|\sigma^{-1}(k)| \leq 1$ for every $k \in \mathcal{D}_1 \cup \mathcal{D}_2$.*

4. *$|\sigma^{-1}(k)| \leq 2$ for every $k \in \mathcal{D}_3$.*

Proof. (1) To see the first property, consider a node k_0 with at least one incoming arc and let $k_1, \ldots, k_l$ be its children ordered in increasing order of distance from k_0. According to our method of defining σ, for k_1, $c_{k_1\sigma(k_1)} = c_{k_1 nbr(k_1)}$. And for $k_r, 2 \leq r \leq l$, $c_{k_r\sigma(k_r)} \leq c_{nbr(k_r)k_{r-1}} + c_{nbr(k_r)k_r} \leq 2c_{nbr(k_r)k_r}$ as $nbr(k_{r-1}) = nbr(k_r)$ and $c_{nbr(k_r)k_{r-1}} \leq c_{nbr(k_r)k_r}$. Thus, for $k = k_r, 1 \leq r \leq l$, $c_{k\sigma(k)} \leq 2c_{knbr(k)}$. We now show that $c_{knbr(k)} \leq 2\gamma_k$. Note that for $\gamma_k = min_{i \notin F_k} c_{ik}$ and let i be the facility attaining this minimum. As $i \notin F_k$, there is another cluster center k' such that $i \in F_{k'}$ and $c_{ik'} \leq \gamma_k$. Thus, $c_{kk'} \leq 2\gamma_k$ and by definition of $nbr(k)$, $c_{knbr(k)} \leq c_{kk'} \leq 2\gamma_k$. Thus, $c_{k\sigma(k)} \leq 2c_{knbr(k)} \leq 4\gamma_k$.

24

(2) For $k \in \mathcal{D}_3$, let $i \in F'_{\sigma(k)}$, and by definition, $c_{i\sigma(k)} \leq 2C_{\sigma(k)}$. For cluster centers $k \in \mathcal{D}_3$ with $\sigma(k) = nbr(k)$, we show that $c_{ik} \leq 3\gamma_k$. For these clusters, $2C_{\sigma(k)} = 2C_{nbr(k)} \leq \frac{4\max\{C_{nbr(k)}, C_k\}}{2} \leq \frac{c_{knbr(k)}}{2} \leq \gamma_k$. Thus, $c_{ik} \leq c_{\sigma(k)k} + c_{i\sigma(k)} = c_{nbr(k)k} + c_{nbr(k)i} \leq 2\gamma_k + 2C_{nbr(k)} \leq 3\gamma_k$.

For cluster centers $k \in \mathcal{D}_3$ with $\sigma(k) \neq nbr(k)$, we will show that $c_{ik} \leq 5\gamma_k$. For these clusters, note that $c_{knc(k)} \geq c_{\sigma(k)nbr(k)}$. Thus, $2C_{\sigma(k)} \leq \frac{4\max\{C_{\sigma(k)}, C_nbr(k)\}}{2} \leq \frac{1}{2}c_{\sigma(k)nbr(k)} \leq \frac{1}{2}c_{knbr(k)} \leq \gamma_k$. Thus, $c_{ik} \leq c_{\sigma(k)k} + c_{i\sigma(k)} \leq 4\gamma_k + 2C_{\sigma(k)} \leq 5\gamma_k$.

For properties (3) and (4), consider a client $k \in \mathcal{D}_1 \cup \mathcal{D}_2$, k doesn't have an outgoing edge. As k can only have incoming edges, it is considered in step $(B1)$ at most once, and when considered, there exists only one $k' \in \mathcal{D}_3$ such that $\sigma(k') = k$.

For a client $k \in \mathcal{D}_3$, we know that k has exactly one outgoing edge and hence it is considered in step (B2) exactly once and moreover, in step (B2), there exists at most one k'' such that $\sigma(k'') = k$. Also, it is possible that k has incoming edges in G. Thus, it can also be considered in step (B1) . Thus for $k \in \mathcal{D}_3, \sigma^{-1}(k) \leq 2$. ∎

We are now ready to define the procedure to obtain half-integral $\hat{x}$.

Algorithm 3: Obtain half-integral assignments for clustered solution.

(A1) <u>For centers $k \in \mathcal{D}_1$.</u> Let $F' \subseteq F_k$ such that $\hat{y}(F_k) = 1$.
Set $\hat{x}_{ik} = \hat{y}_i$ for each $i \in F'$.

(A2) <u>For centers $k \in \mathcal{D}_2$.</u> Make $\lceil \frac{D_k}{U} \rceil$ copies of k.
Set the demand of $\lfloor \frac{D_k}{U} \rfloor$ of these copies to U. If $\frac{D_k}{U}$ is not
integral, set the demand of the remaining copy to
$frac(\frac{D_k}{U}) = \frac{D_k}{U} - \lfloor \frac{D_k}{U} \rfloor$.
Since $\hat{y}(F_k) \geq \lfloor \frac{D_k}{U} \rfloor$ and the capacities $\{U\hat{y}_i\}_{i \in F_k}$ are multiples of $\frac{U}{2}$,
We can assign the $\lfloor \frac{D_k}{U} \rfloor$ copies with demand U to facilities in F_k
such that the fraction of a copy's demand to facilities in F_k is
half-integral, and each facility $i \in F_k$ is assigned at most $U\hat{y}_i$
demand.
For the remaining copy with demand $frac(\frac{D_k}{U})$, if there is some $i \in F_k$
with $\hat{y}_i = 1$, assign this copy completely to i; otherwise, pick any
two facilities $i, i' \in F_k$ with $\hat{y}_i = \hat{y}_{i'} = \frac{1}{2}$, and assign a
$\frac{1}{2}$-fraction of this copy's demand to both i and i'.

(A3) <u>For centers $k \in \mathcal{D}_3$.</u> For each $i \in F_k$ with $\hat{y}_i > 0$, set $\hat{x}_{ik} = \hat{y}_i$.
If $\hat{y}(F_k) < 1$, set $x_{i'k} = 1 - \hat{y}(F_k)$ where $i' \in F'_{\sigma(k)}$.

For $k \in \mathcal{D}_2$, let $\mathcal{D}_{k,c}$ be the set containing copies of k and let $\mathcal{D}_{2,c} = \bigcup_{k \in \mathcal{D}_2} \mathcal{D}_{k,c}$. Finally, let $\mathcal{D}_c = \mathcal{D}_1 \cup \mathcal{D}_{2,c} \cup \mathcal{D}_3$. Thus, each cluster center in $\mathcal{D}_c$ has a half-integral assignment under $\hat{x}$.

