# Approximation Algorithms for Capacitated Facility Location with Outliers

Thesis Submitted to University of Delhi

In Partial Fulfillment of the Requirements

For award of the degree of

**Doctor of Philosophy in Computer Science**

**by**

**Rajni**

Department of Computer Science

University of Delhi

Delhi-110007 India

**May, 2025**

# Approximation Algorithms for Capacitated Facility Location with Outliers

Thesis Submitted to University of Delhi

In Partial Fulfillment of the Requirements

For award of the degree of

**Doctor of Philosophy in Computer Science**

**by**


**Rajni**

Department of Computer Science

University of Delhi

Delhi-110007 India


**May, 2025**

*This thesis is dedicated to my dear parents.*

*Mrs. Nikky and Mr. Rajesh Dabas*

# Declaration

The thesis entitled "*Approximation Algorithms for Capacitated Facility Location with Outliers*", which is being submitted for the award of the degree of Doctor of Philosophy in Computer Science, is a record of original and bona fide research work carried out by me in Department of Computer Science, University of Delhi, Delhi, India.

The work presented in this thesis has not been submitted to any other university or institute for the award of any degree or diploma.

**Rajni**
Department of Computer Science,
University of Delhi,
Delhi, India.

# Certificate

This is to certify that the thesis entitled "*Approximation Algorithms for Capacitated Facility Location with Outliers*" being submitted by Rajni in the Department of Computer Science, University of Delhi, Delhi, for the award of degree of Doctor of Philosophy is a record of original research work carried out by her under the supervision of Prof. Neelima Gupta.

The thesis or any part thereof has not been submitted to any other University or Institute for the award of any degree or diploma.

**Supervisor**
Prof. Neelima Gupta
Department of Computer Science
University of Delhi
Delhi, India.

**Head**
Prof. Neelima Gupta
Department of Computer Science
University of Delhi
Delhi, India.

Department of Computer Science
University of Delhi
Delhi-110007, India
Date:

# Certificate of Originality

The research work embodied in this thesis entitled "*Approximation Algorithms for Capacitated Facility Location with Outliers*" has been carried out by me at the Department of Computer Science, University of Delhi, Delhi, India. The manuscript has been subjected to a plagiarism check using the DrillBit. The work submitted for consideration of award of Ph.D. is original.

**Rajni**

**Student Approval Form**

| | |
|---|---|
| Name of Author | Rajni |
| Department | Computer Science |
| Degree | Ph.D. |
| University | University of Delhi |
| Guide | Prof. Neelima Gupta |
| Thesis Title | Approximation Algorithms for Capacitated Facility Location with Outliers |
| Year of Submission | 2025 |

**Agreement**

1. I hereby certify that, if appropriate, I have obtained and attached hereto a written permission/statement from the owner(s) of each third party copyrighted matter to be included in my thesis/dissertation, allowing distribution as specified below.

2. I hereby grant to the university and its agents the non-exclusive license to archive and make accessible, under the conditions specified below, my thesis/dissertation, in whole or in part in all forms of media, now or hereafter known. I retain all other ownership rights to the copyright of the thesis/dissertation. I also retain the right to use in future works (such as articles or books) all or part of this thesis, dissertation, or project report.

Conditions:

| | |
|---|---|
| 1. Release the entire work for access Worldwide | |
| 2. Release the entire work for 'My University' only for<br>1 Year<br>2 Year<br>3 Year<br>and after this time release the work for access worldwide. | |
| 3. Release the entire work for 'My University' only while at the same time releasing the following parts of the work (e.g. because other parts relate to publications) for worldwide access.<br>a) Bibliographic details and Synopsis only.<br>b) Bibliographic details, synopsis and the following chapters only.<br>c) Preview/Table of Contents/24 page only. | |
| 4. View Only (No Downloads) (worldwide) | |

Signature of the Scholar

Signature and seal of the Guide

Place: New Delhi

Date:

# Acknowledgment

First and foremost, I am deeply grateful to the Almighty for granting me the strength, wisdom, and perseverance to complete this work. It is through his grace that I have been able to bring this research to fruition.

I want to extend my heartfelt gratitude to everyone who has supported and guided me throughout my doctoral journey. I owe a special debt of thanks to my supervisor, Prof. Neelima Gupta, who believed in my potential as a researcher when I had doubts about myself and played a pivotal role in shaping my academic journey. Her faith in my abilities was a constant source of motivation. Prof. Gupta went above and beyond by bringing valuable collaborations and opening doors to research environments at other prestigious institutions. She actively introduced me to leading researchers in my field, providing unparalleled exposure and opportunities to broaden my perspective. Her mentorship has been transformative, guiding me through every phase of this journey and equipping me with the tools to thrive in the research world. I am forever thankful for her unwavering support, vision, and dedication.

I also sincerely thank my co-authors, Prof. Naveen Garg and Dr. Tanmay Inamdar, for their significant contributions and constant support. I also thank Prof. Naveen Garg (IIT Delhi) and Prof. Naveen Kumar (Department of Computer Science, University of Delhi), who generously accepted my invitation to be part of my advisory committee and provided invaluable insights throughout the research process. I am genuinely grateful to Prof. Vasudha Bhatnagar and Prof. Naveen Kumar for fostering a positive and conducive academic environment within the Department of Computer Science. I extend my gratitude to all faculty members and the office staff from the Department of Computer Science and P.G.D.A.V. College for their consistent moral support and practical assistance whenever required. Special thanks to Prof. Krishna Sharma, Principal of P.G.D.A.V. College, for her administrative and moral backing during my Ph.D. tenure.

I am especially thankful to Prof. Arindam Khan and Prof. Saket Saurabh for providing me with the opportunity to engage in the Anvenshan Setu fellowship, which broadened my perspective.

On a deeply personal note, I am eternally grateful to my family, whose unwavering love and support have been the foundation of my strength throughout this journey. I am thankful to my late grandparents, whose blessings continue to guide me. My parents, in particular, have been the driving force behind my pursuit of this doctoral degree. It was their dream long before it became mine, and they instilled that dream in me with relentless belief and encouragement. They nurtured it with all their love, care, and sacrifices. My brother, Chirag Dabas, has been a constant source of joy and encouragement. The small, everyday gestures often go unnoticed; his steady presence has been a silent pillar of strength that I rarely acknowledge but sincerely appreciate. I also carry the memory of my late sister, Pooja, whose spirit of cheer and unshakable optimism lifted me during challenging moments. Finally, I cannot thank my cherished friend, Rahul Dewan, enough for his unwavering support and for always lifting my spirits during my lowest points. His friendship has been a beacon of light, and his steadfast belief in me, even when I struggled to believe in myself, gave me the strength to keep moving forward through the difficult moments of this journey. I am also grateful to all my friends and extended family members for their continuous blessings and support, which have surrounded me with love and warmth throughout this journey.

**Rajni**

# Abstract

Facility location and its many variants are well-studied NP-hard problems in the operations research and theoretical computer science communities. In the *Classical Facility Location* (FL) problem, we are given a set of clients and a set of potential facilities. Every facility has an opening cost, referred to as the facility cost. For every client and facility, the cost of serving the client by the facility is given by the distance between them, which is referred to as the service cost. The objective is to determine a subset of facilities to open and assign clients to the selected facilities in such a way that the total cost, which includes both the facility opening costs and the service costs, is minimized. We assume that the distances form a metric, i.e., the distances are symmetric and satisfy the triangle inequality. When the underlying distances do not form a metric, the facility location problem is at least as hard to approximate as the set cover. Hence, in this thesis, we will rely on the crucial assumption that the distances, and consequently the service costs, form a valid metric.

Facility location problems often arise in practical settings where additional constraints naturally arise. For example, in many real-world applications, a facility may have a limited capacity, restricting the number of clients it can serve. This gives rise to the *Capacitated Facility Location* (CFL) problem, where each facility has a maximum capacity, and the goal is to minimize the total cost of opening the facilities and serving clients while respecting these capacity constraints.

Consider scenarios where some clients are far away from majority of the clients, these clients might disproportionately affect the cost of the overall solution, leading to solutions that are not robust. To handle such cases, Charikar et al. (2001) introduced a variant of facility location problem where a certain number of clients can be left unserved. These clients are popularly called as the *outliers*, and the respective facility location problem is called as the *Facility Location with Outliers* (FLO) problem. This is particularly useful when distant clients could significantly increase the overall cost or when the goal is to focus on serving a more representative subset of clients.

Though capacity and outlier constraints have been studied individually, it is quite nat-

ural to seek solutions that satisfy both constraints simultaneously, especially in complex real-world scenarios. For instance, in a logistics network, there may be a fixed number of distribution centers (facilities), each with limited capacity, but some remote customers may need to be excluded from the solution to keep costs manageable. Thus, a unified model that captures both capacity limitations and the ability to exclude outliers is highly relevant. This motivates us to study a generalization of CFL and FLO in this thesis, the *Capacitated Facility Location with Outliers* (CFLO) problem, which handles both capacity and outliers constraints simultaneously.

We first study the case when the facility opening costs are uniform, meaning all facilities have the same opening cost. We present a $(6.373 + \epsilon)$-approximation algorithm using a 2-operation local search approach, where $\epsilon > 0$ is a fixed constant. To the best of our knowledge, this is the first approximation algorithm for this problem. Furthermore, a simplified version of our local search algorithm and analysis leads to a $(3.733 + \epsilon)$-approximation algorithm for the *Capacitated Facility Location with Uniform Facility Cost* problem, improving the current best-known factor of $4$ by Kao (2023a) (which was achieved in a parallel work).

Next, we relax the assumption of uniform facility costs. We conjecture that the locality gap for Facility Location with Outliers in case non-uniform facility costs are unbounded, even in the uncapacitated setting and with constant factor violation in outliers. To support this conjecture, we provide an example where escaping the unbounded locality gap involves solving another instance of facility location with outliers problem itself. The unbounded locality gap example illustrates that obtaining a constant-factor approximation for CFLO, even with outlier and capacity violations, is difficult using the local search technique. Therefore, we turn our attention to LP-based algorithms. Both CFL (even with uniform capacities) and FLO are known to exhibit unbounded integrality gaps with respect to standard LP formulations. As a result, any algorithm that relies on the LP optimal solution as a lower bound will inevitably incur violations in both the capacity and outlier constraints. To make some progress on the problem, we focus on uniform capacities and introduce a *tri-criteria approximation*, where the solution approximates the cost within a constant factor while allowing small violations in both the capacity and outlier constraints. Specifically, we provide a $O(1/\epsilon^2)$ factor approximation for the problem, with violations in both capacities and outliers by a factor of $(1+\epsilon)$, for a fixed constant $\epsilon > 0$. This tri-criteria approximation could be useful in the future for eliminating violations in capacities, outliers, or both.

We then study the popular $k$-Median ($k$M) problem, which is similar to the Facility Location problem but instead of facility opening costs, it imposes a hard constraint on the number of facilities that can be opened. Both the $k$-Median and the $k$-Median with Outliers problems have been studied in the literature of approximation algorithms to obtain constant factor approximations. However, obtaining a constant-factor approximation for the Capacitated $k$-Median (C$k$M) problem remains one of the major open questions in the field. This challenge has led recent research to explore alternatives to traditional polynomial-time approximations. One promising direction is the study of fixed-parameter tractable (FPT) approximations[1]. Not only have researchers managed to obtain constant-factor approximations (without violations) for the C$k$M problem, but they have also improved approximations for the $k$-Median and $k$-Median with Outliers problems by allowing the running time to be FPT in parameters like $k$ and the number of outliers. Building on these advances in FPT approximations, we study FPT approximation for the Capacitated $k$-Median with Outliers (C$k$MO) problem. Specifically, we present an approximation-preserving reduction from C$k$MO to C$k$M, which runs in FPT time with respect to $k$, the number of outliers, and $\epsilon$, where $\epsilon > 0$ is a small constant. As a corollary, by using the best-known approximation for the C$k$M problem, we obtain a $(3+\epsilon)$-approximation algorithm for C$k$MO, which runs in FPT time with respect to $k$, the number of outliers, and $\epsilon$.

This thesis thus contributes to the ongoing effort to understand and improve the approximation algorithms for capacitated facility location problems, particularly when outliers are allowed.

[1]An algorithm is considered Fixed-Parameter Tractable (FPT) in a parameter $p$ if its running time is of the form $f(p) \cdot |I|^{O(1)}$, where $|I|$ is the input size and $f$ is an arbitrary function of $p$.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

In operations research, a key challenge lies in optimizing the costs associated with setting up infrastructure and serving a potential set of customers. Organisations strive to find solutions to these optimization problems, aiming to minimize costs and enhance the overall effectiveness of their operations. Examples of such operations include the establishment of cell phone towers, internet hubs, polling stations, COVID-19 vaccination centers, warehouses, and ATMs, among others. We build towards the definition of the famous *Facility Location* problem by discussing one of these examples.

Suppose a telecommunication company wants to expand its internet services to a new city. The company has identified potential locations across the city to set up internet distribution hubs, each with an associated set-up cost. The company has also collected data on potential consumers of the service in each locality. One of the major objectives of the company in this scenario is to set up hubs in a manner that minimizes their total expenditure, which includes both the cost of installing the hubs and serving their consumers via fibre connection.

This problem is a typical example of the facility location problem, which have been extensively studied in the literature of computer science and operations research since the early 1960s. Various facility location problems can be modelled by the following four key components:

- A potential set of *facilities* to open where every facility location might have an associated facility opening cost (*facility cost*), like the cost of installing hubs in the above example.

- A set of *clients* that need to be served by one of the opened facilities. Serving a client from a facility incurs a cost (*service cost*), such as the expense of providing a fibre connection from the hub to the customer's location in the above example.

- A set of conditions that must be fulfilled by the opened facilities or by allocation of clients to these facilities. One of the examples can be an upper limit on the maximum number of connections that can be provided from any hub.

- A cost function that calculates the expense associated with a solution, which comprises a subset of facilities and the assignment of clients to these facilities.

The objective of facility location problems is to find a solution that fulfils the specified conditions and minimizes the cost function. One of the simplest variants that can be modelled with the above components is the classical facility location problem, where the objective is to find an optimal set of facility locations and assign *all* clients to these facilities so as to minimize the total cost, without any specific conditions imposed. The problem is referred to as *Uncapacitated Facility Location Problem* (FL) in the literature.

In the more realistic setup, there are natural conditions that are imposed on facilities or on the assignment of clients. For instance, in our example each hub installed has a maximum limit on the number of connections it can provide. FL with upper bounds on the number of clients that can be served from any facility is popularly called as *Capacitated Facility Location Problem* (CFL).

In another practical situation, certain distant clients can significantly impact the overall solution, either because the installation cost of a hub at these distant locations is prohibitively high or because connecting these clients to any other hub incurs excessive costs. In such cases, excluding these clients from being served can substantially reduce the company's expenditure. Such clients are called as *outliers*. However, leaving too many clients unserved can be detrimental to the company's reputation and service commitments. This scenario motivates another well-known generalisation of FL, known as the *Facility*

*Location with Outliers* (FLO) problem. In FLO, an upper limit is imposed on the number of clients that can remain unserved. The objective is to select a subset of facilities and identify a set of outliers such that the combined cost of opening these facilities and serving only the non-outlier clients from these selected facilities is minimized.

Sometimes, the company might impose a constraint on the maximum number of hubs that can be installed. This hard bound on the number of facilities is referred to as *cardinality constraint*, leading to the $k$*-Facility Location* ($k$FL) problem. If the problem involves only a cardinality constraint and the facility opening costs are all zero, it simplifies to the well-known $k$-Median ($k$M) problem.

Facility Location problems are *NP-Hard*, so we can not hope for a polynomial time algorithm that solves every instance of the problem, under the assumption *P* $\neq$ *NP*. Most of the literature on FL and other related problems, therefore, centers on designing *approximation algorithms*. An $\alpha$-approximation algorithm for an optimization problem is a polynomial-time algorithm that guarantees, for any instance of the problem, the cost of the solution it produces is at most $\alpha$ times the cost of the optimal solution. The constant $\alpha$ is referred to as the *approximation factor* of the algorithm. Approximation algorithms have been one of the most successful techniques to deal with NP-hard problems for many decades. However, for certain variants of the problem, achieving a polynomial-time approximation is extremely challenging and has been open for many years in the literature of approximation algorithms. As a result, recent research has shifted focus to exploring alternatives beyond polynomial-time approximations. One such direction is the study of *fixed-parameter tractable* (FPT) approximations, where an algorithm is considered FPT with respect to a parameter $p$ if its running time can be expressed as $f(p) \cdot |I|^{O(1)}$, with $|I|$ denoting the input size and $f(p)$ is an arbitrary function of $p$. FPT approximations have also been employed to achieve improvements over the best-known (polynomial time) approximation results for several problems.

Approximation algorithms for facility location problems have been extensively studied in the fields of Operations Research and Theoretical Computer Science, leading to the development of various techniques. Following are some of the common techniques.

- **LP-Rounding:** In the LP-rounding technique, the algorithm rounds an optimal solution of an underlying linear programming (LP) formulation of the problem to an integral solution. The *integrality gap* of an LP formulation is defined as the maximum ratio of the optimal solution value of the integer program to the optimal solution value of its relaxation. A significant advantage of using LP-rounding techniques is their versatility, as they often extend to other related problems with similar relaxations.

- **Primal-Dual:** The Primal-Dual approach implicitly relies on an LP formulation of the problem. It typically constructs a feasible dual solution and a feasible (integral) primal solution simultaneously, then bounds the cost of the constructed primal solution in terms of the cost of the constructed dual solution.

- **Local Search:** In the local search technique, the algorithm iteratively moves from one feasible solution to a neighbouring solution with a lower cost, terminating with a locally-optimal solution. The maximum ratio of the solution quality of a local optimum to global optimum is known as the *locality gap*. Due to their ease of understanding and implementation, local search methods are often the preferred choice for practitioners.

- **Reduction:** In the reduction technique, the problem to be solved is transformed into another (generally simpler) problem for which an approximation algorithm is already known. Some loss in cost may occur during this transformation. This known algorithm is then used as a black box to obtain a solution, which is subsequently modified to derive a solution for the original problem.

- **Combination:** Solutions (or approximate solutions) of two or more subproblems of the given problem are combined to obtain an approximate solution for the original problem.

- **Greedy:** The greedy technique is an iterative method for constructing a solution. At each step, the algorithm makes the choice that seems best at that moment, extending the partially constructed solution incrementally.

If the facilities and clients are not located in a metric space, the facility location problem and its variants are at least as hard to approximate as set-cover, that is, $(\log n)$ hard to approximate where $n$ is input size, as proved by Hochbaum (1982). Hence, like most of the literature on FL, in this thesis we will be assuming and crucially using that the distances (and hence the service costs) form a metric.

Before discussing the problems studied in this thesis, we will briefly review the related work on the aforementioned facility location problems (detailed related work appears in Chapter 2).

FL and $k$M are relatively well-studied and well-understood in metric spaces. For FL, the best-known approximation ratio of $1.488$, due to Li (2011), is nearly tight, as it is hard to approximate FL within a factor of $1.463$, as proved by Guha and Khuller (1999). For $k$M, a series of advancements in approximation algorithms has culminated in a $2.613$ approximation in polynomial time by Gowda et al. (2023). On the negative side, Jain et al. (1998; 2002) proved that $k$M is hard to approximate within a factor of $1.736$.

Introducing capacity constraints significantly increases the complexity of these problems. For instance, the standard linear program (LP) for CFL is known to have an unbounded integrality gap even when the capacities are uniform[1]. Local search techniques have shown particular success with CFL, achieving the best-known ratio of $3$ by Aggarwal et al. (2010) and $5$ by Bansal et al. (2012) for uniform and non-uniform capacities. In cases where facility opening costs are uniform[2], the current best approximation ratio is $4$, as achieved by Kao (2023a) [3] in case of general capacities. Capacity constraints become notoriously hard in the presence of cardinality constraint– indeed, obtaining a constant-factor approximation for *Capacitated $k$-Median* (C$k$M) problem has been a long-standing open question in the area of approximation algorithms. There is partial progress toward this goal by designing so-called *bi-criteria* approximations that approximate the cost up to a constant factor, while also violating either the cardinality constraints or the capacity constraints by a small factor (1999; 2016; 2005; 2016; 2014; 2015a; 2018a). Recently, attempts have been made to overcome the polynomial-time

[1]Capacities are said to be uniform if they are the same for all facilities.

[2]Facility opening costs are uniform if they are the same for all facilities.

[3]The result is obtained in parallel to our result discussed later.

approximability of C$k$M by the introduction of FPT approximation algorithms. Adamczyk et al. (2019) designed a $(7+\epsilon)$-factor approximation for C$k$M that runs in time FPT in parameter $k$ and some constant $\epsilon$ which was later improved by Cohen-Addad and Li (2019) to a $(3+\epsilon)$-approximation.

Outliers were introduced in FL and $k$M by Charikar et al. (2001). They gave a $3$ factor approximation for FLO as well as a bi-criteria for *$k$-Median with Outliers* ($k$MO) that approximates the cost up to $4(1+\frac{1}{\epsilon})$ factor while violating outliers by a factor of $(1+\epsilon)$. The $3$ factor for FLO was later improved to $2$ by Jain et al. (2003). The first constant (unspecified large) approximation for $k$MO was obtained by Chen (2008), and the current best is $(6.994+\epsilon)$-approximation by Gupta et al. (2021). Agrawal et al. (2023) designed an approximation preserving reduction from $k$MO to $k$-Median that is FPT in $k$, number of outliers and $\epsilon$. As a corollary, they obtained a $(1+2/e+\epsilon)$-approximation for $k$MO that is FPT in $k$, number of outliers and $\epsilon$ where $\epsilon > 0$ is a small constant.

Though capacity and outlier constraints have been studied individually as well as with cardinality constraint in the literature, it is quite natural to seek a solution that satisfies both capacity and outlier constraints simultaneously. To the best of our knowledge, the only work that simultaneously handles both of these constraints is in the context of $k$-Center[4]. Cygan and Kociumaka (2014) gave $25$ and $23$ factor approximations for uniform and non-uniform capacities case respectively. On the other hand, Goyal and Jaiswal (2023) designed a 2-factor tight FPT approximation for the problem parameterized by $k$ and the number of outliers. This motivates us to study FL and $k$M in the presence of both outlier and capacity constraints. In parallel, Jaiswal and Kumar (2023) also studied $k$M in the presence of both capacity and outlier constraints and obtained a $(3+\epsilon)$ FPT approximation, parameterized by $k$, the number of outliers, and $\epsilon$. We present the same result in this thesis using a different technique.

We will next discuss the two problems studied in this thesis in detail.

[4]$k$-Center is same as $k$-Median except in $k$-Center the objective is to minimize the maximum service cost instead of total service cost.

## 1.1 Problems Studied in the Thesis

In this section, we introduce the two problems studied in this thesis: *Capacitated Facility Location with Outliers* (CFLO) and *Capacitated $k$-Median with Outliers* (C$k$MO). These problems provide more realistic models for real-world scenarios. Each subsection begins with defining the problem, followed by highlights of our contributions to the respective problem. A summary of our results is provided in Table 1.1.

### 1.1.1 Capacitated Facility Location with Outliers (CFLO)

In CFLO, we are given a set of clients and (potential) facilities in a metric space. Every facility has an opening cost (*facility cost*) and a capacity specifying the maximum number of clients it can serve. For each client-facility pair, we are provided with the cost of serving the client by the facility (referred to as the *service cost*). We are also given a hard bound on the number of outliers allowed. The objective is to open a subset of facilities and select a subset of clients to serve so that the cost of opening these facilities and serving the selected clients from the open facilities is minimized while respecting the capacity and outlier bounds.

In Chapter 3, we present our first result on CFLO with uniform facility opening costs. The facility opening costs are called uniform if they are the same for all the facilities. The results are obtained using the local search technique. We conjecture that the locality gap for the facility location problem with outliers in case non-uniform facility opening costs is unbounded, even in the uncapacitated case. The locality gap, of course, depends on the specific set of local search operations allowed. Friggstad et al. (2019) gave an example to show that any constant size multi-swap operation can not yield a local search algorithm with a bounded locality gap when the facility opening costs are general. The example provided in Friggstad et al. (2018) can be overcome by employing one of the non-constant swap operations introduced in our algorithm. However, in Section 3.7, we present a challenging example that highlights an issue even with the non-constant size swaps. In this case, escaping the unbounded locality gap in the example requires an operation that, in essence, involves solving an instance of the FLO itself. We are, in fact, able to modify our example to show that even after allowing constant factor violation in outliers, escaping the locality gap involves solving an instance of FLO.

For easy disposition of ideas, we first present a result for CFL problem and then extend it to CFLO. The result is a $(3.733 + \epsilon)$ approximation for CFL with uniform facility costs as stated in Theorem 1.1. The algorithm is a very simple 2-operation local search algorithm. Our result is interesting not only because it lends itself to an extension to CFLO but also because the analysis of the algorithm is quite simple and allows us to argue a sharper approximation for CFL with uniform facility costs.

**Theorem 1.1.** *There exists a polynomial time local search procedure with* $2$ *operations that yields a locally optimal solution which is a* $(3.733+\epsilon)$*-approximation to the optimum solution of the capacitated facility location problem with uniform facility opening costs.*

We next extend the ideas of CFL to obtain a $(6.373 + \epsilon)$- approximation for CFLO assuming uniform facility opening costs as stated in Theorem 1.2. The hard constraints of capacities and number of outliers make the CFLO problem very challenging, and to the best of our knowledge, no approximation is known for this problem.

**Theorem 1.2.** *There exists a polynomial time local search procedure with* $2$ *operations that yields a locally optimal solution which is a* $(6.373+\epsilon)$*-approximation to the optimum solution of the capacitated facility location problem with outliers and uniform facility opening costs.*

Next, in Chapter 4, we relax the assumption of uniform facility opening costs. Much of the existing literature on facility location problems with outliers employs the primal-dual technique or a combination of primal-dual/dual-fitting with greedy/local search schemes. However, the primal-dual method has not been very successful in handling capacities, making it unlikely that these approaches for outliers can be effectively extended to include capacity constraints. For instance, despite the unbounded integrality gap for FLO, Charikar et al. (2001) managed to circumvent the gap by estimating the maximum facility opening cost in an optimal solution and subsequently providing a primal-dual solution for the problem. Conversely, no primal-dual solution has been able to overcome the integrality gap for CFL. On the other hand, as stated above, the local search technique is unlikely to succeed when dealing with outliers and non-uniform facility opening costs, even with constant factor violations in outliers and capacities.

Furthermore, both the CFL (even with uniform capacities) and FLO problems are known to have unbounded integrality gaps with respect to standard linear programming relaxations, as shown by Shmoys et al. (1997) and Charikar et al. (2001), respectively. Thus, obtaining even a bi-criteria solution seems difficult by rounding the solution to the standard LP. We make some progress for the problem by obtaining a *tri-criteria* approximation algorithm when the capacities are uniform. A tri-criteria solution approximates the cost up to a constant factor while violating the outlier and the capacity constraints by a small factor. In particular, we present the result stated in Theorem 1.3. The tri-criteria solution could be beneficial in the future for eliminating violations in capacities, outliers, or both.

**Theorem 1.3.** *There is a polynomial time algorithm that approximates capacitated facility location problem with outliers and uniform capacities within a constant factor* $(O(1/\epsilon^2))$ *violating the capacities and outliers by a factor of at most* $(1+\epsilon)$*, for a given constant* $\epsilon > 0$.

Another natural question that arises is whether an FPT approximation solution can be obtained for CFLO without any violations. The obvious parameter is the number of outliers but it is unclear whether such a solution can be achieved FPT solely in the number of outliers. If the number of facilities opened in an optimal solution is known, it is possible to develop an approximation algorithm FPT in the number of outliers and the solution size. However, we do not know the optimal solution size. One standard approach to using solution size as an FPT parameter is to give a bound $k$ on the solution size as a parameter in the input. This reduces the problem to obtaining a solution to C$k$FLO (*Capacitated $k$-Facility Location with Outliers*), a common generalization of CFLO and C$k$MO. In Chapter 5 (Section 5.6), we give a $(3+\epsilon)$ FPT approximation for C$k$FLO that runs in time FPT in $k$, the number of outliers and $\epsilon$ where $\epsilon > 0$ is a small constant (Theorem 1.4).

**Theorem 1.4.** *There exists a $(3 + \epsilon)$ approximation for CkFLO that runs in time FPT in $k$, the number of outliers and $\epsilon$ where $\epsilon > 0$ is a small constant.*

### 1.1.2 Capacitated $k$-Median with Outliers (C$k$MO)

In C$k$MO, we are given a set of clients and (potential) facilities in a metric space. For each client-facility pair, we are provided with the cost of serving the client by the facility (*service cost*). Each facility has a capacity specifying the maximum number of clients it can serve. Additionally, there are hard bounds on the number of facilities that can be opened and the number of outliers allowed. Recall that the bound on the maximum number of facilities allowed to be open is referred to as the cardinality constraint. The objective is to open a subset of facilities and select a subset of clients to serve such that the total cost of serving the selected clients from the open facilities is minimized while respecting the cardinality, capacity, and outlier bounds.

Note that C$k$MO generalizes C$k$M, and the polynomial-time approximability of the latter itself remains open. Therefore, inspired by FPT approximations on C$k$M, we aim for FPT approximation for C$k$MO and present Theorem 1.5 in Chapter 5.

**Theorem 1.5.** *[Informal] There exists a randomized approximation-preserving reduction from CkMO to CkM that runs in time FPT in $k$, the number of outliers and $\epsilon$, where the underlying metric space remains unchanged and $\epsilon > 0$ is a small constant.*

By plugging in the best-known approximations for the C$k$M, we obtain Corollary 1.6.

**Corollary 1.6.** *There exists a randomized algorithm that runs in time FPT in $k$, the number of outliers and $\epsilon$ where $\epsilon > 0$ is a small constant and returns a $(3 + \epsilon)$ approximation with high probability.*

| Problem | Facility Opening Costs | Capacities | Factor | Violations | Previous Result |
|---|---|---|---|---|---|
| **Polynomial Time Approximations** | | | | | |
| **CFL (Chapter 3)** | U | NU | $(3.733 + \epsilon)$ | Nil | 4 Kao (2023a) |
| **CFLO (Chapter 3)** | U | NU | $(6.373 + \epsilon)$ | Nil | Nil |
| **CFLO (Chapter 4)** | NU | U | $O(1/\epsilon^2)$ | $(1 + \epsilon)u$ $(1 + \epsilon)L$ | Nil |
| **FPT Approximations (FPT in $k$, $L$ and $\epsilon$)** | | | | | |
| **C$k$MO (Chapter 5)** | NA | NU | $(3 + \epsilon)$ | Nil | Nil |
| **C$k$FLO (Chapter 5)** | NU | NU | $(3 + \epsilon)$ | Nil | Nil |

Table 1.1: Summary of our results. U and NU are used to denote uniform and non-uniform respectively. NA stands for 'Not Applicable '. $u$ denotes the uniform capacity, $L$ is the number of outliers and $\epsilon > 0$ is a small constant.

## 1.2 Notations and Preliminaries

In this section, we introduce notations that will be used consistently throughout the thesis. We will use $\mathbb{N}$, $\mathbb{R}$ and, $\mathbb{R}^+$ to denote the set of non-negative integers, the set of reals and the set of non-negative reals. We denote the metric space as $(\mathcal{P}, d)$, where $\mathcal{P}$ is a finite set of points and $d : \mathcal{P} \times \mathcal{P} \rightarrow \mathbb{R}^+$ is a distance function satisfying the triangle inequality and symmetry property. We will use *distance* and *service cost* interchangeably to denote distances in metric space. $F \subseteq \mathcal{P}$ and $X \subseteq \mathcal{P}$ denote the sets of $m$ facilities and $n$ clients, respectively. Typically, indices $i$ and $j$ are used for facilities and clients, respectively.

For a facility $i$, $f_i$ denotes the opening cost of that facility. If facility opening costs are uniform, we use $f$ as the notation. The capacity of facility $i$ is denoted by $u(i)$, while $u$ represents uniform capacities. $L$ represents the maximum number of outliers allowed, and $k$ denotes the number of facilities that can be opened under the cardinality constraint. Finally, $\epsilon > 0$ is used to denote a small constant which may vary for different algorithms described in the thesis.

**Definition 1.1** (Minimum Cost Flow Problem)**.** *Given a directed graph $G = (V, E)$ with a source vertex $s \in V$ and a sink vertex $t \in V$, a capacity function $u : E \rightarrow \mathbb{R}^+$, a cost function $d : E \rightarrow \mathbb{R}$, and a demand $D \in \mathbb{R}^+$, the goal is to find a flow $f : E \rightarrow \mathbb{R}^+$ that satisfies the following conditions:*

*i* ***Capacity constraints:*** *For all $(u, v) \in E$, $0 \leq f(u, v) \leq u(u, v)$.*

*ii* ***Flow conservation:*** *For all $v \in V \setminus \{s, t\}$,*

$$\sum_{(u,v) \in E} f(u, v) = \sum_{(v,w) \in E} f(v, w).$$

*iii* ***Flow value:*** *The total flow from $s$ to $t$ is exactly $D$, i.e.,*

$$\sum_{(s,v) \in E} f(s, v) = D.$$

*iv* ***Objective:*** *Minimize the total cost of the flow:*

$$\sum_{(u,v) \in E} d(u, v) \cdot f(u, v).$$

The minimum cost flow problem can be solved optimally in polynomial time (1993).

**Lemma 1.7.** *Once the set of open facilities (say $\mathcal{F} \subseteq F$) is fixed, the outliers and the assignment of clients to facilities in $\mathcal{F}$ can be determined in polynomial time using minimum cost flow problem.*

**Proof.** Once the set of open facilities (say $\mathcal{F}$) is fixed, the outliers and the assignment of clients to facilities are easily determined by solving a minimum cost flow problem as follows. Refer Figure 1.1. We set up a bipartite graph with vertex sets $X$ and $\mathcal{F} \cup \{o\}$;

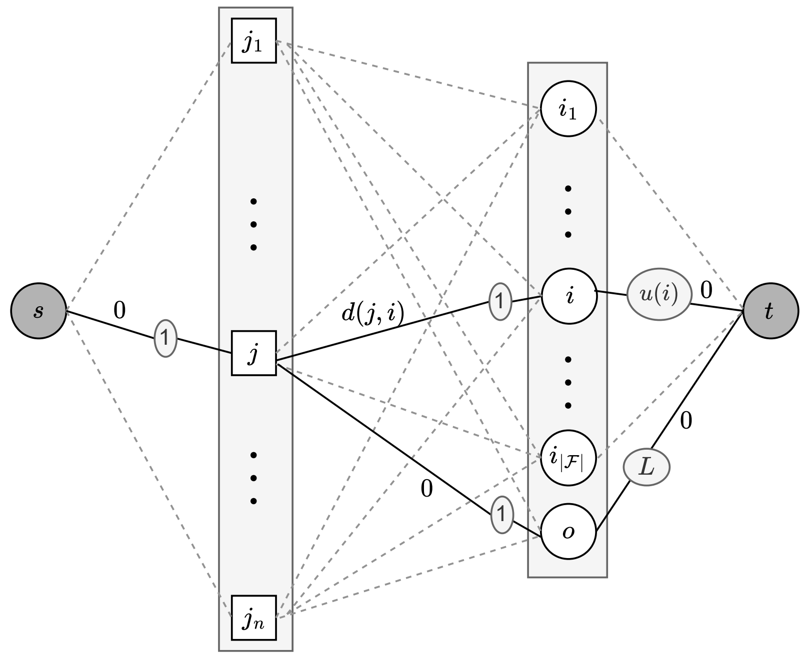



Figure 1.1: Minimum cost flow problem.

the vertex $o$ will allow us to identify outliers. An edge from every client $j \in X$ to every facility $i \in \mathcal{F}$ is added with cost $d(j,i)$ and capacity $1$ while an edge from the client every $j \in X$ to vertex $o$ is added cost $0$ and capacity $1$. There is a zero-cost edge from every vertex $i \in \mathcal{F}$ to the sink, $t$, of capacity $u(i)$ and a zero-cost edge from vertex $o$ to $t$ of capacity $L$. Similarly, there is an edge from the source, $s$, to all vertices in $X$ of cost $0$ and capacity $1$. A min-cost flow that routes $|X|$ demand from $s$ to $t$ gives the best assignment of clients to facilities. Note that, it is no loss of generality to assume that the number of outliers made are exactly $L$. In case the flow from $o$ to $t$ is less than $L$, we can reroute some additional clients from $o$ to $t$ without increasing the cost. Thus an optimal flow routes $L$ units through the edge $(o,t)$, thus identifying the $L$ outliers. $\square$

## 1.3 Organisation of the Thesis

We begin by presenting a detailed related work in Chapter 2. In Chapter 3, we present our results for the case of uniform opening costs. For ease of disposition of ideas, we first discuss the result for CFL with uniform opening costs (Theorem 1.1), followed by the result for CFLO with uniform opening costs (Theorem 1.2). We then relax the assumption of uniform facility opening cost and present our tri-criteria approximation for CFLO with uniform capacities (Theorem 1.3) in Chapter 4. In Chapter 5, we give the FPT approximation results; all the ideas are first presented for C$k$MO (Theorem 1.5 and Corollary 1.6), and then in Section 5.6, we discuss the modifications required to account for facility opening costs (Theorem 1.4). Finally, we conclude in Chapter 6, where we discuss open questions and potential directions for future research.

# Chapter 2

# Related Work

## 2.1 Classical Facility Location (FL)

The Facility Location (FL) problem is NP-hard and has gained significant attention in the literature. Hochbaum (1982) introduced a greedy algorithm with an $O(\log n)$ approximation guarantee, where $n$ denotes the input size. LP rounding and greedy algorithms have proven effective in providing constant-factor approximations for the classical facility location problem. The first constant-factor approximation algorithm for this problem was proposed by Shmoys et al. (1997), who used the LP rounding techniques of Lin and Vitter (1992) to achieve an approximation ratio of $3.16$.

Subsequent work has progressively improved the approximation factor. Guha and Khuller (1998; 1999) enhanced the approximation to $2.41$ by combining LP rounding with greedy augmentation which was further improved to $1.736$ by Chudak and Shmoys (1998; 2003) using the LP rounding technique. Jain et al. (2002; 2003) achieved an approximation ratio of $1.61$ using a greedy algorithm, while Sviridenko (2002) improved this result to $1.582$ through LP rounding. Mahdian et al. (2002) combined Jain et al.'s greedy algorithm (2002; 2003) with cost scaling to reach a $1.52$ approximation. Byrka (2007) modified the LP rounding algorithm of Chudak and Shmoys (1998; 2003) to obtain a $1.50$ approximation, which was later used by Li (2011; 2013) in conjunction with Jain et al.'s greedy algorithm (2002; 2003) to achieve the current best approximation ratio of $1.488$ for the FL problem.

Other techniques, such as the $(5+\epsilon)$ factor local search algorithm by Korupolu et al. (1998; 2000), have also been explored. The primal-dual method, first used by Jain and Vazirani (1999; 2001) to achieve a 3-factor approximation, was later enhanced by Charikar and Guha (1999) to achieve $1.853$ and $1.728$ approximation factors through the combination of greedy augmentation and LP rounding. A summary of the results is provided in Table 2.1.