Lemma 2.11. *The cost of the solution $(\hat{x}, \hat{y})$ is at most the cost of $T(\hat{y}) \leq T(z^*)$.*

Proof. The facility opening cost of the solution $(\hat{x}, \hat{y})$ is $\sum_{i \in \mathcal{F}} f_i \hat{y}_i$. To show that the cost of solution $(\hat{x}, \hat{y})$ is at most $T(\hat{y})$, what remains to be shown is that for each cluster center $k \in \mathcal{D}'$, the assignment cost is at most $T_k(\hat{y})$.

Consider a cluster center $k \in \mathcal{D}_1$. The assignment cost for k is $D_k \sum_{i \in F_k} c_{ik} \hat{x}_{ik}$, which according to how we've set up the assignments is at most $D_k \sum_{i \in F_k} c_{ik} \hat{y}_i = T_k(\hat{y})$.

Consider a cluster center $k \in \mathcal{D}_3$. If $\hat{y}(G_k) = 1$, then the assignment cost is $D_k \sum_{i \in G_k} c_{ik} \hat{x}_{ik} = D_k \sum_{i \in G_k} c_{ik} \hat{y}_i = T_k(\hat{y})$. Else, the assignment cost is $D_k \sum_{i \in G_k} c_{ik} \hat{x}_{ik} + D_k(1 - \hat{y}(G_k)) c_{i'k}$ where $i' \in F_{\sigma(k)}$. From property (2) of Lemma 2.10, we know that $c_{i'k} \leq 5\gamma_k$. Thus, the assignment cost under $\hat{x}$ is $D_k \sum_{i \in G_k} c_{ik} \hat{x}_{ik} + D_k(1 - \hat{y}(G_k))(5\gamma_k)$.

Finally, consider a cluster center $k \in \mathcal{D}_2$. Notice that as we are sending a demand of at most $2U\hat{y}_i$ to a facility $i \in F_k$, the total cost of such a solution is at most $\sum_{i \in F_k} c_{ik}(2U\hat{y}_i) = 2U \sum_{i \in F_k} c_{ik}\hat{y}_i = T_k(\hat{y})$. ∎

Lemma 2.12. *The solution $(\hat{x}, \hat{y})$ is a 3-violated solution for the clustered instance.*

Proof. Notice that according to the assignments, the solution $(\hat{x}, \hat{y})$ satisfies constraints (1), (3), (4) and (5). We will show that it violates constraints (2) by a factor of at most 3. Note that centers in $\mathcal{D}_3$ send at most a demand of $\frac{U}{2}$ outside the cluster.

Consider a cluster $k \in \mathcal{D}_1$. If there is no $k' \in \mathcal{D}_1$ such that $\sigma(k') = k$, then for each facility $i \in F_k, \sum_{j \in \mathcal{D}'} D_j \hat{x}_{ij} \le U\hat{y}_i$ as k is the only cluster center assigning its demand to facilities in F_k, and $D_k \le U$. Otherwise, there exists only 1 center $k' \in \mathcal{D}_1$ (Property 3 in Lemma 2.10) such that $\sigma(k') = k$. As k' sends a demand of at most $\frac{D_{k'}}{2} \le \frac{U}{2}$ to a facility in F'_k, we can conclude that for each facility $i \in F_k, \sum_{j \in \mathcal{D}'} D_j \hat{x}_{ij} \le 2U\hat{y}_i$.

Consider a cluster $k \in \mathcal{D}_2$. If there is no $k' \in \mathcal{D}_1$ such that $\sigma(k') = k$, then for each facility $i \in F_k, \sum_{j \in \mathcal{D}'} D_j \hat{x}_{ij} \le 2U\hat{y}_i$ as only the copies created for k send demand to facilities in F_k, and the $\hat{x}$ assignment sends at most $(2U)\hat{y}_i$ demand to any facility in F_k. Otherwise, there exists only 1 center $k' \in \mathcal{D}_1$ such that $\sigma(k') = k$. As k' sends a demand of at most $\frac{D_{k'}}{2} \le \frac{U}{2}$ to a facility in F'_k, we can conclude that for each facility $i \in F_k$ (that is open to an extent of at least $\frac{1}{2}$), $\sum_{j \in \mathcal{D}'} D_j \hat{x}_{ij} \le 3U\hat{y}_i$.

Finally consider a cluster $k \in \mathcal{D}_3$. If there is no $k' \in \mathcal{D}_1$ such that $\sigma(k') = k$, then for each facility $i \in F_k, \sum_{j \in \mathcal{D}'} D_j \hat{x}_{ij} \le U\hat{y}_i$ as k is the only center assigned to these facilities and $D_k \le U$. Otherwise, there are at most 2 centers in $\mathcal{D}_3$ that are sending a demand of at most $\frac{U}{2}$ each to a facility in F'_k. Thus, we can conclude that for each facility $i \in F_k, \sum_{j \in \mathcal{D}'} D_j \hat{x}_{ij} \le 3U\hat{y}_i$.

This finishes the proof. ∎

2.5 Getting an integral solution

From Lemma 2.11 and Lemma 2.12, we know that $(\hat{x}, \hat{y})$ is a 3-violated solution to the clustered instance of cost at most $35\mathtt{OPT}$.

We are now working with the client set $\mathcal{D}_c$ which contains the copies of cluster centers in $\mathcal{D}_2(\mathcal{D}_{2,c})$ and cluster centers $\mathcal{D}_1$ and $\mathcal{D}_3$. We obtain an integral solution $(\tilde{x}, \tilde{y})$ for these cluster centers. This naturally translates to an integral solution $(x', \tilde{y})$ for the instance with cluster centers $\mathcal{D}'$, were x' is fractional by setting $x'_{ik} = \frac{\text{total demand assigned to copies of } k}{D_k}$.

It is easy to see from this definition that every facility i will be assigned the same amount of demand under both $(\tilde{x}, \tilde{y})$ and $(x', \tilde{y})$, and so, the cost of $(x', \tilde{y})$ is the same as the cost of $(\tilde{x}, \tilde{y})$.

We now describe the algorithm that takes as input a half-integral solution $(\hat{x}, \hat{y})$ and outputs an integral solution $(\tilde{x}, \tilde{y})$ which is done using a simple facility-location clustering step.

Algorithm 4: Half-integral to integral

1. Let $(\hat{x}, \hat{y})$ be a half-integral solution for the clients in $\mathcal{D}_c$.
   -Define $\hat{C}_k = \sum_{i \in \mathcal{F}} c_{ik} \hat{x}_{ik}$ for each $k \in \mathcal{D}_c$. (Assignment cost under $\hat{x}$)
   -Define $S_k = \{i \in \mathcal{F} : \hat{x}_{ik} > 0\}$ for each $k \in \mathcal{D}_c$.
2. Clustering. Let $\tilde{\mathcal{D}} = \emptyset$.
   -Let L denote list of clients in $\mathcal{D}_c$ ordered in increasing order of
    assignment costs under $\hat{x}$.
   -**while** $L \neq \emptyset$ **do**

   > Add first client $k \in L$ to $\tilde{\mathcal{D}}$.
   > **for** *each $j \in L$ (including k) such that $S_j \cap S_k \neq \emptyset$* **do**
   > > Set $ctr(j) = k$
   > > Remove j from L.
   >
   > **end**

   end
3. Obtain integral solution.
   (Note, when $S_j \not\subseteq S_{ctr(j)}$, $|S_j \setminus S_{ctr(j)}| = |S_j \cap S_{ctr(j)}| = 1$.)
   -Obtain integral opening vector $\tilde{y}$ by solving the following program,

   $$\min_{z \in \tilde{P}} f_i z_i + \sum_{k \in \mathcal{D}_c} H_k(z) \text{ ,where}$$

   $$\tilde{P} = \left\{ z(S_k) = 1 \quad \forall k \in \tilde{\mathcal{D}}, \quad z(S) \leq r(S) \quad \forall S \subseteq \mathcal{F} \right\}$$

   $$H_k(z) = \begin{cases} D_k \sum_{i \in ctr(k)} c_{ik} z_i & S_k \subseteq S_{ctr(k)} \\ D_k c_{ik} z_i + (1 - z_i) D_k (c_{i'k} + \sum_{i'' \in S_{ctr(k)}} c_{i'' ctr(k)}) & \text{otherwise, with} \\ & S_k \setminus S_{ctr(k)} = \{i\}, \\ & S_k \cap S_{ctr(k)} = \{i'\} \end{cases}$$

4. Integral assignments
   for *every client $j \in \mathcal{D}_c$* **do**

   > If $S_j \subseteq S_{ctr(j)}$, assign j to facility open in $S_{ctr(j)}$
   > Else, Let i be facility in $S_j \setminus S_{ctr(j)}$
   > > if $\tilde{y}_i = 1$, Assign j to i
   > > Else, assign j to facility open in $S_{ctr(j)}$

   end

Note that the polytope $\tilde{P}$ described in the algorithm above is integral. This is because it is the intersection of the matroid polytope for $\mathcal{M}$ with the matroid base polytope for the partition matroid defined by the sets S_j for $j \in \tilde{\mathcal{D}}$. Such a polytope is known to have integral extreme points. Thus, minimizing $H(z) = f_i z_i + \sum_{k \in \mathcal{D}_c} H_k(z)$ over $\tilde{P}$ yields an integral vector $\tilde{y}$

In Lemma 2.13, we show that the cost of $H(\tilde{y})$ is at most twice the cost of $(\hat{x}, \hat{y})$ and thus at most $70\texttt{OPT}$. In Lemma 2.14 we show that $(\tilde{x}, \tilde{y})$ violates capacities by a factor of at most 6 and that its cost is at most $H(\tilde{y})$.

Lemma 2.13. *The cost of $H(\tilde{y})$ is at most twice the cost of $(\hat{x}, \hat{y})$, that is, $H(\tilde{y}) \le 70\texttt{OPT}$.*

Proof. We prove this Lemma by exhibiting a z' such that $z' \in \tilde{P}$ with $H(z') \le 70\texttt{OPT}$. As $\tilde{y}$ was obtained by solving the linear program $\min_{z \in \tilde{P}} f_i z_i + \sum_{k \in \mathcal{D}_c} H_k(z)$, we know that $H(\tilde{y}) \le H(z')$. z' is defined as follows: $z'_i = x_{ij}$ if $i \in S_j$ for some $j \in \tilde{\mathcal{D}}$, else $z'_i = \hat{y}_i$. According to the definition of z', for each $j \in \tilde{\mathcal{D}}, \sum_{i \in \mathcal{F}} \hat{x}_{ij} = 1 = z(S_j)$. Moreover, $z'_i \le \hat{y}_i \le y_i$ for each $i \in \mathcal{F}$. Thus, $z' \in \tilde{P}$.

As $z'_i \le \hat{y}_i$ for each $i \in \mathcal{F}$, $\sum_{i \in \mathcal{F}} f_i z'_i \le \sum_{i \in \mathcal{F}} f_i \hat{y}_i$. As the cost of $(\hat{x}, \hat{y})$ is $\sum_{i \in \mathcal{F}} f_i \hat{y}_i + \sum_{k \in \mathcal{D}_c} D_k \hat{C}_k$, to complete the proof, we will show that for each center $k \in \mathcal{D}_c$, $H_k(z') \le 2D_k \hat{C}_k$. A property that is a consequence of our clustering procedure and that we will utilize repeatedly in this proof is that for $k \in \mathcal{D}_c, \hat{C}_k \ge \hat{C}_{ctr(k)}$.

For $k \in \mathcal{D}_c$ such that $S_k = S_{ctr(k)}$, $H_k(z) = D_k \sum_{i \in S_{ctr(k)}} c_{ik} z_i$. Thus, $H_k(z') = D_k \hat{C}_k \le 2D_k \hat{C}_k$. If for $k \in \mathcal{D}_c, S_k \subset S_{ctr(k)}$, let i be the facility in $S_k \cap S_{ctr(k)}$ and i' be the facility in $S_{ctr(k)} \setminus S_k$. Then,

$$\begin{aligned} H_k(z') &= \frac{1}{2} D_k(c_{ik} + c_{i'k}) \\ &\le \frac{1}{2} D_k(c_{ik} + c_{i'ctr(k)} + c_{ictr(k)} + c_{ik}) \\ &= D_k c_{ik} + D_k C_{ctr(k)} \\ &\le 2D_k c_{ik} = 2D_k C_k \end{aligned}$$