Regarding hardness results, Guha and Khuller (1999) proved that it is impossible to obtain an approximation ratio better than $1.463$, assuming $NP \notin DTIME(n^{O(\log\log n)})$.

| **Approximation Factor** | **Reference** | **Technique** |
|---|---|---|
| $O(\log n)$ | Hochbaum (1982) | Greedy |
| $3.16$ | Shmoys et al. (1997) | LP rounding |
| $2.41$ | Guha and Khuller (1998; 1999) | LP rounding and greedy |
| $1.736$ | Chudak and Shmoys (1998; 2003) | LP rounding |
| $5+\epsilon$ | Korupolu et al. (1998; 2000) | Local Search |
| $3$ | Jain and Vazirani (1999; 2001) | Primal Dual |
| $1.853$ | Charikar and Guha (1999) | Primal Dual and greedy |
| $1.728$ | Charikar and Guha (1999) | LP rounding, primal-dual and greedy |
| $1.61$ | Jain et al. (2002; 2003) | Greedy |
| $1.582$ | Sviridenko (2002) | LP rounding |
| $1.52$ | Mahdian et al. (2002) | Greedy and cost scaling |
| $1.50$ | Byrka (2007) | LP rounding |
| $1.488$ | Li (2011; 2013) | LP rounding and greedy |

Table 2.1: State of the Art: Classical Facility Location

## 2.2 Capacitated Facility Location (CFL)

The capacitated facility location problem (CFL) was first addressed by Shmoys et al. (1997) for the case of uniform capacities. They used the LP rounding technique for rounding the solution of the standard LP. Since the standard LP for CFL is known to have an unbounded integrality gap, they provided a bi-criteria approximation, achieving a 7-factor approximation with a $7/2$ factor violation in capacities. Korupolu et al. (1998; 2000) introduced the first local search-based algorithm, focusing on uniform capacities and achieving an approximation factor of $(8+\epsilon)$. Since then, most results for this problem have been based on local search heuristics. Chudak and Williamson (1999) improved the factor to $(5.83+\epsilon)$ by using simpler analysis for the same heuristic. The best-known approximation for the uniform capacity case, $(3+\epsilon)$, was achieved by Aggarwal et al. (2010), who also used the local search procedure but with strengthened analysis. A summary of these results is provided in Table 2.2.

| **Approximation Factor** | **Violation in Capacities** | **Reference** | **Technique** |
|---|---|---|---|
| $7$ | $7/2$ | Shmoys et al. (1997) | LP Rounding |
| $8+\epsilon$ | Nil | Korupolu et al. (1998; 2000) | Local Search |
| $5.83+\epsilon$ | Nil | Chudak and Williamson (1999) | Local Search |
| $3+\epsilon$ | Nil | Aggarwal et al. (2010) | Local Search |

Table 2.2: State of the Art: Uniform Capacitated Facility Location

Local search for the general capacity setting was first explored by Pal et al. (2001), who proposed a $(9+\epsilon)$-factor approximation algorithm. This was followed by improvements to $(8+\epsilon)$ and $(5.83+\epsilon)$ by Mahdian and Pal (2003) and Zhang et al. (2005), respectively. The approximation ratio was then reduced to $(5+\epsilon)$ by Bansal et al. (2012), which remains the best-known approximation for the problem. Unlike the extensive LP-based techniques developed for FL, it was surprising that no LP-based algorithm with a constant approximation guarantee for the CFL had been found for a long time. In fact, the development of an LP-based approximation algorithm with an $O(1)$ guarantee for

CFL was considered one of the ten open problems in the textbook by Williamson and Shmoys (2011). This challenge was ultimately addressed in the influential work of An et al. (2014), who introduced a novel multi-commodity flow network (MFN) relaxation to obtain a $288$ factor approximation algorithm. In a subsequent paper, Kao (2023b) presented an iterative rounding approach and demonstrated that the integrality gap of the MFN relaxation is at most $9.0297$. A summary of these results can be found in Table 2.3.

| **Approximation Factor** | **Reference** | **Technique** |
|---|---|---|
| $9 + \epsilon$ | Pal et al. (2001) | Local Search |
| $8 + \epsilon$ | Mahdian and Pal (2003) | Local Search |
| $5.83 + \epsilon$ | Zhang et al. (2005) | Local Search |
| $5 + \epsilon$ | Bansal et al. (2012) | Local Search |
| $288$ | An et al. (2014) | Strengthened LP |
| $9.0927$ | Kao (2023b) | Strengthened LP |

Table 2.3: State of the Art: Capacitated Facility Location

In the pursuit of determining the approximability of the CFL problem, an important variation, where the *facility cost is uniform*, was studied by Levi et al. (2012). They gave a $5$ factor approximation for this special case via the LP-rounding technique. On the other hand, Aardal et al. (2015) presented a $4.562$- approximation based on the local search technique. Recently, Kao (2023a) presented an LP-based $4$-approximation algorithm for the problem. In a parallel work, we gave a $(3.733 + \epsilon)$ factor approximation using the local search technique, which is also the current best ratio. Refer Table 2.4.

| **Approximation Factor** | **Reference** | **Technique** |
|---|---|---|
| $5$ | Levi et al. (2012) | LP-Rounding |
| $4.562$ | Aardal et al. (2015) | Reduction + Combinatorial |
| $4$ | Kao (2023a) | LP-Rounding |
| $3.733 + \epsilon$ | Dabas et al. (2024) | Local Search |

Table 2.4: State of the Art: Capacitated Facility Location with Uniform Facility Costs

## 2.3 Facility Location with Outliers (FLO)

The concept of outliers in facility location was first introduced by Charikar et al. (2001). The standard linear programming (LP) formulation for the Facility Location with Outliers (FLO) has an unbounded integrality gap, making it impossible to achieve a bounded approximation guarantee using the standard LP as a lower bound for the optimal solution. To address this challenge, Charikar et al. (2001) guessed the cost of the most expensive facility in an optimal solution. Building on the primal-dual framework of Jain and Vazirani (1999), they proposed a $3$- factor approximation algorithm. The algorithm is applied to a modified instance where the guessed facility has a facility-opening cost of $0$, while the opening costs of facilities more expensive than the guessed facility are set to infinity. The algorithm is run only until $(n - L)$ clients are connected, rather than connecting all clients as done in Jain and Vazirani's primal-dual algorithm (2001). The costs associated with the most expensive facility and the last facility opened in the solution are then separately bounded.

The approximation factor was subsequently improved to $2$ by Jain et al. (2003) using a simple greedy algorithm. This algorithm iteratively selects the most cost-effective facility at each step, where cost-effectiveness is defined as the ratio of the incurred cost to the number of new clients covered. The analysis of the algorithm employs the dual-fitting technique. To leverage LP-duality in analyzing this approach, the authors provide an alternative description of the greedy algorithm, which can be viewed as a modification of Jain and Vazirani's original method. Specifically, when a client is connected to an open facility, it withdraws its contribution towards the opening cost of other facilities. This withdrawal step is crucial as it ensures that the primal solution is fully covered by the dual. A further distinction is that a client may change its connection to a different, closer facility, and in such cases, it contributes to the difference in cost towards the opening of the new facility.

Refer to Table 2.5 for related work on FLO.

| Approximation Factor | Reference | Technique |
|---|---|---|
| 3 | Charikar et al. (2001) | Primal-Dual |
| 2 | Jain et al. (2003) | Greedy algorithm analyzed using dual-fitting |

Table 2.5: State of the Art: Facility Location with Outliers

## 2.4 Classical $k$-Median ($k$M)

The problem of $k$-median ($k$M) has been studied extensively, and several approaches have been developed to obtain constant-factor approximations. Three primary methods include LP rounding, local search, and bi-point rounding. The LP rounding approach, which utilizes a half-integral solution as an intermediate step, was first applied by Charikar et al. (1999) to achieve the first constant-factor approximation with a ratio of $6\frac{2}{3}$. Later, Charikar and Li (2012) enhanced this method by employing dependent rounding, improving the approximation to a factor of $3.25$.

The second approach, local search, was initially used by Arya et al. (2001) to obtain a $(3 + \epsilon)$-approximation. This analysis was later simplified by Gupta and Tangwongsan (2008). Cohen-Addad et al. (2022) improved the local search ratio to $(2.836 + \epsilon)$.

The third approach involves generating an intermediate fractional solution, called a bi-point solution, which is then rounded to an integral solution. Jain and Vazirani (2001) introduced this technique, achieving a $6$-approximation. This was later improved by Jain et al. (2003) to $4$-approximation. Further improvements were made by Li and Svensson (2013), Byrka et al. (2015b), and Cohen et al. (2023), who reduced the factor to $2.7322$, $2.674$, and $2.6705$ respectively.

The current best approximation factor of $2.613$ is due to Gowda et al. (2023) using the bi-point solution technique. On the other hand, a hardness lower bound of $1.736$ by Jain et al. (2002) has been known for over 20 years. A summary of approximation results is provided in Table 2.6.

| Approximation Factor | Reference | Technique |
|---|---|---|
| $6\frac{2}{3}$ | Charikar et al. (1999) | LP rounding |
| $6$ | Jain and Vazirani (2001) | Bi-point rounding |
| $4$ | Jain et al. (2003) | Bi-point rounding |
| $3 + \epsilon$ | Arya et al. (2001) | Local Search |
| $3 + \epsilon$ | Gupta and Tangwongsan (2008) | Local Search |
| $3.25$ | Charikar and Li (2012) | LP rounding |
| $2.7322$ | Li and Svensoon (2013) | Bi-point rounding |
| $2.674$ | Byrka et al. (2015b) | Bi-point rounding |
| $2.836 + \epsilon$ | Cohen-Addad et al. (2022) | Local Search |
| $2.6705$ | Cohen-Addad et al. (2023) | Bi-point rounding |
| $2.613$ | Gowda et al. (2023) | Bi-point rounding |

Table 2.6: State of the Art: Classical $k$-Median

## 2.5 Capacitated $k$-Median (C$k$M)

Finding a (true) constant factor approximation for capacitated $k$-Median problem ($k$M) is one of the biggest open questions in the literature of theoretical computer science. The standard LP of the problem has an unbounded integrality gap when either of the two constraints - capacity or cardinality, is allowed to be violated by a factor of less than $2$ without violating the other. Indeed, polynomial time constant factor approximation algorithms are known if the capacities or the cardinality can be violated.

Charikar et al. (1999) first studied C$k$M for the case of uniform capacities and gave a $16$ factor approximation algorithm violating capacities by a factor of $3$. Li (2014) and Grover et al. (2018a) reduced the violation in capacities to $(2 + \epsilon)$ at a $O(1/\epsilon^2)$ loss in approximation factor, for a fixed $\epsilon > 0$. The barrier of $2$ in capacity violation was broken by Byrka et al. (2016) by giving a $O(1/\epsilon^2)$ approximation algorithm at $(1 + \epsilon)$ factor loss in capacity using strengthened LP. With regard to cardinality violation, Korupolu et al. (2000) used local search to give a $(1 + 5\epsilon)$ factor approximation, violating the cardinality constraint by a factor of $(5 + \epsilon)$. They also gave another $(1 + \epsilon)$ factor approximation at

$(5+5\epsilon)$ factor violation in the cardinality. The loss in cardinality was reduced to $3$ factor by Grover et al. (2018b) for a $(5+\epsilon)$ factor approximation algorithm using a similar local search technique. Li (2015) further reduced the violation in cardinality to $(1+\epsilon)$ at $O(1/\epsilon^2)$ factor loss in approximation factor by rounding the solution to a strengthened LP. Refer to Table 2.7 for the summary of the results on the uniform capacitated $k$-Median problem.

| **Approximation Factor** | **Capacity Violation** | **Reference** | **Technique** |
|---|---|---|---|
| 16 | 3 | Charikar et al. (1999) | LP rounding |
| $O(1/\epsilon^2)$ | $2+\epsilon$ | Li (2014) | LP rounding |
| $O(1/\epsilon^2)$ | $2+\epsilon$ | Grover et al. (2018a) | LP rounding |
| $O(1/\epsilon^2)$ | $1+\epsilon$ | Bryka et al. (2016) | Rounding with strengthened LP |
| **Approximation Factor** | **Cardinality Violation** | **Reference** | **Technique** |
| $1+5\epsilon$ | $5+\epsilon$ | Korupolu et al. (2000) | Local Search |
| $1+\epsilon$ | $5+\epsilon$ | Korupolu et al. (2000) | Local Search |
| $O(1/\epsilon^2)$ | $1+\epsilon$ | Li (2015) | Rounding with strengthened LP |
| $5+\epsilon$ | 3 | Grover et al. (2018b) | Local Search |

Table 2.7: State of the Art: Uniform Capacitated $k$-Median

Byrka et al. (2015a) studied the problem with general capacities to obtain a $O(1/\epsilon)$ factor approximation algorithm with $(3+\epsilon)$ factor violation in capacities, for a fixed $\epsilon > 0$, by using LP rounding. The violation in capacity was reduced to $(1+\epsilon)$ by Demirci and Li (2016) at $O(1/\epsilon^5)$ factor loss in approximation factor by using strengthened LP. With regard to cardinality violation, $O(\frac{1}{\epsilon^2}\log\frac{1}{\epsilon})$ factor approximation algorithm with $(1+\epsilon)$ factor violation in cardinality was given by Li (2016), again by using strengthened LP.

Adamczyk et al. (2019) were the first ones to break the barrier of constant-factor approximability by designing a $(7+\epsilon)$ factor approximation that runs in time FPT in $k$ and $\epsilon$. This was later improved by Cohen-Addad and Li (2019) to a $(3+\epsilon)$-approximation for C$k$M, in time FPT in $k$ and $\epsilon$.

Refer Table 2.8 to see the summary of the results for C$k$M with general capacities.

| **Polynomial Time Approximations** | | | |
|---|---|---|---|
| **Approximation Factor** | **Capacity Violation** | **Reference** | **Technique** |
| $O(1/\epsilon)$ | $3+\epsilon$ | Bryka et al. (2015a) | LP rounding |
| $O(1/\epsilon^5)$ | $1+\epsilon$ | Demirci and Li (2016) | Rounding with strengthened LP |
| **Approximation Factor** | **Cardinality Violation** | **Reference** | **Technique** |
| $O(\frac{1}{\epsilon^2}\log\frac{1}{\epsilon})$ | $1+\epsilon$ | Li (2016) | Rounding with strengthened LP |
| **FPT Approximation (FPT in $k$ and $\epsilon$)** | | | |
| **Approximation Factor** | | **Reference** | **Technique** |
| $7+\epsilon$ | | Adamczyk et al. (2019) | Combinatorial |
| $3+\epsilon$ | | Cohen-Addad and Li (2019) | Using Coresets |

Table 2.8: State of the Art: Capacitated $k$-Median

## 2.6 $k$-Median with Outliers ($k$MO)

Similar to the facility location, the standard linear programming (LP) formulation for the $k$-Median with Outliers ($k$MO) problem has an unbounded integrality gap. However, unlike FLO, it is not straightforward to overcome this challenge using techniques like guessing the most expensive facility. As a result, the $k$MO problem remains less understood in the literature.

The problem was first introduced by Charikar et al. (2001) in their pioneering work on outliers and penalties. They presented a bi-criteria approximation algorithm for $k$MO, achieving an approximation factor of $4(1+\frac{1}{\epsilon})$ and a $(1+\epsilon)$ factor violation in the number of outliers, using a primal-dual framework. Their approach involves guessing the cost of the optimal solution (since the exact optimal cost cannot be determined, they approximate it within a factor of $(1+\epsilon')$ for some small $\epsilon' > 0$). This guessed cost is then used to construct a new instance of the $k$-Median problem with penalties, and the algorithm for $k$-Median with penalties is applied to this transformed instance to obtain the desired approximation result.

Friggstad et al. (2018) employed natural multiswap local search heuristics to address outliers in the $k$-Median problem. Their approach provides a $(3+\epsilon)$-factor approximation with a $(1+\epsilon)$-factor violation in the cardinality. One of the key challenges in local search algorithms for $k$-Median with Outliers is handling clients that are not necessarily outliers in both global and local solutions. This is difficult because bounding their service cost does not directly (or indirectly) follow from the triangle inequality. To overcome this challenge, they introduce a novel mapping between the outliers in a global optimal solution and those in the local optimal solution. Furthermore, they demonstrate that natural local search heuristics with constant size operations, which do not violate the number of clusters and outliers for $k$MO, lead to an unbounded locality gap.

The first true constant approximation for the problem was achieved by Chen (2008), although the approximation factor is relatively large and unspecified. Chen's algorithm is based on the Lagrangian relaxation framework developed by Jain and Vazirani (2001) for the $k$-Median problem. Two solutions are computed for the Facility Location (FLO) problem, which corresponds to the Lagrangian relaxation of $k$MO: one solution with at most $k$ facilities ($S_-$) and another with at least $k+1$ facilities ($S_+$). When $S_+$ uses at least $k+2$ facilities, a greedy algorithm is employed to merge the two solutions. However, when $S_+$ uses exactly $k+1$ facilities, merging the solutions becomes challenging because the cost of $S_-$ cannot be directly bounded by the cost of the optimal solution. To address this issue, Chen (2008) employs a successive local search algorithm for $k$MO with penalties, where the penalties are gradually increased over the iterations.

Krishnaswamy et al. (2018) applied iterative rounding on a strengthened linear program (LP) to achieve a $7.081$-approximation, significantly improving upon the large implicit constant approximation factor of Chen (2008). While the standard LP for $k$MO has an unbounded integrality gap, they demonstrate that the gap arises primarily from the difference between an almost-integral solution (those with at most two fractionally open facilities) and a fully-integral solution. In fact, a key technical contribution of their work is showing that the standard LP performs well when we are willing to accept solutions that open at most $k + 1$ facilities. They propose an iterative algorithm that rounds the solution of the standard LP to achieve a $(7.081 + \epsilon)$-factor approximation with the addition of only one extra facility. Furthermore, through a preprocessing and sparsification technique, they show how to convert this almost-integral solution into a fully-integral one, introducing only a small additive loss of $\epsilon$ in the cost, where $\epsilon > 0$ is a small constant.

The $(7.081 + \epsilon)$ approximation ratio achieved by Krishnaswamy et al. (2018) was further improved to $(6.994+\epsilon)$ by Gupta et al. (2021) through a refined iterative rounding algorithm. This improvement is based on an analysis of the extreme points of certain set-cover-like LPs. These extreme points emerge during the intermediate steps of the iterative rounding process, and by exploiting their structural properties, Gupta et al. (2021) obtain a $(6.387 + \epsilon)$-approximate solution with at most $k + 1$ facilities. To address the issue of opening an additional facility, they complement the sparsification and preprocessing techniques from Krishnaswamy et al. (2018) with a post-processing step. This post-processing step introduces a small additional loss in the approximation ratio, ultimately yielding a final $(6.994 + \epsilon)$-factor approximation algorithm.

Goyal et al. (2020) obtained a $(3 + \epsilon)$ approximation algorithm, FPT in $k$, the number of outliers and $\epsilon$. Later, Agrawal et al. (2023) introduced an approximation-preserving reduction from the $k$MO to the $k$-Median problem, which is FPT in $k$, the number of outliers, and $\epsilon$. As a result, they achieved an improved $(1 + 2/e + \epsilon)$-approximation for $k$MO, which is also FPT in $k$, the number of outliers, and $\epsilon$ but the running time is slightly worse than Goyal et al. (2020). Their approach begins by creating an instance of $k'$-median problem with $k' = k + L$ and obtaining a constant-factor approximation for the instance by using any approximation algorithm for $k$-Median problem. They then perform a sampling process to obtain a weighted set of points, where the cost of the weighted set can be related to the service cost of the original set of points. Specifically, for any set of $k$ centers, with high probability, the difference between the original and the weighted costs remains "small," even after excluding up to $L$ outliers from both sets. This concentration bound intuitively holds because the sample size is sufficiently large relative to both $k$ and the number of outliers, ensuring that the approximation remains accurate.

Refer to Table 2.9 for a summary of approximation algorithms for $k$MO.

| **Polynomial Time Approximations** | | | |
|---|---|---|---|
| **Approximation Factor** | **Outlier Violation** | **Reference** | **Technique** |
| $4(1+\frac{1}{\epsilon})$ | $1+\epsilon$ | Charikar et al. (2001) | Primal Dual |
| **Approximation Factor** | **Cardinality Violation** | **Reference** | **Technique** |
| $3+\epsilon$ | $1+\epsilon$ | Friggstad et al. (2018) | Local Search |
| **Approximation Factor** | | **Reference** | **Technique** |
| Large, unspecified | | Chen (2008) | Successive Local Search and greedy |
| $7.081+\epsilon$ | | Krishnaswamy et al. (2018) | Iterative rounding |
| $6.994+\epsilon$ | | Gupta et al. (2021) | Iterative rounding |
| **FPT Approximations (FPT in $k$, $L$ and $\epsilon$)** | | | |
| **Approximation Factor** | | **Reference** | **Technique** |
| $3+\epsilon$ | | Goyal et al. (2020) | Combinatorial |
| $1+2/e+\epsilon$ | | Agrawal et al. (2023) | Combinatorial |

Table 2.9: State of the Art: $k$-Median with Outliers

## 2.7 Other Related Work

The $k$-Center problem is a fundamental facility location problem that closely resembles the $k$-Median problem, with the key difference being that in the $k$-Center problem, the objective is to minimize the maximum service cost, rather than the total service cost. The approximation complexity of this problem is well-established, with a simple 2-approximation greedy algorithm known to be optimal unless $P = NP$, by Hochbaum (1985). Building upon this, various extensions of the $k$-Center problem have been explored in the literature.

The $k$-Center with Outliers ($k$CO) problem was introduced by Charikar et al. (2001), who developed a $(3 + \epsilon)$-approximation algorithm using a greedy approach. Similar to other work on the $k$-Center problem, the algorithm first guesses the optimal solution value, denoted as $R$. Disks of radius $R$ are constructed around each point. Initially, all points are uncovered. The algorithm then selects the point that covers the maximum number of uncovered points within its disk. After each selection, all points within a distance of $3R$ from the chosen point are marked as covered. This process is repeated $k$ times. The key distinction from the greedy algorithm for the non-outlier version of the problem lies in selecting disks that cover the maximum number of uncovered points within a radius of $R$, but marking all points within a $3R$ radius as covered. Interestingly, the algorithm fails when attempting to mark points within a $2R$ radius instead.

After 15 years, the ratio for $k$CO was improved to $2$ by Chakrabarty et al. (2016) which is best possible unless $P = NP$. They give this result as a special case of a problem called as *non-uniform $k$-Center* (NU$k$C) problem. To give a bi-criteria for NU$k$C, which in turn gives a 2-approximation for $k$C, they show a strong connection between NU$k$C and the so-called *resource minimization for fire containment problem on trees*. This connection is one of the main findings of their paper.

To the best of our knowledge, the only work that combines capacity constraints to the outliers constraint is in the context of $k$-Center. Cygan and Kociumaka (2014) developed 25-factor and 23-factor approximation algorithms for the *Capacitated $k$-Center with Outliers* problem, addressing non-uniform and uniform capacities, respectively.

They demonstrate that the standard LP formulation for the problem has an unbounded integrality gap, and therefore, they strengthen the LP to obtain the desired approximation results. For strengthening the LP, a key concept introduced in their work is the notion of a *skeleton*. The skeleton can be thought of as a solution that is structurally similar to at least one optimal solution. The process of rounding the solution of the strengthened LP uses techniques similar to those employed in *Capacitated $k$-Center* algorithm (2015). On the other hand, Goyal and Jaiswal (2023) designed a $2$-factor tight FPT approximation for the problem parameterized by $k$ and the number of outliers.

In this thesis, we handle both the capacity and outlier constraint simultaneously in the context of FL and $k$M. In parallel, Jaiswal and Kumar (2023) also studied $k$M in the presence of both capacity and outlier constraints and obtained a $(3 + \epsilon)$ FPT approximation, parameterized by $k$, the number of outliers, and $\epsilon$. We present the same result in this thesis using a different technique.

# Chapter 3

# $(6.373 + \epsilon)$-Approximation for CFLO with Uniform Facility Opening Costs[1]

## 3.1 Introduction

In this chapter, we study Capacitated Facility Location with Outliers (CFLO) when facility opening costs are uniform. For ease of disposition of ideas, we first present a result for the Capacitated Facility Location problem (CFL) and then extend it to the CFLO problem. We start with a formal definition of CFL problem. Recall that $(\mathcal{P}, d)$ denotes a metric space where $\mathcal{P}$ is a finite set of points and $d : \mathcal{P} \times \mathcal{P} \to \mathbb{R}^+$ is a distance function satisfying triangle inequality and symmetry.

**Definition 3.1** (Capacitated Facility Location with Uniform Opening Costs)**.** *We are given a set, $X \subseteq \mathcal{P}$, of $n$ clients and a set, $F \subseteq \mathcal{P}$ of $m$ facility locations. A facility $i \in F$ has a facility opening cost $f$ and a capacity $u(i) \in \mathbb{N}$. The objective is to find,*

- *a set $S \subseteq F$ of facilities to open and,*
- *an assignment $\sigma : X \to S$ respecting the capacities, i.e., for each facility $i \in S$, $|\sigma^{-1}(i)| \leq u(i)$*

*such that the total cost, $f|S| + \sum_{j \in X} d(j, \sigma(j))$, is minimized.*

[1]The results presented in this chapter appeared in Dabas et al. (2024)

Levi et al. (2012) obtained a 5-approximation for CFL with uniform facility costs, Aardal et al. (2015) improved this guarantee to $(4.562 + \epsilon)$ and this was further improved to a 4-approximation by Kao (2023b) in parallel to our work. Our result is a very simple 2-operation local search algorithm that is a $(3.733 + \epsilon)$ approximation for CFL with uniform facility costs (see Theorem 1.1). The analysis of our algorithm is quite simple, and the technique can be adapted to handle outliers.

**Theorem 1.1.** *There exists a polynomial time local search procedure with* $2$ *operations that yields a locally optimal solution which is a* $(3.733+\epsilon)$*-approximation to the optimum solution of the capacitated facility location problem with uniform facility opening costs.*

We next study CFLO with uniform facility opening costs. The hard constraints of capacities and number of outliers make the CFLO problem very challenging, and to the best of our knowledge, no approximation is known for this problem. The problem is formally defined as follows:

**Definition 3.2** (Capacitated Facility Location with Outliers and Uniform Opening Costs)**.** *In addition to the input for the capacitated facility location problem, we are given a bound,* $L \leq n$*, on the number of outliers permitted in a feasible solution. The objective is to find,*

- *a set* $S \subseteq F$ *of facilities,*
- *a set* $X' \subseteq X$ *of outliers of size at most* $L$*,*
- *an assignment* $\sigma : (X \setminus X') \to S$ *respecting the capacities, i.e., for each facility* $i \in S$, $|\sigma^{-1}(i)| \leq u(i)$

*such that the cost,* $f|S| + \sum_{j \in X \setminus X'} d(j, \sigma(j))$*, is minimized.*

Our second and main result is the first constant factor approximation for CFLO assuming uniform facility opening costs. We use local search and extend the ideas of CFL problem to obtain a $(6.373+\epsilon)$ approximation for CFLO problem (see Theorem 1.2).

**Theorem 1.2.** *There exists a polynomial time local search procedure with $2$ operations that yields a locally optimal solution which is a $(6.373+\epsilon)$-approximation to the optimum solution of the capacitated facility location problem with outliers and uniform facility opening costs.*

Our local search algorithms for both CFL and CFLO require only 2 operations, one of which is an `add` operation to add a facility not already part of the solution. For CFL, our second operation `open` opens a facility, $t$, and closes a subset of facilities, $S'$. The best such set - under certain restrictions - can be found by solving a knapsack problem, and this operation has also been key in prior work on capacitated facility location. For the outlier version, we modify the open operation to open $2$ facilities, $t_1, t_2$ and close a subset $S'$ of facilities. We impose novel restrictions - clients served by the facilities in $S'$ are either served by $t_1$ or made outliers, and $t_2$ serves the right outliers from the current solution - on the operation to allow us to find the best $S'$ in polynomial time, and this involves solving a 2-dimensional knapsack problem.

As with most local search algorithms, we need to put together a suitable set of inequalities to bound the quality of the locally optimum solution. The outliers pose a challenge here and our key contribution is to define a suitable bijection between the outliers in the locally optimum and global optimum solutions. Unlike CFL where the choice of inequalities is such that each facility of the optimum solution is opened only a bounded number of times, the presence of outliers does not permit us this luxury. In fact, the inequalities we use to bound the facility cost of the locally optimum solution might require some facilities to be opened many times. However, since facility costs are uniform, we can amortize this and argue that, on average, each facility of the optimum solution is opened only a small number of times.

We conjecture that the locality gap for FLO with non-uniform facility opening costs is unbounded, even in the uncapacitated case. The locality gap, of course, depends on the specific set of local search operations allowed. Friggstad et al. (2019) gave an example to show that any constant size multi-swap operation can not yield a local search algorithm with a bounded locality gap when the facility opening costs are general. The example provided in Friggstad et al. (2018) can be overcome by employing one of the non-constant swap operations introduced in our algorithm. However, in Section 3.7, we present a challenging example that highlights an issue even with the non-constant size swaps. In this case, escaping the unbounded locality gap requires an operation that, in essence, involves solving an instance of the facility location with outliers itself. We also modify our example to show that even after allowing constant factor violation in outliers, escaping the locality gap involves solving an instance of facility location with outliers.

### 3.1.1 Organisation of the Chapter

The remainder of the chapter is organized as follows. We start by introducing some notations in Section 3.2. In Section 3.3, we discuss some prior work by Pál et al. (2001) on the CFL problem, which is used to obtain our result for CFL with uniform facility costs in Section 3.4. This also serves as a starting point for our discussion on CFLO in Section 3.5. In Section 3.6, we analyze the cost of our locally optimum solution for CFLO. In Section 3.7, we present a challenging example for the problem when the facility opening costs are non-uniform.

## 3.2 Notations

In this section, we introduce some notation that will be used throughout the chapter. Given a set $S$ of facilities, the assignment of clients to facilities can be determined by solving a minimum cost flow problem (/with outliers for CFLO). Hence, the set of open facilities completely determines the solution, and so we use $S$ to denote both the solution and the set of open facilities in the solution. For a solution $S$, let $C(S) = C_f(S) + C_s(S)$ denote the cost of solution $S$ where $C_s(S)$ and $C_f(S)$ denote the service cost and facility cost of solution $S$ respectively. Let $S$ be a locally optimum solution and $S^*$ be any feasible solution for a given instance of the problem. Let $X(s)$ be the set of clients served by facility $s$ in $S$ and $X(A) = \cup_{s \in A} X(s)$ for $A \subseteq S$ denote the set of all clients served by facilities in $A$ in solution $S$. Similarly, let $X^*(t)$ be the set of clients served by facility $t$ in $S^*$ and $X^*(A) = \cup_{t \in A} X^*(t)$ for $A \subseteq S^*$ denote the set of all clients served by facilities in $A$ in solution $S^*$. Further, let $\sigma : X \rightarrow S$ denote the assignment of clients in $S$ and $\sigma^* : X \to S^*$ denote the assignment of clients in $S^*$.

In the case of CFLO, let $O$ and $O^*$ represent the set of outliers in solutions $S$ and $S^*$, respectively. For brevity of notations, let $\sigma : (X \setminus O) \to S$ be the assignment of clients in $S$ and $\sigma^* : (X \setminus O^*) \to S^*$ be the assignment of clients in $S^*$.

## 3.3 Adaptions of Previous Work

In this section, we will reproduce the prior work of Pál et al. (2001) on the CFL problem, with some minor modifications. The techniques discussed in this section will be used in the analysis of the local search algorithm for CFL with uniform costs in Section 3.4. The modifications help us extend the same ideas to CFLO with uniform costs in Section 3.6.

Pál et. al. (2001) construct a directed bipartite graph $G = (F \cup X, E)$ with $F, X$ as the two sides of the vertex set. If $j \in X(i)$, edge $(i, j)$ is added to $E$ and give it a length $d(i, j)$. Similarly, if $j \in X^*(i)$, edge $(j, i)$ is added to $E$ and give it a length $d(i, j)$. Every vertex $j \in X$ has one incoming and one outgoing edge. The edges of $E$ are then decomposed into cycles and maximal paths. Note that the number of maximal paths starting from $i \in S$ equals $\max\{0, |X(i)| - |X^*(i)|\}$ and the number of maximal paths ending at $i \in S^*$ equals $\max\{0, |X^*(i)| - |X(i)|\}$. By triangle inequality, a maximal path from $s \in S$ to $t \in S^*$ has a length of at least $d(s, t)$. Further, the total length of all maximal paths is, at most, the total length of all edges in $E$, which equals $C_s(S) + C_s(S^*)$.

Pál et al. (2001) next formulate a transhipment problem where a facility $i \in S$ is a supply node with supply $|X(i)|$ and a facility $i \in S^*$ is a demand node with demand $|X^*(i)|$. Note that a facility $i \in S \cap S^*$ is both a supply and a demand node. The cost of shipping 1 unit of flow from $s \in S$ to $t \in S^*$ equals $d(s, t)$. Total supply over all supply nodes ($\sum_{i \in S} |X(i)|$) = total demand over all demand nodes ($\sum_{i \in S^*} |X^*(i)|$) = $n$.

Let $x$ be an optimum solution to the above problem, and let $x(s, t)$ be the amount of flow shipped from $s \in S$ to $t \in S^*$. Pál et al. (2001) next construct an undirected *exchange graph*, $H = (V', E')$ where $V'$ has a vertex for each facility in $S$ and a vertex for each facility in $S^*$; thus a facility in $S \cap S^*$ corresponds to two vertices in $V'$. Define $E' = \{(s, t) | x(s, t) > 0\}$. The exchange graph $H$ and the optimum solution $x$ to the transhipment problem have important properties stated in Lemmas 3.1, 3.2 and 3.3.

**Lemma 3.1.** *$(i)$ $H$ is acyclic.*

*$(ii)$ If $i \in S \cap S^*$ then $H$ has an edge between the two vertices in $V'$ corresponding to $i$ and one of these vertices is a leaf of $H$.*

**Proof.** $(i)$ Suppose if possible, there exists a cycle $C$. Partition the edges of $C$ into two sets (say, $E_1$ and $E_2$) consisting of alternate edges. Let $S_1$ and $S_2$ be the sum of length of edges in $E_1$ and $E_2$ respectively. We will first prove $S_1 = S_2$. If $S_1 > S_2$, we can obtain a flow of lower cost by decreasing one unit of flow from edges in set $E_1$ and increasing one unit of flow on edges in $E_2$; this is a contradiction to the fact that $x$ is an optimal flow. Thus, $S_1 = S_2$. In this case, the graph can be made acyclic by iteratively decreasing one unit of flow on edges in one of the sets $E_1$ or $E_2$, while simultaneously increasing one unit of flow on edges in the other set.

$(ii)$ Since $d(i,i) = 0$, for $i \in S \cap S^*$ any optimal solution for the transhipment problem would ship $\min\{|X(i)|, |X^*(i)|\}$ flow from the supply node $i$ to the demand node $i$. If $\min\{|X(i)|, |X^*(i)|\}$ = $|X(i)|$, then $i \in S$ is a leaf whereas if $\min\{|X(i)|, |X^*(i)|\}$ = $|X^*(i)|$, then $i \in S^*$ is the leaf.

□

**Lemma 3.2.** *$(i)$ $\forall s \in S$, $\sum_{t \in S^*} x(s,t) = |X(s)| \leq u(s)$.*

*$(ii)$ $\forall t \in S^*$, $\sum_{s \in S} x(s,t) = |X^*(t)| \leq u(t)$.*

**Proof.** $(i)$ Firstly note that, for any $s \in S$, $\sum_{t \in S^*} x(s,t)$ can not be greater than $|X(s)|$ because the maximum supply at node $s$ is $|X(s)|$. Suppose if possible, for some $s \in S$, $\sum_{t \in S^*} x(s,t) < |X(s)|$. Then, $\sum_{s \in S} \sum_{t \in S^*} x(s,t) < \sum_{s \in S} |X(s)| = n$. This is a contradiction to the fact that $x$ is a feasible solution because using less than $n$ units of supply is not sufficient to serve $n$ units of demand. Therefore, $\sum_{t \in S^*} x(s,t) = |X(s)|$. The inequality, $|X(s)| \leq u(s)$, follows by capacity constraints imposed on the solution $S$.

$(ii)$ Firstly note that, for any $t \in S^*$, $\sum_{s \in S} x(s,t)$ can not be greater than $|X^*(t)|$ because the maximum demand at node $t$ is $|X^*(t)|$. Suppose if possible, for some $t \in S^*$, $\sum_{s \in S} x(s,t) < |X^*(t)|$. Then, $\sum_{t \in S^*} \sum_{s \in S} x(s,t) < \sum_{t \in S^*} |X(t)| = n$ which is a contradiction to the fact that $x$ is a feasible solution serving $n$ units of demand. Therefore, $\sum_{s \in S} x(s,t) = |X^*(t)|$. The inequality, $|X^*(t)| \leq u(t)$, follows by capacity constraints imposed on the solution $S^*$.

□

**Lemma 3.3.** $\sum_{s \in S, t \in S^*} x(s,t)d(s,t) \leq C_s(S) + C_s(S^*)$.

**Proof.** To prove this, we construct a feasible solution, $x'(s,t)$, for the transhipment problem of cost no more than $C_s(S) + C_s(S^*)$. For $s \in S \cap S^*$, we send $x'(s,s) = \min\{|X(s)|, |X^*(s)|\}$ units of flow from the supply node $s$ to the demand node $s$. since $d(s,s) = 0$, this can be done at $0$ cost. Note that, for $s \in S \setminus S^*$, the remaining supply is same as the original supply whereas for nodes $s \in S \cap S^*$, the remaining supply is $|X(s)| - \min\{|X(s)|, |X^*(s)|\} = \max\{0, |X(s)| - |X^*(s)|\}$. Consider the maximal paths in the decomposition of edges in $E$. Let $w(s,t)$ be the number of paths starting at $s$ and ending at $t$. Send $x'(s,t) = w(s,t)$ unit of flow from $s$ to $t$. Note that the total amount of flow from $s$ is no more than the supply at $s$ because the number of maximal paths starting from $s \in S \cap S^*$ equals $\max\{0, |X(s)| - |X^*(s)|\}$ whereas number of paths starting from $s \in S \setminus S^*$ is $|X(s)|$. Similarly, on the demand side, the total flow into $t \in S^*$ is exactly equal to demand at node $t$ as number of maximal paths ending at $t \in S \cap S^*$ equals $\max\{0, |X^*(t)| - |X(t)|\}$ and the number of maximal paths ending at $t \in S^* \setminus S$ equals $|X^*(t)|\}$. Recall that the total length of all maximal paths is at most $C_s(S) + C_s(S^*)$. Hence, we have defined a feasible solution $x'(s,t)$ to the transhipment problem of cost at most $C_s(S) + C_s(S^*)$ and the cost of the optimal solution $x(s,t)$ can only be lower. □

We remark that the above description of Pál et. al. (2001) is not entirely accurate. In particular, the exchange graph constructed by Pál et. al. (2001) has a vertex for each facility in $S \setminus S^*$ and a vertex for each facility in $S^*$. Our more symmetric construction of the exchange graph helps us extend it to handle outliers.

## 3.4 Capacitated Facility Location with Uniform Facility Costs

Our algorithm does local search: we start with an arbitrary feasible solution (set of open facilities of total capacity at least $n$) and keep performing local search steps till they improve the cost of the solution. Let $S$ be the solution at any step in this algorithm.

### 3.4.1 Local Search Operations

**add($t$):** For $t \notin S$, if $C(S \cup \{t\}) < C(S)$ then $S \leftarrow S \cup \{t\}$. A facility $t$ which is not in the current solution $S$ is added to $S$ if its addition improves the cost of the solution.

We define open($t, S'$) as an operation which opens a facility $t \in F$ and closes $S' \subseteq S$. If $t \in S$, then the operation is defined only if $S'$ contains $t$; in this case, the operation closes facilities in $S' \setminus \{t\}$. In determining the cost of this operation, we assume all clients in $X(S')$ are reassigned to $t$. Thus the reduction in $C(S)$ if this operation is performed is $\sum_{s \in S'} \sum_{j \in X(s)} (d(j,s) - d(j,t)) + f(|S'| - 1)$.

**open($t$):** This is the same as the operation open($t, S'$) for a subset $S' \subseteq S$ for which the cost of the operation is minimum. Given $t$, the problem of finding the optimal such $S'$ can be formulated as a knapsack problem and solved in polynomial time (Lemma 3.4). Note that if $t \in S$ then open($t$) does not result in multiple copies of $t$ in $S$. Instead, we include $t$ in $S'$ and open($t$) then leads to closing of facilities in $S' \setminus t$ and their clients getting assigned to $t$. The operation $S \leftarrow S \setminus S' \cup \{t\}, S' \subseteq S$, is performed only if it improves the cost of the solution $S$.

**Lemma 3.4** (Pal et al. (2001))**.** *Given $t$, one can, in polynomial time, find a set $S'$ that minimizes the cost of open($t, S'$) among all subsets $S' \subseteq S$.*

**Proof.** For a fixed $s \in S$, let $v(s)$ represent the cost saved by closing $s \in S$. If $t \notin S$, then $v(s) = \sum_{j \in X(s)} d(j,s) + f - \sum_{j \in X(s)} d(j,t)$ otherwise $v(s) = \sum_{j \in X(s)} d(j,s) - \sum_{j \in X(s)} d(j,t)$. The value of a set $S' \subseteq S$ is $\sum_{s \in S'} v(s)$. We find the set $S'$ of the maximum value in polynomial time by solving the following knapsack problem: we have an object corresponding to each $s \in S$. The weight and value of the object corresponding to $s$ are $|X(s)|$ and $v(s)$, respectively. We wish to pick a set of objects of total weight at most $u(t) - |X(t)|$ and have maximum total profit. $\square$

The algorithm stops if neither of the two operations improves the cost of the solution. Note that if a facility in $S$ does not serve any client, we close that facility. The solution $S$ at the end of the algorithm is a locally optimum solution. The number of improvement steps of this local search algorithm can be made polynomial in the input size and $1/\epsilon$ by requiring that a local improvement be made only if it improves the cost of the solution by a $(1+\epsilon)$-factor. Refer Korupolu et al. (2000) for details.

### 3.4.2 Analysis

The service cost of the solution $S$ can be bounded by the add operation.

**Lemma 3.5** (Arya et al. (2004), Lemma 4.1)**.** $C_s(S) \leq C_s(S^*) + C_f(S^*) = C(S^*)$.

We need a suitable set of inequalities to bound the facility opening cost of the locally optimum solution $S$. We construct a bipartite graph $G$ and formulate a transhipment problem similar to Pál et al. (2001) as described in Section 3.3. Let $x$ be the optimum solution of the transhipment problem and $H$ be the exchange graph defined in Section 3.3.

Every connected component of $H$ is a tree and contains an edge of $E'$ and hence a vertex in $S^*$. We root each tree in $H$ at an arbitrary vertex in $S^*$. Let $\mathcal{L}$ be the set of facilities in $S$ that are leaves in $H$. For $t \in S^*$, let $C(t)$ be the set of children of $t$ and let $C_{\mathcal{L}}(t)$ be the set $C(t) \cap \mathcal{L}$. Further, let $F^* = \{t \in S^* : |C_{\mathcal{L}}(t)| \geq 1\}$. Thus, for $t \in F^*, C_{\mathcal{L}}(t) \neq \phi$.

Since our solution is locally optimal, if we open $t \in F^*$ and close facilities in $C_{\mathcal{L}}(t)$, we do not improve the cost of the solution. We capture this fact by writing an inequality that says that the cost of open($t, C_{\mathcal{L}}(t)$) is non-negative for every $t \in F^*$. (Refer Figure 3.1). Note that if $t \in S$ then by Lemma 3.1, $t \in C_{\mathcal{L}}(t)$ and hence the operation open($t, C_{\mathcal{L}}(t)$) is well-defined.