Otherwise, $H_k(z') = D_k c_{ik} z_i + (1 - z_i) D_k(c_{i'k} + \sum_{i'' \in S_{ctr(k)}} c_{i''ctr(k)})$, with $S_k \setminus S_{ctr(k)} = \{i\}, S_k \cap S_{ctr(k)} = \{i'\}$. Again, if $z'_i = 1$, and $H_k(z') = D_k c_{ik} \le 2D_k \hat{C}_k$. Else, $z'_i = \frac{1}{2}$ and $H_k(z')$ can be bounded as follows

$$H_k(z') = \frac{1}{2}D_k c_{ik} + \frac{1}{2}D_k c_{i'k} + \frac{1}{2}D_k(c_{i'ctr(k)} + c_{i''ctr(k)})$$
$$= D_k \hat{C}_k + D_k \hat{C}_{ctr(k)}$$
$$\leq 2D_k \hat{C}_k.$$

Hence, we have shown that $H_k(z') \leq 2D_k \hat{C}_k$ for every $k \in \mathcal{D}_c$. ∎

Lemma 2.14. *The solution $(\tilde{x}, \tilde{y})$ is a 6-violated solution and has cost at most $H(\tilde{y})$.*

Proof. For $k \in \mathcal{D}^c$ such that $S_k \subseteq S_{ctr(k)}$, the assignment cost under $\tilde{x}$ is $D_k c_{ik}$ where i is the facility open in $S_{ctr(k)}$. Note that this is exactly equal to $\sum_{i \in S_{ctr(k)}} c_{ik}\tilde{y}_i = H_k(\tilde{y})$.

For $k \notin \tilde{\mathcal{D}}$ with $S_k \nsubseteq S_{ctr(k)}$, let i be the facility in $S_k \setminus S_{ctr(k)}$ and i' be the facility in $S_k \cap S_{ctr(k)}$. If $|S_{ctr(k)}| = 1$, if $\tilde{y}_i = 1$, then k's assignment cost is c_{ik}, else it is equal to $c_{i'k}$, hence, k's assignment cost is $D_k(c_{ik}\tilde{y}_i + c_{i'k}\tilde{y}_{i'}) = H_k(\tilde{y})$.

If $|S_{ctr(k)}| = 2$, let i'' be the facility in $S_{ctr(k)} \setminus S_k$. if $\tilde{y}_i = 1$, then k's assignment cost is $D_k c_{ik} = H_k(\tilde{y})$. Else, k's assignment cost is $D_k c_{i'k}$ or $D_k c_{i''k}$ and we can see that $D_k c_{i'k}, D_k c_{i''k} \leq D_k(c_{i'k} + c_{i'ctr(k)} + c_{i''ctr(k)}) = H_k(\tilde{y})$.

Consider a cluster S_k for $k \in \mathcal{D}_c$. The total demand being handled by the facilities in the cluster is at most $3U$ (as the total facility weight $\hat{y}(S_k)$ is 1). According to our assignments, the total demand being sent to this cluster is at most doubled. All this demand will be assigned to only one facility as according to the integer opening vector $\tilde{y}$, only one facility is open inside S_k - hence, this facility is handling a demand of at most $6U$. The other case is when there is a cluster center $k \in \mathcal{D}_c \setminus \tilde{\mathcal{D}}$ and it's assigned completely to $i \in S_k \setminus S_{ctr(k)}$. As the demand being sent to this facility is also at most doubled, the demand its handling is at most $6U$. ∎

Thus, from Corollary 2.5, Lemma 2.14 and Lemma 2.15, we can conclude Theorem 2.1.

Chapter 3

Integrality gaps

In this chapter, we provide integrality gap results for the natural LP relaxation for capacitated matroid median. For the special case of capacitated k-median and even with no facility-opening costs, there are instances, where any integral solution has to open at least $2k$ facilities or violate the capacity constraint by a factor of at least 2 in order to have cost that can be bounded in terms of the LP optimum cost.

Our algorithm from the previous chapter shows that for CMM with uniform capacities, we can get an integral solution of cost at most `76OPT` if the integral solution is allowed violate the capacity constraint by a factor of at most 6. We show in Section 3.1 that a similar result based on rounding the natural LP is not possible for CMM with non-uniform capacities. In Section 3.2, we then provide an integrality gap example showing that even for capacitated matroid median with uniform capacities, if we are not allowed to violate capacities, then we must violate the matroid constraints by a non-constant factor in order to obtain a solution whose cost can be bounded relative to the natural LP optimum (we define what we mean by violating the matroid constraints precisely in section 3.2). This shows that a result analogous to capacitated k-median, wherein we respect capacities but violate k, is not possible (based on rounding the natural LP) for capacitated matroid median, even with uniform capacities.

3.1 Gap example for CMM with non-uniform capacities

The natural LP-relaxation for CMM with non-uniform capacities is the same as (P), but the right hand side of constraint (2) is now $U_i y_i$, where U_i is the capacity of facility i. We show that for every constant $\beta > 0$, there exists an instance of CMM-NC such that the LP optimum is 0, but every β-violated *integral* solution has strictly positive cost. Thus, we cannot obtain a result where we return a solution that violates capacities by an $O(1)$-factor and has cost that is bounded in terms of the LP-optimum. This proves Theorem 1.2 from Section 1.2.

Let $\beta > 0$ and let U be an integer greater than β. The instance has U groups, each consisting of a client with demand U and two facilities - one with capacity U, which we call a red facility and the other with capacity 1, which we call a blue facility. The clients and the facilities are co-located in each group, that is, the distance between them is 0. The distance between points in different groups is, say 1 (the exact value of the distance between groups is not important, but it is important that this distance between groups is strictly positive). The matroid defined over the set of facilities, $\mathcal{M} = (\mathcal{F}, \mathcal{I})$, is a partition matroid: $\mathcal{I} = \{F' \subseteq \mathcal{F} : |F' \cap F_U| \leq U - 1\}$ where F_U is the set of facilities of capacity U, that is, the red facilities.

The $LP_{\texttt{OPT}}$ for the above instance is 0 as we can construct a fractional solution (x, y) of cost 0. The solution is built as follows. Consider a group and let j be the client and let i_U and i_1 be the facilities of capacities U and 1 respectively in the group. Let $y_{i_1} = 1$, $y_{i_U} = 1 - \frac{1}{U}$ and let $x_{i_1 j} = \frac{1}{U}$, $x_{i_U j} = 1 - \frac{1}{U}$. It is easy to see that this solution is feasible for (P). In any β-violated integral solution, there must be a group in which i_U is not open due to the matroid $\mathcal{M}$. As capacity can only be violated up to an extent of β, there must be some demand that has to be assigned to a facility outside the group, which ensures strictly positive cost.
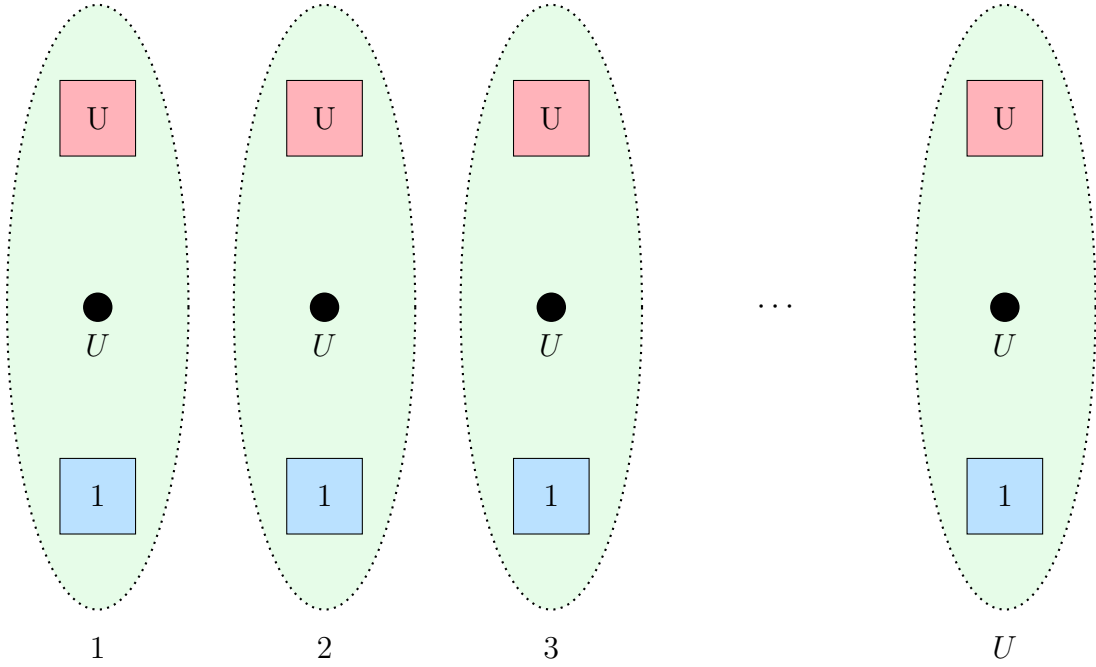
Figure 3.1: Integrality gap example for CMM with non-uniform capacities

3.2 Gap example for CMM with uniform capacities

Generalizing the definition of β-violated solution, we say that (x, y) is an (α, β)-violated solution to (P), if it violates the matroid rank constraints by an α factor i.e., $y(S) \leq \alpha \, r(S) \quad \forall S \subseteq \mathcal{F}$, and capacities by a β factor, and satisfies the remaining constraints of (P); that is, (x, y) satisfies: $\sum_{i \in \mathcal{F}} x_{ij} = 1$ for all $j \in \mathcal{D}$ and $0 \leq x_{ij} \leq y_i \leq 1$ for all $i \in \mathcal{F}, j \in \mathcal{D}$. Using results from matroid intersection, one can show that if (x, y) is an integral (α, β)-violated solution , then the set of open-facilities can be partitioned into at most α independent sets. Thus, this notion can be viewed as a generalization of the notion of violating the cardinality bound k in capacitated k-median.

We show that for every constant $\alpha > 0$, there exists an instance of CMM, even with uniform capacities, such that the LP-optimum is 0, but every $(\alpha, 1)$-violating integral solution has strictly positive cost. This shows that we cannot obtain a solution that respects capacities, and violates the matroid constraints by a constant-factor, and has cost that is bounded relative to the LP-optimum. This proves Theorem 1.3 in Section 1.2.

Let $\alpha > 0$ and let U be an integer greater than α. The instance has U groups. The facility set is partitioned into U blue facilities and U red facilities, each with capacity

1. Each group contains a client with demand $1 + \frac{1}{U}$ and, one red and one blue facility. The client and facilities are co-located in each group and the distance between points in different groups is 1 (again, the exact distance is not important, but this distance should be positive). The matroid over the set of facilities, $\mathcal{M} = (\mathcal{F}, \mathcal{I})$, is a partition matroid: $\mathcal{I} = \{F' \subseteq \mathcal{F} : |F' \cap F_r| \leq 1\}$ where F_r is the set of red facilities.
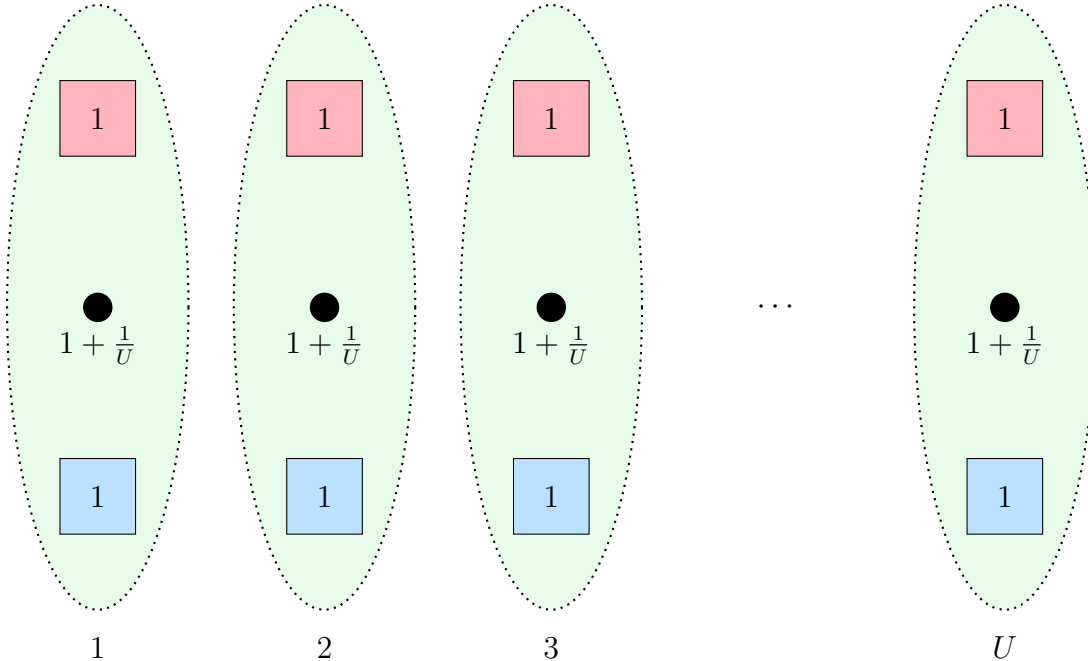