$(i)$ Since $s \in C_{\mathcal{L}}(t)$ is a leaf of $H$, $x(s,t) = |X(s)|$. In operation open($t, C_{\mathcal{L}}(t)$), $t \in F^*$, the number of clients assigned to $t$ is $\sum_{s \in C_{\mathcal{L}}(t)} |X(s)| = \sum_{s \in C_{\mathcal{L}}(t)} x(s,t) \leq$

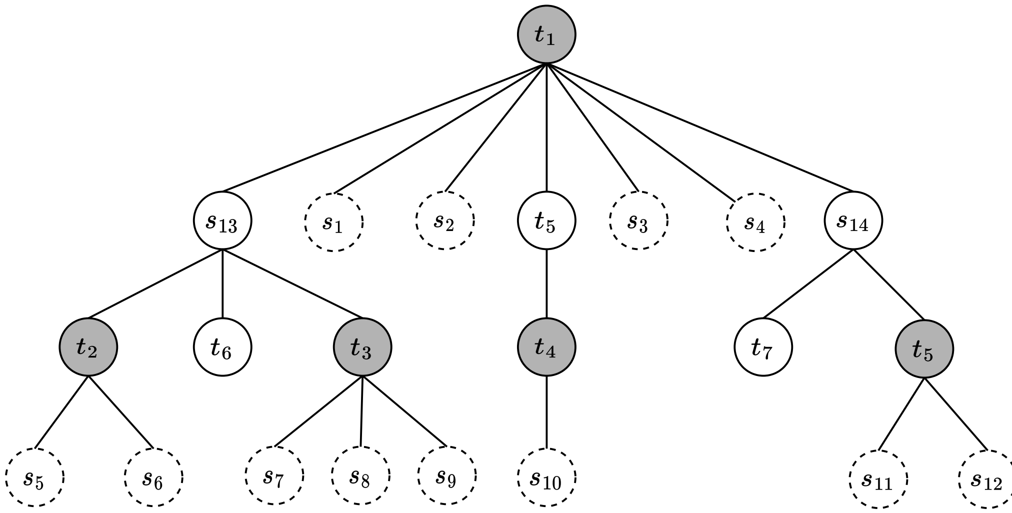



Figure 3.1: Illustration of a tree in $G$ rooted at $t_1 \in S^*$. The dashed circles represent facilities in $\mathcal{L}$, (facilities labelled $s_1$-$s_{12}$), and the solid grey circles represent facilities in $F^*$ (facilities labelled $t_1$-$t_5$). The operations, open($t_1$, $\{s_1, s_2, s_3, s_4\}$), open($t_2$, $\{s_5, s_6\}$), open($t_3$, $\{s_7, s_8, s_9\}$), open($t_4$, $s_{10}$) and, open($t_5$, $\{s_{11}, s_{12}\}$) help us write the required inequalities.

$\sum_{s \in S} x(s,t) \leq u(t)$ due to Lemma 3.2 $(ii)$ and hence the capacity of $t$ is not violated.

$(ii)$ The increase in service cost over all operations open($t, C_{\mathcal{L}}(t)$), $t \in F^*$, is bounded by $\sum_{t \in F^*} \sum_{s \in C_{\mathcal{L}}(t)} x(s,t)d(s,t) \leq \sum_{t \in S^*} \sum_{s \in S} x(s,t)d(s,t) = C_s(S) + C_s(S^*)$ due to Lemma 3.3.

$(iii)$ The decrease in facility opening costs over all operations open($t, C_{\mathcal{L}}(t)$), $t \in F^*$, is $f \cdot (|\mathcal{L}| - |F^*|)$.

Since $S$ is a locally optimum solution, using points $(ii)$, and $(iii)$ above, we get

$$f \cdot |\mathcal{L}| \leq C_s(S) + C_s(S^*) + f \cdot |F^*|. \tag{3.1}$$

**Lemma 3.6.** *The number of facilities in $S \setminus \mathcal{L}$ is bounded by $|S^*|$.*

**Proof.** Let $T$ be a tree in $H$ rooted at an arbitrary vertex in $S^*$. With every facility in $S$ which is not a leaf of $T$, we associate a child in $S^*$. A facility in $S^*$ can be associated with, at most, one facility of $S$ in this manner. Summing over all trees in $H$ then proves the lemma. $\square$

Combining Equation 3.1 with Lemma 3.6 and then applying Lemma 3.5 yields,

$$C_f(S) \leq 2C_f(S^*) + C_s(S) + C_s(S^*) \leq 2C_s(S^*) + 3C_f(S^*). \tag{3.2}$$

Having bounded both the facility cost and the services cost of our solution, we now employ a scaling technique introduced by Charikar et al. (2005). This involves scaling the facility costs by a factor $\gamma$ and then running the local search algorithm. The service cost of the optimum solution is unchanged but the facility cost scales by $\gamma$. Lemma 3.5 can now be written as

$$C_s(S) \leq C_s(S^*) + \gamma C_f(S^*) \tag{3.3}$$

and inequality 3.2 corresponds to

$$\gamma C_f(S) \leq 2C_s(S^*) + 3\gamma C_f(S^*). \tag{3.4}$$

Thus,

$$C(S) = C_s(S) + C_f(S) \leq (1 + \frac{2}{\gamma})C_s(S^*) + (\gamma + 3)C_f(S^*).$$

For $\gamma = \sqrt{3} - 1$, we get,

$$C(S) \leq (\sqrt{3} + 2)(C_s(S^*) + C_f(S^*)) = (\sqrt{3} + 2)C(S^*),$$

which implies a $(3.733 + \epsilon)$ approximation when $S^*$ is an optimum solution for the given instance of CFL.

## 3.5 The Algorithm for CFLO with Uniform Facility Costs

We start with a feasible solution $S$ and perform the following operations whenever they improve the cost of the solution.

**add($t$):** The operation is the same as defined in Section 3.4.1.

We define multiSwap($t_1, t_2, S'$) as an operation which opens facilities $t_1, t_2 \in F$ ($t_1, t_2$ may be identical) and closes $S' \subseteq S$. As in the open operation in Section 3.4.2 if $t_1 \in S$ then multiSwap($t_1, t_2, S'$) is defined only if $S'$ contains $t_1$; in this case, the operation closes facilities in $S' \setminus \{t_1\}$. In determining the cost of this operation we assume that

$(i)$ clients in $X(S \setminus S')$ continue to be served by the facility that served them in $S$,

$(ii)$ clients in $X(S')$ are either served by $t_1$ or are made outliers, and

$(iii)$ clients in $O$ are either served by $t_2$ or remain outliers.

Thus the reduction in $C_s(S)$ if multiSwap($t_1, t_2, S'$) is performed equals the service cost of clients in $X(S')$ minus the cost of servicing clients in $X(S')$ by $t_1$ if they are not made outliers minus the cost of servicing clients in $O$ by $t_2$ if they are no longer outliers.

**multiSwap($t_1, t_2$):** This is the same as the operation multiSwap($t_1, t_2, S'$) for a set $S' \subseteq S$ which minimizes the cost of the operation. Given $t_1, t_2$ determining the optimal $S'$ can be formulated as a 2-dimensional knapsack problem and solved in polynomial time (Lemma 3.7). The operation $S \leftarrow S \setminus S' \cup \{t_1, t_2\}, t_1, t_2 \in F, S' \subseteq S$, is performed only if it improves the cost of the solution $S$.

**Lemma 3.7.** *Given $t_1, t_2$, a set $S'$ that minimizes the cost for multiSwap($t_1, t_2, S'$) among all subsets $S' \subseteq S$ can be found in polynomial time.*

**Proof.** Given $t_1$ and $t_2$, we first guess the number of outliers in $O$ which will be served by $t_2$; let this number be $k$. Note that $k$ is at most $u(t_2) - |X(t_2)|$. $k$ is also upper bounded by $|O|$. The clients in $O$ which will be served by $t_2$ are the clients that are closest to $t_2$; let the cost of serving these clients by $t_2$ be $Y$.

Let $r$ be a guess on the maximum service cost of any client served by $t_1$. We now formulate the problem of determining the best set $S' \subseteq S \setminus \{t_1\}$ as a knapsack problem.

$(i)$ For every $s \in S$, let $X(s, r) \subseteq X(s)$ be the set of clients served by $s$ which are at a distance at most $r$ from $t_1$.

$(ii)$ The cost of serving clients in $X(s, r)$ by $t_1$ is $\sum_{j \in X(s,r)} d(j, t_1)$. Let $v(s, r)$ represent the cost saved by closing $s$. Thus, $v(s, r) = \sum_{j \in X(s)} d(j, s) + f - \sum_{j \in X(s,r)} d(j, t_1)$.

$(iii)$ If a set $S' \subset S \setminus \{t_1\}$ of facilities is closed then the number of clients served by $t_1$ is $|X(t_1, r)| + \sum_{s \in S'} |X(s, r)|$ and the requirement that this is at most the capacity of $t_1$ gives the first knapsack constraint, i.e., $\sum_{s \in S'} |X(s,r)| \leq u(t_1) - |X(t_1,r)|$. Recall that $|X(t_1,r)| = 0$ if $t_1 \notin S$.

$(iv)$ The number of outliers generated by closing facilities in $S'$ is $(|X(t_1)| - |X(t_1,r)|) + \sum_{s \in S'}(|X(s)| - |X(s,r)|)$ and the requirement that this quantity is at most our guess $k$ gives the second knapsack constraint, i.e., $\sum_{s \in S'}(|X(s)| - |X(s,r)|) \leq k - |X(t_1)| + |X(t_1,r)|$.

$(v)$ If $t_1 = t_2 = t$ (say) then once we have guessed $k$ we can view $u(t_1)$ as equal to $u(t) - k$. Note that this will only affect the right-hand side of the constraint in $(iii)$.

The value of a set $S' \subseteq S \setminus \{t_1\}$ is $\sum_{s \in S'} v(s,r)$. We find the set $S'$ of maximum value, which satisfies the constraints in $(iii)$ and $(iv)$ using a knapsack procedure, detailed below. This value is the cost saved by closing facilities in $S'$. If it exceeds $Y$ plus the facility cost of $t_1$ and $t_2$, then doing the operation improves the cost of our solution.

In the knapsack problem we formulate, we have an object corresponding to each $s \in S \setminus \{t_1\}$. The weight, volume and profit of the object corresponding to $s$ are $|X(s,r)|$, $(|X(s)| - |X(s,r)|)$ and $v(s,r)$ respectively. We wish to pick a set of objects of total weight at most $u(t_1) - |X(t_1,r)|$, total volume at most $k - |X(t_1)| + |X(t_1,r)|$ and having maximum total profit. Note that this can be done in time $O(|S| \cdot u(t_1) \cdot k)$ which is $O(n^2 L)$. Since there are at most $L$ choices for $k$ and at most $n$ choices for $r$, the running time of this procedure is $O(n^3 L^2)$. $\square$

The algorithm stops when neither of the two operations improves the cost of the solution. Note that if a facility in $S$ does not serve any client we close that facility. The solution $S$ at the end of the algorithm is a locally optimum solution. The number of steps of the algorithm can be made polynomial by making a small sacrifice on the quality of the approximation as in Section 3.4.1.

## 3.6 Bounding the Cost of the Locally Optimum Solution

Recall that $S \subseteq F$ denotes the locally optimal solution, and $S^* \subseteq F$ denotes any feasible solution for a given instance of CFLO. $O$ and $O^*$ represents the set of outliers in the solutions $S$ and $S^*$, respectively. Note that, it is no loss of generality to assume $|O| = |O^*| = L$. We will assume $O \cap O^* = \phi$. This is without loss of generality as $S$ is also a locally optimum solution for an instance with clients $X \setminus (O \cap O^*)$ and number of outliers $L - |O^* \cap O|$.

For a facility $s \in S$, $X(s)$ represents the set of clients served by $s$ in the locally optimum solution and for a facility, $t \in S^*$, $X^*(t)$ represents the set of clients served by $t$ in the feasible solution $S^*$. Additionally, $\sigma : (X \setminus O) \rightarrow S$ is the assignment of clients in $S$ and $\sigma^* : (X \setminus O^*) \rightarrow S^*$ is the assignment of clients in $S^*$.

### 3.6.1 Bounding the Service Cost

The add operation can be used to bound the service cost of the solution as stated in Lemma 3.8.

**Lemma 3.8.** $C_s(S) \leq C_s(S^*) + C_f(S^*) = C(S^*)$.

**Proof.** For every facility $t \in S^*$, we add $t$ to $S$ if it is not in $S$ and if it reduces the cost of the solution. Let $\kappa$ be an arbitrary one-one and onto mapping from $O$ to $O^*$; $\kappa$ exists since $|O| = |O^*|$.

Consider an add($t$) operation. For every client $j$ served by $t$ in $S^*$, assign $j$ to $t$ if $j$ was served in $S$. The change in cost is $d(j,t) - d(j,\sigma(j))$. If $j \in O$, make $\kappa(j)$ an outlier and assign $j$ to $t$. The change in cost is $d(j,t) - d(\kappa(j), \sigma(\kappa(j)))$. Note that, all the capacities are respected because we assign only those clients to $t$ which are assigned to it in the feasible solution. Since adding $t \in S^*$ does not improve the cost of the solution, we obtain,

$$f + \sum_{j \in X^*(t)} d(j,t) - \sum_{j \in X^*(t) \setminus O} d(j,\sigma(j)) - \sum_{j \in X^*(t) \cap O} d(\kappa(j), \sigma(\kappa(j))) \geq 0.$$

Summing over all facilities $t$ in $S^*$ we obtain,

$$\sum_{t \in S^*} f + \sum_{t \in S^*} \sum_{j \in X^*(t)} d(j,t) - \sum_{t \in S^*} \sum_{j \in X^*(t) \setminus O} d(j,\sigma(j)) - \sum_{t \in S^*} \sum_{j \in X^*(t) \cap O} d(\kappa(j), \sigma(\kappa(j))) \geq 0.$$

$$\sum_{t\in S^*} f + \sum_{j\in X\setminus O^*} d(j,\sigma^*(j)) - \sum_{j\in X\setminus(O\cup O^*)} d(j,\sigma(j)) - \sum_{j\in O} d(\kappa(j),\sigma(\kappa(j))) \geq 0.$$

Since $\kappa : O \to O^*$ is a bijection $\sum_{j\in O} d(\kappa(j),\sigma(\kappa(j))) = \sum_{j\in O^*} d(j,\sigma(j))$ and hence

$$\sum_{t\in S^*} f + \sum_{j\in X\setminus O^*} d(j,\sigma^*(j)) - \sum_{j\in X\setminus(O\cup O^*)} d(j,\sigma(j)) - \sum_{j\in O^*} d(j,\sigma(j)) \geq 0.$$

$$\sum_{t\in S^*} f + \sum_{j\in X\setminus O^*} d(j,\sigma^*(j)) - \sum_{j\in X\setminus O} d(j,\sigma(j)) \geq 0.$$

or

$$\sum_{j\in X\setminus O} d(j,\sigma(j)) \leq \sum_{t\in S^*} f + \sum_{j\in X\setminus O^*} d(j,\sigma^*(j)).$$

□

### 3.6.2 Bounding the Facility Opening Cost

We can use the same approach for bounding the number of facilities in $S$ as we followed in the case of capacitated facility location. However, when we close a subset $S' \subseteq S$ of facilities, some clients in $X(S')$ might be outliers in $S^*$ (i.e., they belong to $O^*$), and we will not be able to assign these clients to the facility we open. We get around this by letting these clients be outliers in the new solution. Since the number of outliers needs to remain bounded by $L$, we serve some clients in $O$ by opening another facility from $S^*$. To ensure facilities are not opened too often, we will carefully choose a mapping from clients in $O^*$ to $O$.

#### 3.6.2.1 Modifying $H$ to Handle Outliers

We modify the graph $G$ constructed in Section 3.4.2 by introducing two vertices $o, o^*$, which correspond to outliers in $O$ and $O^*$, respectively. If client $j$ is in $O$ then we add edge $(o, j)$ to $G$ and the edge $(j, o^*)$ if $j$ is in $O^*$; these edges are not assigned any cost.

Once again, we decompose edges of $G$ into cycles and maximal paths and ignore all cycles from further consideration.

For a facility, $s \in S$, consider the maximal paths from $s$ to $o^*$ in this decomposition and let $O^*(s) \subseteq O^*$ be the set of clients preceding $o^*$ on these paths. Similarly, let $O(t) \subseteq O$ be the set of clients following $o$ on the maximal paths from $o$ to $t \in S^*$. Refer Figure 3.2.

The decomposition of edges of $G$ into maximal paths gives us paths from a facility $s \in S$ to a facility $t \in S^*$. The number of such paths starting from $s \in S$ is $\max\{0, |X(s)| - |X^*(s)| - |O^*(s)|\}$. Exactly $\max\{0, |X^*(t)| - |X(t)| - |O(t)|\}$ of such paths end at $t \in S^*$. Motivated by this observation we formulate a transhipment problem which has a supply of $\max\{0, |X(s)| - |O^*(s)|\}$ at node $s \in S$ and a demand of $\max\{0, |X^*(t)| - |O(t)|\}$ at node $t \in S^*$. Note that nodes in $S$ may have zero supply, and nodes in $S^*$ zero demand. The cost of shipping one unit of flow from $s \in S$ to $t \in S^*$ is $d(s,t)$. As before, it is no loss of generality to assume that in any solution to the transhipment problem for $s \in S \cap S^*$, $\min(|X(s)| - |O^*(s)|, |X^*(s)| - |O(s)|)$ flow will be shipped from the supply node $s$ to the demand node $s$ at zero cost. Let $x$ be an optimum solution to the transhipment problem.

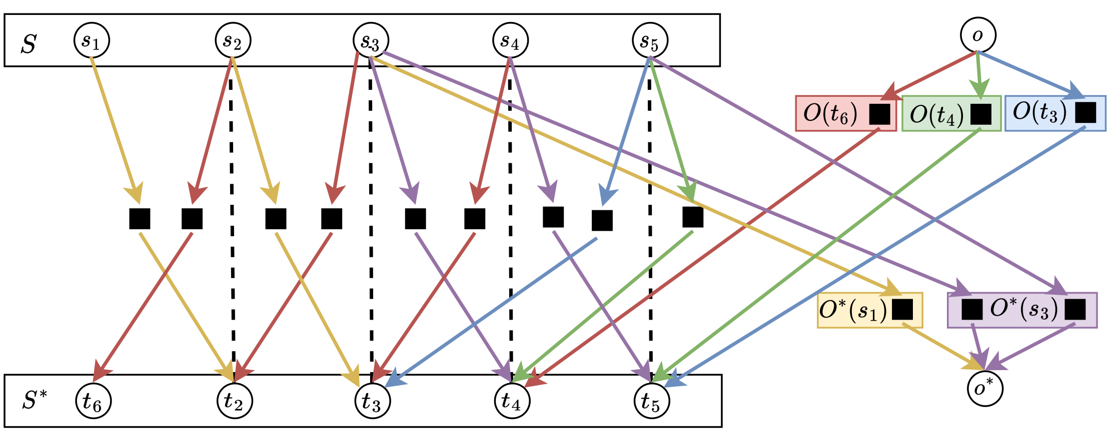



Figure 3.2: An illustration of $o$, $o^*$, $O(t)$ and $O^*(s)$ in graph $G$. Circles and squares represent the facilities and the clients, respectively. For illustration, two vertices are included for each facility in $S \cap S^*$ with a dashed edge between them. Edges of a colour trace a maximal path in the decomposition that either ends at $o^*$ or begins at $o$.

**Lemma 3.9.**

$$\sum_{s\in S,t\in S^*} x(s,t)d(s,t)+\sum_{t\in S^*}\sum_{j\in O(t)} d(j,t)+\sum_{s\in S}\sum_{j\in O^*(s)} d(s,j)\le C_s(S)+C_s(S^*).$$

**Proof.** The decomposition of edges of $G$ into maximal paths gives us 4 kinds of paths:

$(i)$ paths from a facility $s\in S$ to a facility $t\in S^*$. The cost of sending one unit of flow along such a path is $d(s,t)$, which is, at most, the sum of the costs of edges on the path.

$(ii)$ paths from $o$ to a facility $t\in S^*$. Let $j\in O(t)$ be the client following $o$ on this path. Then the total cost of edges on this path - excluding edge $(o,j)$ - is at least $d(j,t)$ and hence the total cost of edges on all such paths - excluding edges incident from $o$ - is at least $\sum_{t\in S^*}\sum_{j\in O(t)} d(j,t)$.

$(iii)$ paths from a facility $s\in S$ to $o^*$. Let $j\in O^*(s)$ be the client preceding $o^*$ on this path. Then the total cost of edges on this path - excluding edge $(j,o^*)$ - is at least $d(s,j)$ and hence the total cost of edges on all such paths - excluding edges incident to $o^*$ - is at least $\sum_{s\in S}\sum_{j\in O^*(s)} d(s,j)$.

$(iv)$ paths from $o$ to $o^*$ which we ignore.

Since the total cost of edges on all paths is at most $C_s(S)+C_s(S^*)$ the lemma follows. □

We use the optimum solution $x$ of the transhipment problem to construct an undirected exchange graph $H=(V',E')$ as in Section 3.4.2. $x, H$ have the following properties.

1. $H$ is acyclic.

2. If $s\in S\cap S^*$, then $H$ has an edge between the two vertices in $V'$ corresponding to $s$, and at least one of these vertices is a leaf of $H$.

3. $\forall s\in S$, $\sum_{t\in S^*} x(s,t)=|X(s)|-|O^*(s)|$.

4. $\forall t\in S^*$, $\sum_{s\in S} x(s,t)=|X^*(t)|-|O(t)|$.

As in Section 3.4.2, we root each component of $H$ at a facility in $S^*$ if one exists. Unlike Section 3.4.2, we might have some isolated facilities in $S$ in the forest $H$; these are the facilities with zero supply. Let $s$ be an isolated facility in $S$ and $t \in S^*$ be the root of some tree in $H$. We add edge $(s, t)$ to $H$ and assign $x(s, t) = 0$.

#### 3.6.2.2 Constructing $\kappa : O^* \to O$

Let $\mathcal{L}$ be the facilities in $S$ that are leaves in $H$. Let $F^* \subseteq S^*$ be the facilities that are the parent of some facility in $\mathcal{L}$. For $t \in S^*$, let $C_{\mathcal{L}}(t)$ be the children of $t$ in $\mathcal{L}$.

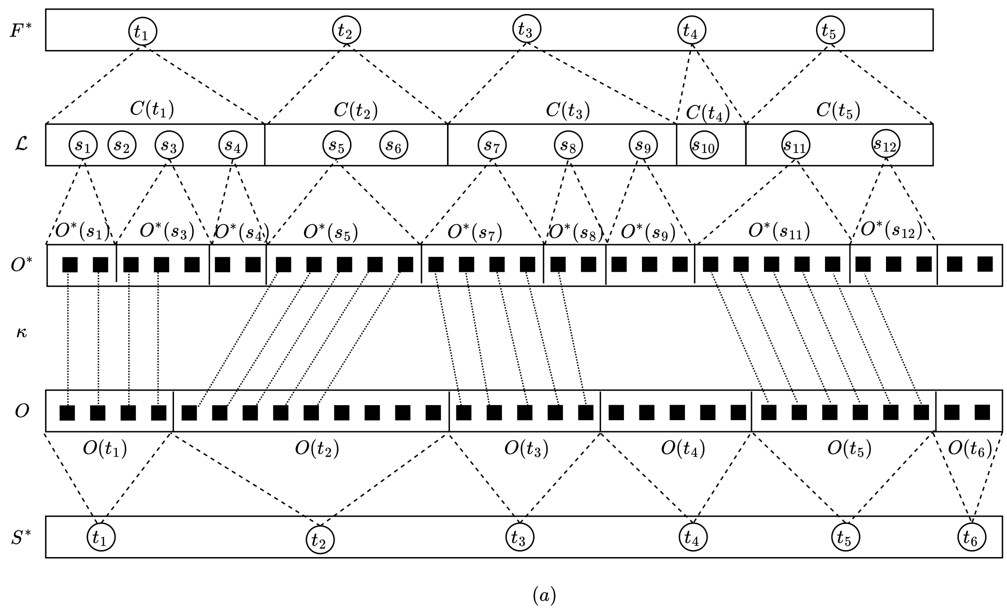
As in Section 3.4, to bound the number of facilities in $\mathcal{L}$ we can write inequalities for operations which open a facility $t \in F^*$ and close facilities in $C_{\mathcal{L}}(t)$. However, note that $t$ may not have sufficient capacity to serve all clients served by facilities in $C_{\mathcal{L}}(t)$ since some of these clients could correspond to outliers in $S^*$. To overcome this difficulty, we define a bijection, $\kappa$, from $O^*$ to $O$, which helps us identify a facility in $S^*$ to open to serve some outliers in $S$ to account for outliers in $S^*$ served by $C_{\mathcal{L}}(t)$.

Let $S^* = \{t_1, t_2, \ldots, t_{|S^*|}\}$. We construct the mapping in 2 steps using Algorithm 3.1 (Refer Figure 3.3).

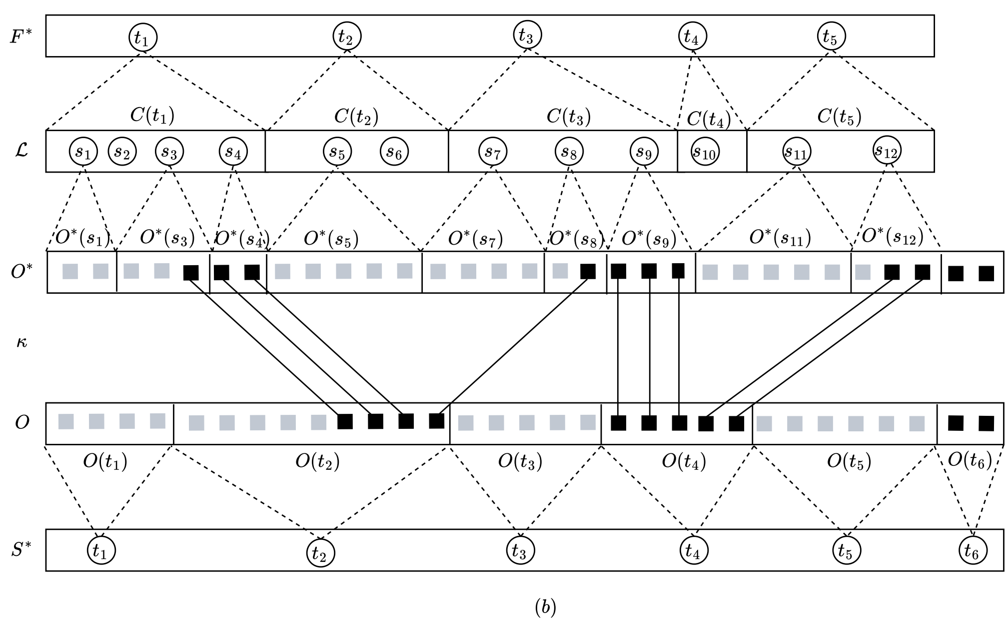
**Step 1** : We consider facilities of $S^*$ in increasing order of the indices and let $t_i$ be the facility under consideration. If $t_i \in F^*$ then consider the facilities in $C_{\mathcal{L}}(t_i)$. For a facility $s \in C_{\mathcal{L}}(t_i)$ we map clients in $O^*(s)$ to clients in $O(t_i)$. If this is not possible since no unmapped clients are remaining in $O(t_i)$, we save the remaining clients in $O^*(s)$ in an array $A$; such clients shall be mapped in Step 2. It is important that we consider all clients in $O^*(s)$ before moving on to the next facility in $C_{\mathcal{L}}(t_i)$. After this step, for all $t_i \in F^*$, either all clients in $O(t_i)$ or all clients in $\bigcup_{s \in C_{\mathcal{L}}(t_i)} O^*(s)$ are assigned under the mapping $\kappa$.

**Step 2:** We consider facilities in $S^*$ in the same order as in Step 1. If $t_i$ is the facility under consideration and a client in $O(t_i)$ is unmapped, we map a client saved in the array $A$ to such a client in $O(t_i)$.

The mapping constructed at the end of the above steps is extended to a bijection from $O^*$ to $O$ by assigning the unmapped clients arbitrarily.




$(a)$




$(b)$

Figure 3.3: $(a)$ An illustration of Step 1 of mapping $\kappa$. Circles and squares represent the facilities and the clients, respectively. Sets $F^*$ and $\mathcal{L}$ correspond to the tree in Figure 3.1. $(b)$ An illustration of Step 2 of mapping $\kappa$. Circles and squares represent the facilities and the clients, respectively. Sets $F^*$ and $\mathcal{L}$ correspond to the tree in Figure 3.1.

**Algorithm 3.1:** Mapping $\kappa$

```
Output :κ : O* → O
for j = 1 to |O*| do
    κ(j) = Null
end
// Step 1
count = 0
for i = 1 to |S*| do
    for s ∈ C_L(t_i) do
        for j ∈ O*(s) do
            if ∃ j' ∈ O(t_i) and κ^{-1}(j') == Null then
                κ(j) = j'
            else
                A[count++] = j // A[] stores clients not
                    mapped in Step 1
            end
        end
    end
end
// Step 2
count = 0
for i = 1 to |S*| do
    if ∃ j' ∈ O(t_i) and κ^{-1}(j') == Null then
        κ(A[count++]) = j
    end
end
return κ
```

#### 3.6.2.3 Constructing the Bipartite Graph, $Q$

We construct a bipartite graph $Q$, with vertex sets $U = \{u_i, 1 \le i \le |S^*|\}$ and $W = \{w_i, 1 \le i \le |S^*|\}$; vertices $u_i, w_i$ correspond to facility $t_i \in S^*$ (see Figure 3.4). For a facility $s \in \mathcal{L}$, we add an edge $(u_i, w_j)$ to $Q$ if $t_i$ is the parent of $s$ in $H$ and some client in $O^*(s)$ is mapped to a client in $O(t_j)$ ( Figure 3.3 and Figure 3.4). The edge is given a label $s$, i.e., $L(u_i, w_j) = \{s\}$. All edges with the same endpoints are combined into one edge and assigned a label which is the union of labels on these edges.

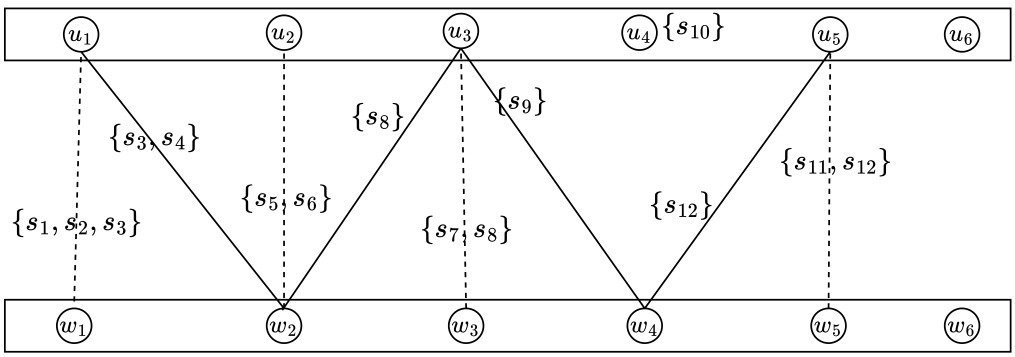



Figure 3.4: The graph $Q$ corresponding to the mapping in Figure 3.3. $\mathcal{L}_0 = \{s_2, s_6, s_{10}\}$, $\mathcal{L}_1 = \{s_1, s_4, s_5, s_7, s_9, s_{12}\}$, $\mathcal{L}_{>1} = \{s_3, s_8, s_{11}\}$. Dashed lines represent Type 1 edges, and solid lines represent Type 2 edges. The following multiSwap operations are performed: multiSwap($t_1, t_1, \{s_1, s_2\}$), multiSwap($t_1, t_2, \{s_4\}$), multiSwap($t_2, t_2, \{s_5, s_6\}$), multiSwap($t_3, t_3, \{s_7\}$), multiSwap($t_3, t_4, \{s_9\}$), multiSwap($t_4, t_4, \{s_{10}\}$) and, multiSwap($t_5, t_5, \{s_{11}\}$).

Edges of $Q$ that arose due to the mapping in Step 1 are called *type 1* edges, and those that arose due to the mapping in Step 2 are called *type 2* edges. Note that,

P1: The edge $(u_i, w_j), 1 \le i, j \le |S^*|$ is type 1 if $i = j$ and type 2 if $i \neq j$.

P2: For $1 \le i \le S^*$, at least one of $u_i, w_i$ has no type 2 edge incident on it.

P3: At most one type 1 edge is incident to any vertex in $Q$.

**Lemma 3.10.** *$Q$ is a forest.*

**Proof.** Two edges $(u_i, w_j)$ and $(u_k, w_l)$ are *crossing* if $i < k$ and $l < j$. We begin with a couple of claims.

**Claim 3.11.** *No pair of type 2 edges are crossing.*

**Proof.** For contradiction assume that $Q$ has edges $(u_i, w_j)$ and $(u_k, w_l)$ where $i < k$ and $l < j$. The existence of type 2 edges incident at $u_i, u_k$ implies that after step 1 not all clients in $\cup_{s\in C_{\mathcal{L}}(t_i)}O^*(s), \cup_{s\in C_{\mathcal{L}}(t_k)}O^*(s)$ are mapped by $\kappa$. Since $i < k$, $t_i$ is considered before $t_k$ in step 1 and unmapped clients in $\cup_{s\in C_{\mathcal{L}}(t_i)}O^*(s)$ appear before unmapped clients in $\cup_{s\in C_{\mathcal{L}}(t_k)}O^*(s)$ in array $A$.

In step 2, since $l < j$, $t_l$ is considered before $t_j$. In Lines 18-19 of Algorithm 3.1, unmapped clients in $\cup_{s\in C_{\mathcal{L}}(t_i)}O^*(s)$ are mapped to unmapped clients in $O(t_l)$ until either all clients in $\cup_{s\in C_{\mathcal{L}}(t_i)}O^*(s)$ or all clients in $O(t_l)$ are mapped. In the former case, $Q$ cannot contain $(u_i, w_j)$, while in the latter case, it cannot contain edge $(u_k, w_l)$. $\square$

**Claim 3.12.** *Type 2 edges form a forest in $Q$.*

**Proof.** For contradiction assume that $C$ is a cycle formed by type 2 edges and let $(u_{i_1}, w_{j_1})$ and $(w_{j_1}, u_{i_2})$ be adjacent edges on $C$. It is no loss of generality to assume $i_1 < i_2$. Let $(u_{i_2}, w_{j_2})$ be the next edge on $C$. Since it does not cross edge $(u_{i_1}, w_{j_1})$, it follows that $j_2 > j_1$. Continuing this argument, we observe that the vertices of $U$ and $W$ on the cycle are monotonically increasing sequences. This yields a contradiction. $\square$

To argue that $Q$ is a forest, we start with the forest of type 2 edges in $Q$ and add type 1 edges one by one, maintaining the invariant that the graph is a forest. Let $Q_i$ be the subgraph at Step $i$, and suppose we add the type 1 edge $(u_j, w_j)$. If this creates a cycle, then there must exist a path in $Q_i$ between $u_j$ and $w_j$. This implies both these vertices have type 2 edges incident at them, which contradicts property **P2**. $\square$

#### 3.6.2.4 Bounding the Number of Facilities

Let $\mathcal{L}_0$ be the facilities in $\mathcal{L}$ that do not label any edge, $\mathcal{L}_1$ the facilities that label exactly one edge, and $\mathcal{L}_{>1}$ the facilities that label more than one edge in $Q$. We first bound the number of facilities in the set $\mathcal{L}_{>1}$.

**Lemma 3.13.** $|\mathcal{L}_{>1}| \leq |S^*|$.

**Proof.** Let $s \in \mathcal{L}_{>1}$ be a facility that labels multiple edges in $Q$. If $s \in C_{\mathcal{L}}(t_i)$, then these edges have $u_i$ as a common endpoint.

Suppose $s$ labels the type 1 edge $(u_i, w_i)$. The other edges incident to $u_i$ are type 2, and by property **P2**, $w_i$ does not have any type 2 edge incident to it. We associate $s$ with $w_i \in W$.

Now assume $s$ does not label any type 1 edge. Let $(u_i, w_j), (u_i, w_{j+1}), \ldots, (u_i, w_k)$ be edges incident at $u_i$ which are labeled $s$. We associate $s$ with $w_j$. If $w_j$ is associated with another $s' \in \mathcal{L}_{>1}$, then there exists $l$ such that edge $(u_l, w_j)$ has a label $s'$. By our argument above $l \neq j$ and hence $(u_l, w_j)$ is type 2. By interchanging roles of $l, i$, we can assume $l > i$. The edge $(u_i, w_{j+1})$ which is type 2 crosses another type 2 edge $(u_l, w_j)$ yielding a contradiction.

We conclude that a node in $W$ is associated with at most one facility in $\mathcal{L}_{>1}$. Since $|W| = |S^*|$ the lemma follows.

□

We now bound $|\mathcal{L} \setminus \mathcal{L}_{>1}|$. Note that for $s \in \mathcal{L}_0$, $O^*(s) = \phi$. Consider a facility $s \in \mathcal{L}_0$ and let $t_i \in S^*$ be its parent in $H$. We add the label $s$ to edge $(u_i, w_i)$ and thus a facility which was earlier in $\mathcal{L}_0$ is now included in $\mathcal{L}_1$ (see Figure 3.4).

To bound the number of facilities in $\mathcal{L}_1$ we perform a multiSwap operation for some edges in $Q$. In particular, for an edge $(u_i, w_j)$ in $Q$, if $L(u_i, w_j) \cap \mathcal{L}_1 \neq \phi$, we write the inequality corresponding to multiSwap($t_i, t_j, S'$) (see Figure 3.4) where $S' = (L(u_i, w_j) \cap \mathcal{L}_1)$. In writing this inequality we assume a certain reassignment of the clients. Recall $S'$ are the facilities we close and let $s \in S'$.

**T1** Clients in $X(s)$ which are not on a maximal path from $s$ to $o^*$ are assigned to $t_i$ while the remaining clients in $X(s)$ are made outliers.

**T2** Clients in $O$ which are mapped to clients in $O^*(s)$ are served by $t_j$.

We now argue that the reassignment of clients does not violate capacities. Note that,

$(i)$ In multiSwap($t_i, t_j, S'$), the total number of clients assigned to facility $t_i$ is $\sum_{s \in S'}(|X(s)| - |O^*(s)|) = \sum_{s \in S'} x(s, t_i) \leq \sum_{s \in S} x(s, t_i) \leq |X^*(t_i)| - |O(t_i)| \leq u(t_i)$ and these inequalities follow from the properties of $x$. If $t_i \in S$ then an additional $|X(t_i)| - |O^*(t_i)| = x(t_i, t_i)$ clients are assigned to $t_i$. The total number of clients assigned to $t_i$ is still bounded by $\sum_{s \in S} x(s, t_i)$.

$(ii)$ The total number of clients assigned to $t_j$ is $\sum_{s \in S'} |O^*(s)|$ which by our mapping $\kappa$ and choice of $S'$ is at most $|O(t_j)|$. This, in turn, is at most $u(t_j)$. If $t_j \in S$ then the total number of clients assigned to $t_j$ is $|X(t_j)| + |O(t_j)|$ and this is at most $u(t_j)$. This follows from the fact that the number of maximal paths that end at $t_j$ in the decomposition of graph $G$ is at most $u(t_j) - |X(t_j)|$ and this is only more than the number of maximal paths from $o$ to $t_j$ which is $|O(t_j)|$.

$(iii)$ If $t_i = t_j$ then from the above argument the total number of clients assigned to $t_i$ is at most $|X^*(t_i)| - |O(t_i)|$ which together with the observation that at most $|O(t_j)| = |O(t_i)|$ outliers are assigned to $t_j$, implies that the capacity of $t_i$ is not violated.

**Lemma 3.14.** *The total increase in service cost due to the multiSwap operations performed is at most $C_s(S) + C_s(S^*)$.*

**Proof.** We perform a multiSwap operation for some edges of the forest $Q$. The total increase in service cost due to assignments of type **T1** for all these operations is bounded by $\sum_{t \in F^*} \sum_{s \in C_{\mathcal{L}}(t)} x(s,t)d(s,t) \leq \sum_{t \in S^*} \sum_{s \in S} x(s,t)d(s,t)$. The total change in service cost due to assignments of type **T2** is bounded by $\sum_{t \in S^*} \sum_{j \in O(t)} d(t,j)$. By Lemma 3.9, it follows that the total increase in service costs due to **T1**, **T2** is at most $C_s(S) + C_s(S^*)$. $\square$

Note that every facility in $\mathcal{L} \setminus \mathcal{L}_{>1}$ is closed in one of the multiSwap operations defined above. Since none of these multiSwap operations leads to an improvement in the total cost, $f \cdot |\mathcal{L} \setminus \mathcal{L}_{>1}|$ is at most the total cost of facilities opened and the total increase in service cost in these multiSwap operations.

The forest $Q$ has $2|S^*|$ vertices. However, $u_i$ has an edge incident to it only if $t_i \in F^*$ and hence the number of edges in $Q$ is at most $|F^*| + |S^*|$ edges. Of these edges exactly $|F^*|$ edges are type 1 and hence at most $|S^*|$ are type 2.

For a type 1 edge $(u_i, w_i)$, we write an inequality for the operation multiSwap($t_i, t_i$). Hence the number of facilities opened due to multiSwap operations corresponding to type 1 edges is at most $|F^*|$. The number of facilities opened due to multiSwap operation corresponding to type 2 edges is at most $2|S^*|$. Therefore, the total number of facilities opened over all mutliSwap operations is at most $3|S^*|$.

This together with the Lemma 3.14 implies that,

$$f \cdot |\mathcal{L} \setminus \mathcal{L}_{>1}| \leq 3f|S^*| + C_s(S) + C_s(S^*). \tag{3.5}$$

By Lemma 3.13, $f \cdot |\mathcal{L}_{>1}| \leq f \cdot |S^*|$ and by Lemma 3.6, $f \cdot |S \setminus \mathcal{L}| \leq f \cdot |S^*|$.

Combining, with inequality (3.5), we get,

$$C_f(S) = f \cdot |S| \leq 5C_f(S^*) + C_s(S) + C_s(S^*). \tag{3.6}$$

### 3.6.3 Putting Things Together

We are now ready to combine the bounds on the service and facility costs of the locally optimum solution, $S$ using the scaling idea from Section 3.4.

After scaling the facility costs by $\gamma$, Lemma 3.8 can be written as

$$C_s(S) \leq C_s(S^*) + \gamma C_f(S^*) \tag{3.7}$$

and inequality (3.6) corresponds to

$$\gamma C_f(S) \leq 2C_s(S^*) + 6\gamma C_f(S^*). \tag{3.8}$$

Thus,

$$C(S) = C_s(S) + C_f(S) \leq (1 + \frac{2}{\gamma})C_s(S^*) + (\gamma + 6)C_f(S^*).$$

For $\gamma = \frac{\sqrt{33}-5}{2}$, we get,

$$C(S) \leq (\frac{\sqrt{33}+7}{2})(C_s(S^*) + C_f(S^*)) = (\frac{\sqrt{33}+7}{2})C(S^*),$$

which implies a $(6.373 + \epsilon)$ approximation when $S^*$ is an optimum solution for the given instance of CFL.