Figure 3.2: Integrality gap example for CMM with uniform capacities

The $LP_{\mathtt{OPT}}$ for the above instance is 0. A fractional solution (x, y) of cost 0 is described as follows. Consider a group and let j be the client in the group and let r and b denote the red and blue facilities in the group respectively. Set $y_b = 1$ and $y_r = \frac{1}{U}$. For j's assignment, set $x_{jr} = \frac{1}{U+1}$ and $x_{jb} = \frac{U}{U+1}$. It is easy to see that the solution satisfies the constraints of (P). In any $(\alpha, 1)$-violating integral solution, as the matroid constraints are allowed to be violated up to a factor of at most α, we can open at most α red facilities. So since, $U > \alpha$, there will be a group whose red facility is not open. Since we are not allowed to violate capacities, some of the demand in this group must be assigned to facilities outside of this group, which incurs a positive cost.

Chapter 4

Applications

It was shown in [Swa16] that matroid median is quite a versatile model which, in addition to generalizing k-median and facility location, is able to capture various (seemingly disparate) facility-location problems considered in literature. As remarked earlier, many of these facility location problems can be considered in the context of facility capacities, which often lead to more realistic models. We show in this chapter that our result for capacitated matroid median yields results for certain capacitated variants of the facility location problems captured by matroid median. Our reductions from the capacitated facility location problems to capacitated matroid median naturally mimic the reductions in [Swa16] from the (uncapacitated) problem to the (uncapacitated) matroid median problem.

4.1 The Data Placement problem

Problem definition. In the data placement problem, we are given a set of facilities $\mathcal{F}$, a set of clients $\mathcal{D}$ and a set of data objects $\mathcal{O}$. We have assignment costs $\{c_{ij}\}_{i \in \mathcal{F}, j \in \mathcal{D}}$ where the c_{ij}'s form a metric. Each client $j \in \mathcal{D}$ is associated with a demand d_j of data object $o(j) \in \mathcal{O}$ and each facility $i \in \mathcal{F}$ has an object capacity $o_i \in \mathbb{Z}_+$ which corresponds to the number of different types of objects the facility i can serve. A storage cost of f_o^i is incurred if facility i is being used to serve a client j of data object type o. We want to determine a set of data objects $\mathcal{O}(i) \subseteq \mathcal{O}$ to assign to each facility $i \in \mathcal{F}$ such that $|\mathcal{O}(i)| \leq o_i$ and assign each client to a facility $i(j) \in \mathcal{F}$ such that $o(j) \in \mathcal{O}(i(j))$. The objective is to minimize $\sum_{i \in \mathcal{F}} \sum_{o \in \mathcal{O}(i)} f_i^o + \sum_{j \in \mathcal{D}} d_j c_{ji(j)}$.

We can consider a capacitated variant, where for each facility $i \in \mathcal{F}$, and object $o \in \mathcal{O}$, we have have an associated capacity u_i^o which limits the total demand of clients demanding

object o that can be assigned to i: that is, a solution is feasible if additionally, we also have $\sum_{j:i(j)=i,o(j)=o} d_j \le u_i^o$ for every $i \in \mathcal{F}, o \in \mathcal{O}$.

Baev and Rajaraman gave the first constant factor approximation for the uncapacitated version of the problem [BR01], and this factor was improved to 10 [BRS08] and then to 8 [Swa16].

Reduction to capacitated matroid median. Given an instance of the capacitated data placement problem, we reduce it to an instance of capacitated matroid median as follows. The facility set is $\mathcal{F} \times \mathcal{O}$; each facility (i, o) has an opening cost of f_i^o and a capacity of u_i^o. The client set is $\mathcal{D}$. We set $c_{(i,o)j} = c_{ij}$ if $o(j) = o$ and ∞ otherwise. This ensures that each client j is assigned to a facility containing object $o(j)$. These new distances form a metric if the c_{ij}'s form a metric. The object capacity constraints are incorporated via the matroid constraint where a set $S \subseteq \mathcal{F} \times \mathcal{O}$ is independent if $|(i', o) \in S : i' = i| \le o_i$ for every $i \in \mathcal{F}$.

Thus, our result for CMM with uniform capacities yields a 76-approximation algorithm for capacitated data placement with uniform capacities, while violating capacities by a factor of 6. In Chapter 5, we discuss a different capacitated variant of the data placement problem.

4.2 The mobile facility location problem

Problem definition. The input is a metric space $(V, \{c_{ij}\})$. We have a set $\mathcal{D} \subseteq V$ of clients with each client j having a demand of d_j, and a set $\mathcal{F} \subseteq V$ of initial facility locations. A solution moves each facility $i \in \mathcal{F}$ to a final location $s_i \in V$ incurring a movement cost of c_{is_i} and assigns each client j to a final location s of some facility incurring an assignment cost of $d_j c_{sj}$. The goal is to minimize the sum of all the movement and assignment costs.

We can consider various versions of the capacitated generalization of mobile facility location. In one version, we can consider the case where each destination $s \in V$ is associated with a capacity U_s and at most U_s demand can be assigned to the facility that is moved to s. A more general version would be to associate a capacity of $U_{i,s}$ for every pair $i \in \mathcal{F}, s \in V$ such that a facility i can be assigned at most $U_{i,s}$ demand when moved to a location s. Note that this general version is flexible enough to capture the case when more than one facility is assigned to the same location s. If $F' \subseteq \mathcal{F}$ are the set of facilities assigned to a location $s \in V$, then the capacity at s is equal to $\sum_{i \in F'} U_{i,s}$. Our reduction to CMM below can capture this more general version of capacitated mobile facility location.

Friggstad and Salavatipour gave an 8-approximation for the uncapacitated version of the problem [FS11b]. Another algorithm with an approximation guarantee of 8 is through the reduction from UMM by Swamy [Swa16].

Reduction to capacitated matroid median. Given an instance of capacitated mobile facility location, we reduce it to an instance of capacitated matroid median as follows. The facility set is $\mathcal{F} \times V$ and the client set is $\mathcal{D}$. The facility (i, s_i) denotes that $i \in \mathcal{F}$ is moved to location $s_i \in V$ in the final solution and has opening cost c_{is_i} and a capacity of U_{i,s_i}. We set $c_{(i,s)j} = c_{sj}$ for every facility $(i, s) \in \mathcal{F} \times V$ and client j. These new distances form a metric: we have $c_{(i,s)j} \leq c_{(i,s)k} + c_{(i',s')k} + c_{(i',s')j}$ as $c_{sj} \leq c_{sk} + c_{s'k} + c_{s'j}$. The constraint that a facility can only be moved to one final location is encoded in the matroid constraint as follows: A set $S \subseteq \mathcal{F} \times V$ is independent if $|\{(i', s) \in S : i' = i\}| \leq 1$ for all i in $\mathcal{F}$.

Thus, our result for CMM with uniform capacities yields a 76-approximation algorithm for capacitated mobile facility location with uniform capacities, while violating capacities by a factor of 6.

4.3   The metric uniform MLCFL

Problem definition. We are given a set $\mathcal{F}$ of facilities and a set $\mathcal{D}$ of clients with assignment costs $\{c_{ij}\}_{j \in \mathcal{D}, i \in \mathcal{F}}$, where the c_{ij}'s form a metric. Also, we have a monotone latency-cost function $\lambda : \mathbb{Z}_+ \to \mathbb{R}_+$. The goal is to choose a set $F \subseteq \mathcal{F}$ of facilities to open, assign each open facility $i \in \mathcal{F}$ a distinct time-index $t_i \in \{1, \ldots, |\mathcal{F}|\}$, and assign each client j to an open facility $i(j) \in F$ so as to minimize $\sum_{i \in F} f_i + \sum_{j \in \mathcal{D}} (c_{i(j)j} + \lambda(t_{i(j)}))$.

In the capacitated version, each facility $i \in \mathcal{F}$ is associated with a capacity u_i, which limits the number of clients that may be assigned to it.

The problem was introduced by Chakrabarty and Swamy who gave a 10.773-approximation algorithm for the problem [CS11] which was improved to an 8-approximation in [Swa16].

Reduction to capacitated matroid median. The facility set is defined as $\mathcal{F} \times \{1, \ldots, |\mathcal{F}|\}$ and the matroid on this set encodes that a set S is independent if $|\{i \in \mathcal{F} : (i, t) \in S\}| \leq 1$ for all $t \in \{1, \ldots, |\mathcal{F}|\}$. We set $f_{(i,t)} = f_i, u_{(i,t)} = u_i$ and $c_{(i,t)j} = c_{ij} + \lambda(t)$; note that these distances form a metric.

Thus, our result for CMM with uniform capacities yields a 76-approximation algorithm for capacitated MLCFL with uniform capacities, while violating capacities by a factor of 6.

Chapter 5

Conclusions and Future Work

In this thesis, we presented an approximation algorithm for the capacitated matroid median problem with uniform capacities (CMM-UC) which violates the capacity constraint by a factor of at most 6. Moreover, we were able to show that the natural LP has unbounded integrality gap for (i) CMM-UC when the solution is not allowed to violate capacities and only allowed to violate the matroid constraint, and for (ii) CMM-NC when the solution is not allowed to violate the matroid constraint and only allowed to violate the capacity constraint. Our work leads to various further interesting open questions, and we mention some of them below.

5.1 Results based on the natural LP

- Is it possible to get a constant factor approximation algorithm for CMM-NC when the solution is allowed to violate capacities and only allowed to violate the matroid constraint by a small factor $(1 + \epsilon)$? Note that our integrality gap for CMM-NC fails when a small violation in the matroid constraint is allowed.

- Similar to the question above, is it possible to obtain a constant factor approximation algorithm for CMM-UC when the solution is allowed to violate the matroid constraint and allowed to violate the capacity constraint by a small factor $(1 + \epsilon)$? Our gap example for this case also fails when a small violation of the capacity constraint is allowed.

- The constants that our algorithm achieves, both in the approximation factor and capacity violation, are pretty large. It will be interesting to find algorithms that

obtain smaller constants. In the same vein, we know that for k-median, the capacities have to be violated by a factor of at most 2, is this the same for CMM? Or can we find gap examples which show that the capacity violation has to be at least a factor of c for $2 < c \leq 6$? As a step in this direction, our algorithm loses an additional factor of 2 in the capacity violation in converting a half-integral solution to an integral one. It would be good to understand whether such a loss is necessary in order to round losing only an $O(1)$-factor in the cost.

5.2   Results based on newer LP's

Recent years have seen great progress on the capacitated k-median problem. This progress was spurred by [Li17] and [Li16] in which constant factor approximation algorithms for CKM-UC and CKM-NC were given in which the solution opens at most $(1 + \epsilon)k$ facilities. The configuration LP of [Li16] was further used to give a constant factor approximation algorithm for CKM-UC which opens at most k facilities and violates the capacity constraint by a factor of at most $(1 + \epsilon)$ [BRU16]. This result was further generalized by [DL16] to the non-uniform capacities case thus settling the k-median problem up to pseudo-approximations.

The breakthrough result of [ASS14] gave the first LP-based constant factor approximation algorithm for capacitated facility location. They introduce a stronger LP based on multi-commodity flows and show that this LP has a constant integrality gap for CFL.

Hence, this leads us to the natural question of whether the power of these newer LP's can be harnessed to get better results for CMM-UC and CMM-NC.

5.3   Applications

In section 4.1, we showed that our our approximation algorithm for capacitated matroid median yields a 76-approximation for capacitated data placement with uniform capacities with capacity violation at most 6. That is, our algorithm produces a solution, that for every open facility $i \in \mathcal{F}$, assigns at most o_i different data objects to i and assigns at most a demand $6U$ to i for each kind of data object (where $u_i^o = U$ for each $i \in \mathcal{F}, i \in \mathcal{O}$). Myerson, Mungala and Plotkin considered a different variant of capacitated data placement in [MMP01]. They considered the version in which every facility $i \in \mathcal{F}$ has a object-capacity o_i limiting the different types of data objects that can be assigned to

the facility i and a global client capacity u_i that limits the total demand that can be assigned to i across the different types of data objects assigned to it. Their algorithm provides a constant-factor guarantee, but with logarithmic violation in the object and client capacities. Thus, it is an interesting question to see if it is possible to obtain a constant-factor approximation algorithm for this variant with no object-capacity violation and a constant-capacity violation.

References

[AFS13]   Sara Ahmadian, Zachary Friggstad, and Chaitanya Swamy. Local-search based approximation algorithms for mobile facility location problems. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2013, New Orleans, Louisiana, USA, January 6-8, 2013*, pages 1607–1621, 2013.