## 3.7 A Bad Example for FLO with Non-Uniform Facility Costs

In this section, we present a challenging example for the (uncapacitated) facility location problem with outliers problem, where facility opening costs are non-uniform. This example demonstrates that any local search algorithm with a bounded locality gap must have an operation that requires solving an instance of FLO itself. We then modify the example to show that even with a constant factor violation allowed for outliers, escaping the unbounded locality gap requires the same operation.

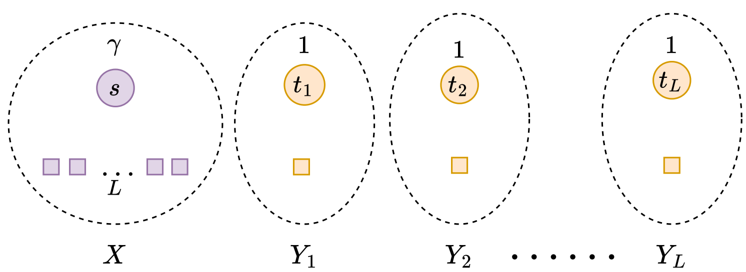



Figure 3.5: A bad example for FLO with non-uniform opening costs. Dashed boundaries represent the disjoint sets, circles represent facilities and squares represent clients.

Consider the following instance of facility location with outliers and non-uniform facility opening costs (see Figure 3.5).

- The client set and the facility set are partitioned into $L+1$ disjoint sets, $X, Y_1, Y_2, \ldots, Y_L$, and the distance between any two sets is very large.
- $X$ has one facility $s$ with opening cost $\gamma$ and $L$ clients at zero distance from $s$.
- $Y_\ell, 1 \leq \ell \leq L$ has one facility $t_\ell$ with facility opening cost $1$ and one client co-located with $t_\ell$.

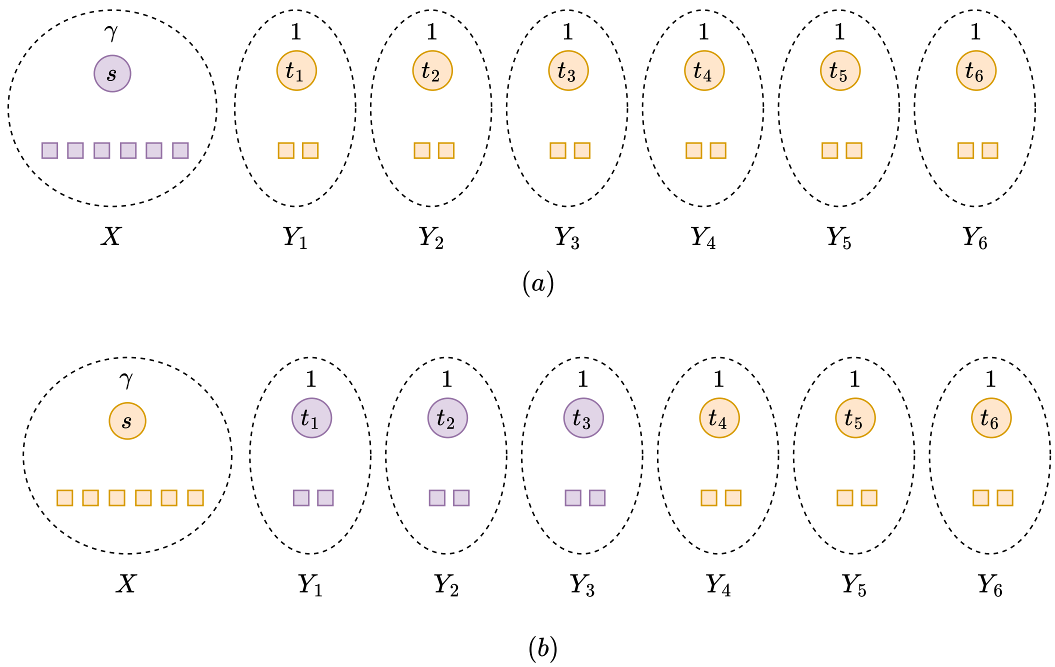



Figure 3.6: Dashed boundaries represent the disjoint sets, circles represent facilities and squares represent clients. Let $L = 6$ and $c = 2$. $(a)$ A bad example for FLO with non-uniform opening costs and violation in outliers. $(b)$ We need to open at least 3 facilities even with $c = 2$ factor violation in outliers to close $s$.

The solution $S = \{s\}$ is locally optimum. The $L$ clients co-located with $s$ are served by $s$ while the remaining $L$ clients are outliers in $S$. The cost of this solution is $\gamma$. Consider the solution $S^* = \{t_1, t_2, \ldots, t_L\}$ which serves the clients in $Y_1, Y_2, \ldots, Y_L$ and leaves the clients in $X$ unserved. The cost of solution $S^*$ is $L$. Hence, $cost(S)/cost(S^*) = \gamma/L$, and this ratio can be made arbitrarily large.

The only way to improve the cost of $S$ is to close $s$. If we close $s$ we need to open all facilities $t_1, t_2, \ldots, t_L$ to maintain the number of outliers. This implies that any local search algorithm would need an operation that closes a facility in the solution $S$ and opens a set of facilities $S' \subseteq F \setminus S$. However, an algorithm for finding the best such operation is the same as solving the facility location problem with outliers.

Next, we will show that even if we allow constant factor violation in outliers, we need an operation that requires solving a facility location with outliers instance (see Figure 3.5). Suppose we are allowed to violate outliers by a constant factor $c$. Modify the instance such that $Y_\ell, 1 \leq \ell \leq L$ has $c$ clients co-located with $t_\ell$ instead of $1$. The solution $S = \{s\}$ is locally optimum. The $L$ clients co-located with $s$ are served by $s$ while the remaining $cL$ clients are outliers in $S$. The cost of this solution is $\gamma$. Consider the solution $S^* = \{t_1, t_2, \ldots, t_L\}$ which serves the clients in $Y_1, Y_2, \ldots, Y_L$ and leaves the clients in $X$ unserved, that is, at most $L$ outliers. The cost of solution $S^*$ is $L$. Hence $\frac{cost(S)}{cost(S^*)} = \frac{\gamma}{L}$ which can be made arbitrarily large. The only way to improve the cost of $S$ is to close $s$. If we close $s$ we need to open at least $\frac{L}{c}$ facilities from $t_1, t_2, \ldots, t_L$ to maintain the number of outliers to be less than or equal to $cL$. This is the same as finding the best operation that closes a facility in the solution $S$ and opens a set of facilities $S' \subseteq F \setminus S$, which requires solving the facility location problem with outliers.

# Chapter 4

# Tri-Criteria for CFLO with Uniform Capacities[1]

## 4.1 Introduction

In this chapter, we relax the assumption of uniform facility costs and study CFLO with uniform capacities as defined in Definition 5.1. Recall that $(\mathcal{P}, d)$ denotes a metric space where $\mathcal{P}$ is a finite sets of points and $d : \mathcal{P} \times \mathcal{P} \to \mathbb{R}^+$ is a distance function satisfying triangle inequality and symmetry.

**Definition 4.1** (Capacitated Facility Location with Outliers and Uniform Capacities)**.** *We are given a set, $X \subseteq \mathcal{P}$, of $n$ clients and a set, $F \subseteq \mathcal{P}$ of $m$ facility locations. A facility $i \in F$ has a facility opening cost $f_i$ and a capacity $u \in \mathbb{N}$. We are also given a bound, $L \leq n$, on the number of outliers permitted in a feasible solution. The objective is to find,*

- *a set $S \subseteq F$ of facilities,*
- *a set $X' \subseteq X$ of outliers of size at most $L$,*
- *an assignment $\sigma : (X \setminus X') \to S$ respecting the capacities, i.e., for each facility $i \in S$, $|\sigma^{-1}(i)| \leq u$*

[1] The results presented in this chapter appeared in Dabas et al. (2022; 2025)

*such that the cost,* $\sum_{i \in S} f_i + \sum_{j \in X \setminus X'} d(j, \sigma(j))$, *is minimized.*

In Chapter 3, we presented a challenging example to demonstrate that achieving a constant-factor approximation for handling non-uniform opening costs can be difficult using local search techniques, even when we allow constant factor violation in outliers and capacities. On the other hand, both the CFL (even with uniform capacities) and FLO problems are known to have unbounded integrality gaps with respect to standard linear programming relaxations, as shown by Shmoys et al. (1997) and Charikar et al. (2001), respectively. Thus obtaining even a bi-criteria solution seems difficult using these techniques.

We make some progress on the problem by obtaining a tri-criteria solution that violates both capacities and outliers a little. Our solution is obtained by rounding a solution to the standard LP relaxation. In particular, we present a constant-factor approximation that incurs only small violations—specifically, a factor of $(1+\epsilon)$—in both capacities and outliers, as stated in Theorem 1.3. Thus, our result represents the best achievable outcome using the employed technique. Furthermore, the tri-criteria approach could be useful in the future for eliminating violations in capacities, outliers, or both.

**Theorem 1.3.** *There is a polynomial time algorithm that approximates capacitated facility location problem with outliers and uniform capacities within a constant factor* $(O(1/\epsilon^2))$ *violating the capacities and outliers by a factor of at most* $(1+\epsilon)$*, for a given constant* $\epsilon > 0$.

**High Level Idea:** After solving the standard linear program for the problem, our rounding algorithm proceeds in three steps. In Step 1, we identify a subset of clients that will serve as outliers in our solution. We incur at most a $(1+\epsilon)$ factor loss in both outliers and capacities during this process. In Step 2, we modify the technique used by Grover et al. (2022) for CFL (without outliers) to open the facilities integrally while preserving the extent to which the remaining clients are serviced from step 1. In Step 3, we solve minimum cost flow with outliers problem (see Lemma 1.7) to obtain the integral assignments for the remaining clients (some more clients may become outliers in the process). In the rest of the chapter, we focus only on the first two steps.

### 4.1.1 Organisation of the Chapter

The remainder of the chapter is organized as follows. The result for CFLO with uniform capacities is presented in Section 4.2, where we begin by describing the integer programming formulation of the problem. Details of Steps 1 and 2 are provided in Sections 4.2.1 and 4.2.2, respectively. The running time of the algorithm is discussed in Section 4.2.3.

## 4.2 Capacitated Facility Location with Outliers and Uniform Capacities

To achieve our result, we will be rounding the solution to the standard LP. Before diving into the details of the algorithm, we begin by first formulating the problem as an integer program (IP) as follows:

$$\text{Min } CostCFLO(x, y, z) = \sum_{j \in X} \sum_{i \in F} d(i, j) x_{ij} + \sum_{i \in F} f_i y_i$$

$$
\begin{aligned}
subject\ to \quad & \sum_{i \in F} x_{ij} + z_j \geq 1 && \forall\, j \in X && (4.1)\\
& \sum_{j \in X} x_{ij} \leq u\, y_i && \forall\, i \in F && (4.2)\\
& x_{ij} \leq y_i && \forall\, i \in F,\ j \in X && (4.3)\\
& \sum_{j \in X} z_j \leq L && && (4.4)\\
& x_{ij}, y_i, z_j \in \{0, 1\} && \forall\, i \in F,\ j \in X && (4.5)
\end{aligned}
$$

In this formulation, $y_i$ indicates whether facility $i$ is open, $z_j$ indicates whether client $j$ is an outlier, and $x_{ij}$ denotes whether client $j$ is served by facility $i$. Constraints 4.1 and 4.3 ensure that each client is either assigned to an open facility or designated as an outlier. Constraints 4.2 and 4.4 impose bounds on facility capacities and the total number of outliers, respectively.

Next, we relax the integer constraints, allowing $x_{ij}$, $y_i$, and $z_j$ to take values in the continuous range $[0, 1]$. This results in the LP relaxation. We solve the LP to obtain an optimal solution $\rho^* = \langle x^*, y^*, z^* \rangle$. For any solution $\rho = \langle x, y, z \rangle$ to LP, we define $cost(\rho)$ as the cost of the solution.

### 4.2.1 Identifying a Subset of Outliers (Step 1)

In this subsection, we discuss Step 1 of our algorithm where we identify a set of clients that will be treated as outliers in our solution. The idea is to make a client an outlier if it is an outlier to a large extent in the LP optimal solution, $\rho^*$. For a given $0 < \epsilon < 1/2$, we divide our client set $X$ into two sets,

$(i)$ $X_o = \{j : z_j^* \geq 1 - \epsilon; \forall j \in X\}$ and,

$(ii)$ $X_r \leftarrow X \setminus X_o$.

We next create a solution, $\hat{\rho} = \langle \hat{x}, \hat{y}, \hat{z} \rangle$ in which all the clients in $X_o$ are made outliers to the full extent. The assignments of the remaining clients ($X_r$) and the facility openings remain the same. Formally,

$(i)$ for $j \in X_o$, set $\hat{z}_j = 1$ and $\hat{x}_{ij} = 0 \; \forall i \in F$,

$(ii)$ for $j \in X_r$, set $\hat{z}_j = z_j^*$ and $\hat{x}_{ij} = x_{ij}^* \; \forall i \in F$ and,

$(iii)$ $\forall i \in F$, set $\hat{y}_i = y_i^*$.

Note that, $\sum_{j \in X} \hat{z}_j \leq (\frac{1}{1-\epsilon}) \sum_{j \in X} z_j^* \leq (1 + 2\epsilon)t$ for $\epsilon \leq 1/2$. Also, $cost(\hat{\rho})$ is bounded by $cost(\rho^*)$.

### 4.2.2 Obtaining Integrally Open Solution (Step 2)

In this subsection, we focus on Step 2 of our algorithm, that is, obtaining an *integrally open solution*. A solution $\langle x, y, z \rangle$ is said to be an integrally open solution if each facility is either fully opened or fully closed, that is, $\forall i \in F$, $y_i$ is $0$ or $1$. We obtain an integrally open solution for the problem instance with the reduced client set $X_r$ of clients preserving the extent to which the clients in $X_r$ are served in step $1$. Recall that the clients in our reduced client set ($X_r$) need not be fully served; however, $\sum_{i \in F} \hat{x}_{ij} > \epsilon, \forall j \in X_r$.

To obtain an integrally open solution, we first sparsify the problem instance using clustering techniques discussed in Section 4.2.2.1. The sparsified clusters are then categorized into two types, each of which is handled separately in Section 4.2.2.2 and Section 4.2.2.3, respectively.

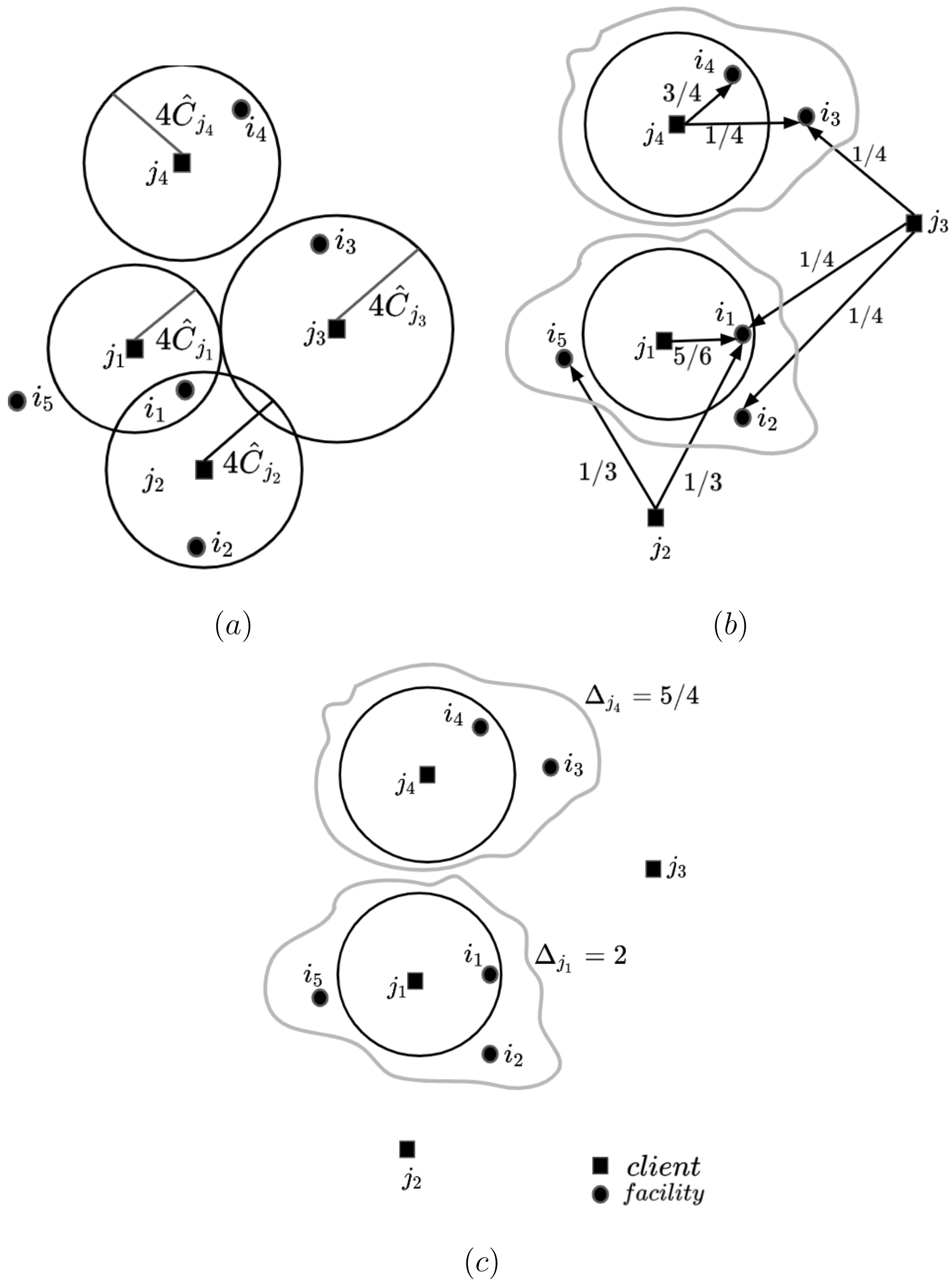



Figure 4.1: $(a)$ Balls around the clients in $X_r$. $(b)$ Reduced set of clients $X' = \{j_1, j_4\}$, clusters $(\mathcal{F})$ : $\mathcal{F}_{j_1} = \{i_1, i_2, i_5\}$, $\mathcal{F}_{j_4} = \{i_3, i_4\}$ and assignment by LP solution. $(c)$ $\Delta_{j_1} = \sum_{j \in X_r} (\hat{x}_{i_1 j} + \hat{x}_{i_2 j} + \hat{x}_{i_5 j})$, $\Delta_{j_4} = \sum_{j \in X_r} (\hat{x}_{i_3 j} + \hat{x}_{i_4 j})$.

#### 4.2.2.1 Sparsification and Clustering

We sparsify the problem instance by removing some clients from the client set $X_r$ such that clients that are retained are far from each other and clients that are removed are close to one of the retained clients. This is done using the standard filtering and clustering technique of Lin and Vitter (1992) and used in several previous works (2011; 2012; 2015a; 2018a).

**Algorithm 4.1:** Algorithm for Sparsification and Clustering.

**Input** : A fractional solution $\hat{\rho}$, a parameter $2 \leq \ell \leq 1/\epsilon$.

**Output** : Set of cluster centers ($X'$) and clusters ($\mathcal{F}$).

$\forall j \in X_r$, set $\hat{\mathcal{C}}_j = \frac{\sum_{i \in F} \hat{x}_{ij} d(i,j)}{\sum_{i \in F} \hat{x}_{ij}}$ `// Average connection cost`

$X' \leftarrow \emptyset, X_{temp} \leftarrow X$

**while** $X_{temp}$ *is not empty* **do**

  Pick $j \in X_{temp}$ with minimum $\ell \hat{\mathcal{C}}_j$ value

  Add $j$ to $X'$ and set $rep(j) = j$

  **for** $k \in X_{temp}$ **do**

    **if** $d(j,k) \leq 2\ell \max\{\hat{\mathcal{C}}_j, \hat{\mathcal{C}}_k\}$ **then**

      remove $k$ from $X_{temp}$

      set $rep(k) = j$

    **end**

  **end**

**end**

**for** $j' \in X'$ **do**

  set $\mathcal{F}_{j'} = \{i \in F : j'$ is nearest to $i$ amongst all $j' \in X'\}$

**end**

For $j \in X_r$, let $\hat{\mathcal{C}}_j$ denote the average connection cost of $j$ in $\hat{\rho}$, i.e., $\hat{\mathcal{C}}_j = \frac{\sum_{i \in F} \hat{x}_{ij} d(i,j)}{\sum_{i \in F} \hat{x}_{ij}}$. Clients in $X_r$ are considered in the non-decreasing order of $\ell \hat{\mathcal{C}}_j$, breaking the ties arbitrarily, where $\ell \geq 2$ is a tunable parameter. For a client $j$ at hand, remove any client $k : d(j,k) \leq 2\ell \max\{\hat{\mathcal{C}}_j, \hat{\mathcal{C}}_k\}$; $j$ is then called as the representative of $k$ (i.e., $rep(k) = j$).