[ALB+13]   Ankit Aggarwal, Anand Louis, Manisha Bansal, Naveen Garg, Neelima Gupta, Shubham Gupta, and Surabhi Jain. A 3-approximation algorithm for the facility location problem with uniform capacities. *Math. Program.*, 141(1-2):527–547, 2013.

[ASS14]   Hyung-Chan An, Mohit Singh, and Ola Svensson. Lp-based algorithms for capacitated facility location. In *Proceedings of the 2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, FOCS '14, pages 256–265, Washington, DC, USA, 2014. IEEE Computer Society.

[AvdBGL15] Karen Aardal, Pieter L. van den Berg, Dion Gijswijt, and Shanfei Li. Approximation algorithms for hard capacitated k-facility location problems. *European Journal of Operational Research*, 242(2):358–368, 2015.

[BFRS15]   Jaroslaw Byrka, Krzysztof Fleszar, Bartosz Rybicki, and Joachim Spoerhase. Bi-factor approximation algorithms for hard capacitated k-median problems. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 722–736, 2015.

[BGG12]   Manisha Bansal, Naveen Garg, and Neelima Gupta. A 5-approximation for capacitated facility location. In *Algorithms - ESA 2012 - 20th Annual European Symposium, Ljubljana, Slovenia, September 10-12, 2012. Proceedings*, pages 133–144, 2012.

[BR01]   Ivan D. Baev and Rajmohan Rajaraman. Approximation algorithms for data placement in arbitrary networks. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '01, pages 661–670, Philadelphia, PA, USA, 2001. Society for Industrial and Applied Mathematics.

[BRS08]   Ivan Baev, Rajmohan Rajaraman, and Chaitanya Swamy. Approximation algorithms for data placement problems. *SIAM J. Comput.*, 38(4):1411–1429, August 2008.

[BRU16]   Jaroslaw Byrka, Bartosz Rybicki, and Sumedha Uniyal. An approximation algorithm for uniform capacitated k-median problem with 1+\epsilon capacity violation. In *Integer Programming and Combinatorial Optimization - 18th International Conference, IPCO 2016, Liège, Belgium, June 1-3, 2016, Proceedings*, pages 262–274, 2016.

[CGTS99]   Moses Charikar, Sudipto Guha, Éva Tardos, and David B. Shmoys. A constant-factor approximation algorithm for the k-median problem (extended abstract). In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, May 1-4, 1999, Atlanta, Georgia, USA*, pages 1–10, 1999.

[CR05]   Julia Chuzhoy and Yuval Rabani. Approximating k-median with non-uniform capacities. In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2005, Vancouver, British Columbia, Canada, January 23-25, 2005*, pages 952–958, 2005.

[CS11]   Deeparnab Chakrabarty and Chaitanya Swamy. Facility location with client latencies: Linear programming based techniques for minimum latency problems. In *Integer Programming and Combinatoral Optimization - 15th International Conference, IPCO 2011, New York, NY, USA, June 15-17, 2011. Proceedings*, pages 92–103, 2011.

[DL16]   H. Gökalp Demirci and Shi Li. Constant approximation for capacitated k-median with (1+epsilon)-capacity violation. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, pages 73:1–73:14, 2016.

[FS11a]   Zachary Friggstad and Mohammad R. Salavatipour. Minimizing movement in mobile facility location problems. *ACM Trans. Algorithms*, 7(3):28:1–28:22, 2011.

[FS11b]   Zachary Friggstad and Mohammad R. Salavatipour. Minimizing movement in mobile facility location problems. *ACM Trans. Algorithms*, 7(3):28:1–28:22, July 2011.

[HKK10]   MohammadTaghi Hajiaghayi, Rohit Khandekar, and Guy Kortsarz. Budgeted red-blue median and its generalizations. In *Algorithms - ESA 2010, 18th Annual European Symposium, Liverpool, UK, September 6-8, 2010. Proceedings, Part I*, pages 314–325, 2010.

[JMM⁺03]   Kamal Jain, Mohammad Mahdian, Evangelos Markakis, Amin Saberi, and Vijay V. Vazirani. Greedy facility location algorithms analyzed using dual fitting with factor-revealing lp. *J. ACM*, 50(6):795–824, November 2003.

[JV01]   Kamal Jain and Vijay V. Vazirani. Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation. *J. ACM*, 48(2):274–296, March 2001.

[KKN⁺11]   Ravishankar Krishnaswamy, Amit Kumar, Viswanath Nagarajan, Yogish Sabharwal, and Barna Saha. The matroid median problem. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pages 1117–1130, 2011.

[KLS17]   Ravishankar Krishnaswamy, Shi Li, and Sai Sandeep. Constant approximation for k-median and k-means with outliers via iterative rounding. *CoRR*, abs/1711.01323, 2017.

[Li11]   Shi Li. A 1.488 approximation algorithm for the uncapacitated facility location problem. In *Automata, Languages and Programming - 38th International Colloquium, ICALP 2011, Zurich, Switzerland, July 4-8, 2011, Proceedings, Part II*, pages 77–88, 2011.

[Li16]   Shi Li. Approximating capacitated k-median with 1+\epsilon k open facilities. In *Proceedings of the Twenty-seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '16, pages 786–796, Philadelphia, PA, USA, 2016. Society for Industrial and Applied Mathematics.

[Li17]   Shi Li. On uniform capacitated k-median beyond the natural lp relaxation. *ACM Trans. Algorithms*, 13(2):22:1–22:18, January 2017.

[LV92]   Jyh-Han Lin and Jeffrey Scott Vitter. Approximation algorithms for geometric median problems. *Inf. Process. Lett.*, 44(5):245–249, December 1992.

[MMP01]   Adam Meyerson, Kamesh Munagala, and Serge Plotkin. Web caching using access statistics. In *Proceedings of the Twelfth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '01, pages 354–363, Philadelphia, PA, USA, 2001. Society for Industrial and Applied Mathematics.

[Shm00]   David B. Shmoys. Approximation algorithms for facility location problems. In *Approximation Algorithms for Combinatorial Optimization, Third International Workshop, APPROX 2000, Saarbrücken, Germany, September 5-8, 2000, Proceedings*, pages 27–33, 2000.

[STA97]   David B. Shmoys, Éva Tardos, and Karen Aardal. Approximation algorithms for facility location problems (extended abstract). In *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, El Paso, Texas, USA, May 4-6, 1997*, pages 265–274, 1997.

[Swa16]   Chaitanya Swamy. Improved approximation algorithms for matroid and knapsack median problems and applications. *ACM Trans. Algorithms*, 12(4):49:1–49:22, 2016.