Repeat the process with the remaining clients. Let $X'$ be the set of clients remaining after all the clients in $X_r$ have been considered.

Clusters, of facilities, are formed around the clients in $X'$ by assigning a facility to the cluster of the nearest client in $X'$, i.e. if, for $j' \in X'$, $\mathcal{F}_{j'}$ denotes the cluster centered at $j'$, then a facility $i$ belongs to $\mathcal{F}_{j'}$ if and only if $j'$ is the closest client in $X'$ to $i$ (breaking the ties arbitrarily). The clients in $X'$ are called the cluster centers. Refer to Figure 4.1 and Algorithm 4.1.

**Lemma 4.1.** $(i)$ *Any two cluster centers $j', k'$ in $X'$ satisfy: $d(j', k') > 2\ell \max\{\hat{\mathcal{C}}_{j'}, \hat{\mathcal{C}}_{k'}\}$,*

$(ii)$ *let $\mathcal{B}_{j'}$ be the set of facilities within a distance $\ell\hat{\mathcal{C}}_{j'}$ of $j'$, i.e., $\mathcal{B}_{j'} = \{i \in F\colon d(i, j') \le \ell\hat{\mathcal{C}}_{j'}\}$ then $\mathcal{B}_{j'} \subseteq \mathcal{F}_{j'}$,*

$(iii)$ *total extent up to which facilities in $\mathcal{B}_{j'}$, for any $j' \in X'$, are opened under $\hat{\rho}$ is at least $\epsilon/2$, and*

$(iv)$ *total extent up to which facilities in $\mathcal{F}_{j'}$, for any $j' \in X'$, are opened under $\hat{\rho}$ is at least $(1 - \frac{1}{\ell})^2$.*

**Proof.** $(i)$ Follows from step 7 of Algorithm 4.1.

$(ii)$ Let $i$ be a facility that belongs to $\mathcal{B}_{j'}$ but not $\mathcal{F}_{j'}$. Note that, $d(i, j') \le \ell\hat{\mathcal{C}}_{j'}$ by definition of $\mathcal{B}_{j'}$. Since $i \notin \mathcal{F}_{j'}$, there exist a cluster center $k'$ which is nearer to $i$ as compared to $j'$, that is, $d(i, k') \le d(i, j') \le \ell\hat{\mathcal{C}}_{j'}$. By triangle inequality, $d(j', k') \le d(j', i) + d(i, k') \le 2\ell\hat{\mathcal{C}}_{j'}$ which is contradiction because for any two cluster centers $j'$ and $k'$, $d(j', k') > 2\ell \max\{\hat{\mathcal{C}}_{j'}, \hat{\mathcal{C}}_{k'}\}$ (from $(i)$).

$(iii)$ Note that, in the weighted average $\ell\hat{\mathcal{C}}_{j'}$, less than $1/\ell$ of the total weight can be given to values more than $\ell\hat{\mathcal{C}}_{j'}$. That is,

$$\sum_{i: d(i,j') > \ell\hat{\mathcal{C}}_{j'}} \hat{x}_{ij'} < \frac{1}{\ell} \sum_{i \in F} \hat{x}_{ij'}.$$

This also implies,

$$\sum_{i: d(i,j') \le \ell\hat{\mathcal{C}}_{j'}} \hat{x}_{ij'} \ge \Big(1 - \frac{1}{\ell}\Big) \sum_{i \in F} \hat{x}_{ij'}.$$

From Constraints 4.3 of LP and definition of $\mathcal{B}_{j'}$, we have,

$$\sum_{i \in \mathcal{B}_{j'}} \hat{y} \geq \sum_{i : d(i,j') \leq \ell \hat{\mathcal{C}}_{j'}} \hat{x}_{ij'} \geq \Big(1 - \frac{1}{\ell}\Big) \sum_{i \in F} \hat{x}_{ij'} \geq \frac{\epsilon}{2}.$$

where the last inequality follows because $1 - \frac{1}{\ell} \geq \frac{1}{2}$ for $\ell \geq 2$ and $\sum_{i \in F} \hat{x}_{ij'} \geq \epsilon$.

$(iv)$ The claim follows from $(ii)$ and $(iii)$. $\square$

For $j \in X_r$, $j' \in X'$, let $\phi(j,\ j')$ be the extent up to which $j$ is served by the facilities in $\mathcal{F}_{j'}$. Let $\Delta_{j'} = \sum_{j \in X_r} \phi(j,\ j'), \forall j' \in X'$.

A cluster is said to be $small$ if $\Delta_{j'} \leq u$; otherwise, it is called $big$. Let $\mathcal{X}_S = \{j' \in X' : \Delta_{j'} \leq u\}$ and $\mathcal{X}_B = X' \setminus \mathcal{X}_S$.

#### 4.2.2.2 Handling Small Clusters

In this subsection, we obtain integrally open solution for small clusters. For $j' \in \mathcal{X}_S$, we open the cheapest facility, say $i_{j'}$, in $\mathcal{B}_{j'}$ and close all the remaining facilities in the cluster. All the assignments coming into the cluster is transferred onto $i_{j'}$. Note that $\Delta_{j'} \leq u$ for all $j' \in \mathcal{X}_S$; therefore, there is no violation in the capacity. The following lemma bounds the service and facility opening costs.

**Lemma 4.2.** *The facilities in small clusters are opened integrally at $\frac{2}{\epsilon}$ factor loss in facility cost and $4(\ell + 1)$ factor loss in service cost.*

**Proof.**

- *Facility Cost*: Since the total opening in each cluster is $\geq \frac{\epsilon}{2}$, the cost of opening the cheapest facility is bounded by $(\frac{2}{\epsilon} \sum_{i \in \mathcal{F}_{j'}} f_i \hat{y}_i)$.

- *Service Cost*: Let $j \in X_r$, if $j$ lies in the ball of $j'$ then $\ell \hat{\mathcal{C}}_{j'} \leq 2\ell \hat{\mathcal{C}}_j$ (suppose if possible $\ell \hat{\mathcal{C}}_{j'} > 2\ell \hat{\mathcal{C}}_j$, then there exists a $k'$ such that $d(j, k') \leq 2\ell \hat{\mathcal{C}}_j$ and hence $d(j', k') \leq d(j', j) + d(j, k') \leq \ell \hat{\mathcal{C}}_{j'} + 2\ell \hat{\mathcal{C}}_j < 2\ell \hat{\mathcal{C}}_{j'}$ which is a contradiction). Let $i \in \mathcal{F}_{j'}$ be a facility in $\mathcal{F}_{j'}$ such that $\hat{x}_{ij} > 0$. Refer Figure 4.2. We have,

$$
\begin{aligned}
d(i_{j'}, j) &\le d(j', j) + d(i_{j'}, j') \\
&\quad \text{(using triangle inequality)} \\
&\le d(j', j) + \ell\hat{\mathcal{C}}_{j'} \\
&\le \max\{2d(j', j), d(j, j') + 2\ell\hat{\mathcal{C}}_j\} \\
&\quad \text{(since } \ell\hat{\mathcal{C}}_{j'} \le 2\ell\hat{\mathcal{C}}_j \text{ if } j \text{ lies in the ball of } j' \text{and } \le d(j', j) \text{ otherwise)} \\
&= \max\{2(d(j', i) + d(i, j)), (d(j', i) + d(i, j)) + 2\ell\hat{\mathcal{C}}_j\} \\
&\quad \text{(using triangle inequality)} \\
&\le \max\{2(d(k', i) + d(i, j)), (d(k', i) + d(i, j)) + 2\ell\hat{\mathcal{C}}_j\} \\
&\quad \text{(since } i \text{ is nearest to } j' \text{ in } X'\text{)} \\
&\le \max\{2(d(k', j) + d(j, i) + d(i, j)), (d(k', j) + d(j, i) + d(i, j)) + 2\ell\hat{\mathcal{C}}_j\} \\
&\quad \text{(using triangle inequality)} \\
&\le 4(\ell\hat{\mathcal{C}}_j + d(i, j))
\end{aligned}
$$

□

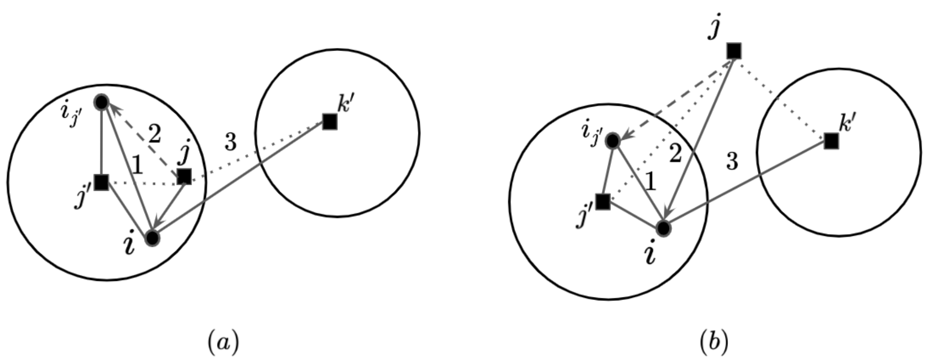



Figure 4.2: Bound of service cost for small clusters. 1: $d(i_{j'}, j) \le d(j', j) + d(i_{j'}, j')$, 2: $d(j', j) \le d(j', i) + d(i, j)$ and, 3: $d(k', i) \le d(k', j) + d(j, i)$.$(a)$ When $j$ lies in the ball of $j'$. $(b)$ When $j$ lies outside the ball of $j'$.

#### 4.2.2.3 Handling Big Clusters

In this subsection, we obtain integrally open solution for big clusters. To handle big clusters, we first move the demand $\Delta_{j'}$ to $j'$. The cost of moving $\Delta_{j'}$ to $j'$ can be bounded similarly to Lemma 4.3.

**Lemma 4.3** (Grover et al. (2022)). *For $j' \in X'$, the cost of moving $\Delta_{j'}$ to $j'$ is bounded by $2(\ell+1)cost(\hat{\rho})$.*

Let us fix a $j' \in \mathcal{X}_B$. We next solve the following LP to handle the big cluster centered at $j'$.

$$\text{Minimize } Cost_{CI}(w) = \sum_{i \in \mathcal{F}_{j'}} (f_i + ud(i,j'))w_i$$

$$subject\ to \quad u \sum_{i \in \mathcal{F}_{j'}} w_i \geq \Delta_{j'} \tag{4.6}$$

$$w_i \in [0,1] \tag{4.7}$$

Note that $w_i = \sum_{j \in X_r} \hat{x}_{ij}/u$ is a feasible solution with cost at most

$$\sum_{i \in \mathcal{F}_{j'}} \Big( f\hat{y}_i + \sum_{j \in X_r} \hat{x}_{ij}(d(i,j) + 2\ell\hat{\mathcal{C}}_j) \Big).$$

A solution to a cluster instance is called an *almost integral* if it has at most one fractionally opened facility. An almost integral solution $w'$ is obtained by arranging the fractionally opened facilities in $w$ in non-decreasing order of $f + d(i,j')u$ and greedily transferring the openings $w$ without increasing the cost of the solution. We obtain an integrally open solution $\hat{w}$ from the almost integral solution as follows: if the opening of the fractionally opened facility, if any, is $\leq \epsilon$, we close it else we open it at $(1/\epsilon)$-factor loss in (facility) cost. $\Delta_{j'}$ is distributed equally to the facilities opened in the cluster. Let $n'$ be the number of facilities opened in the cluster centered at $j'$. Note that, by constraint (4.6), $n' \geq \left\lfloor \frac{\Delta_{j'}}{u} \right\rfloor \geq \frac{\Delta_{j'}}{(1+\epsilon)u}$. Therefore, we incur a loss of at most $(1+\epsilon)$-factor in capacities and service cost.

By choosing $\ell = 2$ and summing over all small and big clusters, we obtain an integrally open solution that violates the capacities by a factor of $(1+\epsilon)$ and is of cost bounded by $O(1/\epsilon)$.

**Algorithm 4.2:** Algorithm for CFLO with uniform capacities.

**Input** : An instance $I$ of CFLO with uniform capacities, $0 < \epsilon < 1/2$

1 Solve LP.

2 Identify a subset of outliers such that violation in outliers $\leq (1+\epsilon)L$.

3 Make clusters on remaining clients.

4 Open facilities integrally in small clusters.

5 Open facilities integrally in big clusters such that violation in capacities $\leq (1+\epsilon)u$.

6 Solve min-cost flow with outliers to obtain integral assignments and identify remaining outliers.

### 4.2.3 Time Complexity of the Algorithm

Recall that $|X| = n$ and $|F| = m$. The algorithm involves solving a linear programming problem with $O(mn)$ variables, which takes $O((mn)^{2.16})$ time (Cohen et al. (2021)). Step 1 involves checking the value of a variable for each client, which takes $O(n)$ time. In Step 2, sparsification and clustering are performed in Algorithm 4.1, requiring $O(n^2 + nm)$ time. Handling sparse clusters takes $O(m)$ time while processing dense clusters requires $O(m \log m + n)$ time. Step 3 can be completed in $O(mn(m+n))$ time, given $O(m+n)$ nodes and $O(mn)$ edges in the min-cost flow instance. As a result, the overall time complexity of the algorithm is $O((mn)^{2.16})$.

# Chapter 5

# (3+$\epsilon$) - FPT Approximation for C$k$MO[1]

## 5.1 Introduction

In this chapter, we study Capacitated $k$-Median with Outliers (C$k$MO) problems. We start with a formal definition of C$k$MO problem. Recall that $(\mathcal{P}, d)$ denotes a metric space where $\mathcal{P}$ is a finite set of points and $d : \mathcal{P} \times \mathcal{P} \rightarrow \mathbb{R}^+$ is a distance function satisfying triangle inequality and symmetry.

**Definition 5.1** (Capacitated $k$-Median with Outliers (C$k$MO))**.** *We are given a set, $X \subseteq \mathcal{P}$, of $n$ clients and a set, $F \subseteq \mathcal{P}$ of $m$ facility locations. A facility $i \in F$ has a capacity $u(i) \in \mathbb{N}$. We are also given bounds, $k \leq m$ and $L \leq n$, on the number of facilities and outliers permitted in a feasible solution. The objective is to find,*

- *a set $\mathcal{F} \subseteq F$ of facilities of size at most $k$,*
- *a set $X' \subseteq X$ of outliers of size at most $L$,*
- *an assignment $\sigma : (X \setminus X') \rightarrow \mathcal{F}$ respecting the capacities, i.e., for each facility $i \in \mathcal{F}$, $|\sigma^{-1}(i)| \leq u(i)$*

*such that the cost, $\sum_{j \in X \setminus X'} d(j, \sigma(j))$, is minimized.*

[1]The results presented in this chapter appeared in Dabas et al. (2025)

Note that C$k$MO generalizes C$k$M, and the polynomial-time approximability of the latter itself remains open. Another direction of work in the field is to explore approaches beyond polynomial-time approximations, aiming for FPT approximations. Adamczyk et al. (2019) broke the barrier of constant-factor approximability in general metrics, by designing $(7+\epsilon)$ factor approximation for C$k$M that runs in time FPT in $k$ and $\epsilon$. This was later improved by Cohen-Addad and Li (2019) to a $(3+\epsilon)$-approximation, in time in FPT in $k$ and $\epsilon$. Inspired by this success of FPT approximations in the context of C$k$M, we study FPT approximation for C$k$MO and present Theorem 1.5.

**Theorem 1.5.** *[Informal] There exists a randomized approximation-preserving reduction from CkMO to CkM that runs in time FPT in $k$, the number of outliers and $\epsilon$, where the underlying metric space remains unchanged and $\epsilon > 0$ is a small constant.*

By plugging in the best-known approximations for the C$k$M, we obtain Corollary 1.6.

**Corollary 1.6.** *There exists a randomized algorithm that runs in time FPT in $k$, the number of outliers and $\epsilon$ where $\epsilon > 0$ is a small constant and returns a $(3+\epsilon)$ approximation with high probability.*

**Coresets.** Creating compact data representations that approximately maintain the cost of solutions—commonly referred to as *coresets*—has been a key area of research for many years. In the context of clustering, an $\alpha$-coreset is the weighted subset of points that preserves the clustering cost within $\alpha$-relative error w.r.t. all possible feasible solutions. Coresets for vanilla $k$-Median clustering have been extensively studied. We do not cover the detailed literature for $k$-Median here, but it can be found in Cohen-Addad et al. (2021).

Coresets for constrained clustering, specifically C$k$M, is a relatively new research direction. Cohen-Addad and Li (2019) designed an $\epsilon$-coreset of size poly$(k\epsilon^{-1}\log n)$, en route to their $(3+\epsilon)$-approximation for C$k$M in FPT time. Recently, there has been some work on the *fair* version of $k$-Median, where we impose upper- and lower-bounds on the number of points of each color in each cluster. This version is known to capture C$k$M with uniform capacities, and (2019; 2021; 2019) designed coresets for fair $k$-Median. All of these coreset results were recently improved and unified in a recent result by Braverman et al. (2022), who gave a general framework via uniform sampling. In an orthogonal direction, coresets for $k$MO have also been studied in the literature, with the most recent developments being the construction of $\epsilon$-coreset of size poly$((k+L)\epsilon^{-1}\log n)$ by Agrawal et al. (2023).

At first glance, it may appear that the capacity constraints are already powerful enough to model outliers, i.e., one can obtain coresets for C$k$MO via the aforementioned results for C$k$M, by reducing the $L$ outliers to $L$ extra clusters of unit capacity. However, it is not clear whether each solution to the resulting C$k$M instance can be mapped back to a solution to the original C$k$MO instance of (approximately) equal cost. Thus, we need to handle outliers more carefully in order to obtain coresets for C$k$MO.

**Parallel Independent Work.** In a recent independent work Jaisal and Kumar (2023), obtain similar results for constrained versions of $k$-MEDIAN/MEANS WITH OUTLIERS for various constraints, including capacities. However, their techniques appear quite different from ours and do not seem to yield coresets. As mentioned in Backurs et al. (2019), using coresets not only saves storage space but can also accelerate the computation time for clustering problems like fair clustering. Another advantage of having coresets is that once the coresets are formed, they can be repeatedly used as a proxy for the full dataset to compare the clustering performance under different fairness constraints or with different capacities.

### 5.1.1 Organisation of the Chapter

The rest of the chapter is organised as follows. In Section 5.2, we provide a high-level idea of our algorithm and compare it with existing related work. Section 5.3 presents a detailed discussion of the algorithm for C$k$MO and an overview of the analysis. The complete analysis is covered in Sections 5.4 and 5.5. We discuss the modifications required to account for the facility opening costs in Section 5.6.

## 5.2 High Level Idea and Comparison to Related Work

A starting point of our algorithm—like many other FPT approximations for $k$-Median and variants ((2019; 2019; 2023))—is the elegant "ring sampling" approach of Chen (2009). We start with a brief overview of this construction and subsequently discuss how this is used in our reduction framework to handle capacities and outliers.

In this construction, the idea is to do random sampling for "compressing" the given set of $n$ clients into a "small" number of *weighted clients* that is a good approximation of the original set[2]. Let us consider the simplest setting of $k$-Median without capacities or outliers as Chen (2009). We start from a crude (constant factor) approximation $\mathcal{F} \subseteq F$ for $k$-Median, and use it to partition the clients into *concentric rings* around the centers in $\mathcal{F}$. Here a ring is a subset of clients that are within a distance $r$ and $2r$ from some $i \in \mathcal{F}$, where we consider geometrically increasing radii $r$. Next, we take a uniform sample of clients from each ring of large enough size and assign appropriate weight to each sampled client. The resulting weighted sample has size $O(k \log n/\epsilon)^2)$. It can be shown that the *weighted clients* in the sample approximately preserves the cost of clustering w.r.t. any solution with high probability. This is shown using a Chernoff-Hoeffding type concentration bound, which uses the following: even though the distance of each weighted client in the sample is a random variable, all random variables lie in a bounded interval due to triangle inequality. Precisely, for any solution $\mathcal{F} \subseteq F$, and for any ring $R \subseteq X$ with radius $r$, the distances $\{d(j, \mathcal{F})\}_{j \in R}$ lie within an interval of length $2r$.

Since the size of the sample is relatively small, it is used in Cohen-Addad et al. (2019) for partial enumeration of a subset of approximate solutions for $k$-Median, which leads to the FPT running time. Agrawal et al. (2023) argue that in the context of $k$MO, by taking an even larger sample size (that now also depends on $L$), the concentration bound is robust enough to handle outliers. In particular, they show that for each ring, the concentration bound continues to hold *even after* ignoring the contribution of at most $L$ points that are farthest from the solution.

Cohen-Addad and Li (2019) extend the ring sampling approach to C$k$M, but the

[2]This notion can be formalized to define *coresets*. We omit a formal definition here.

analysis is more intricate due to the following reason. Consider a particular ring $R$ with center $c$, and a hypothetical solution $\mathcal{F} \subseteq F$ of size $k$. For each $c_i \in \mathcal{F}$, let $X_{c_i}$ denote the subset of clients from $R$ that are assigned to $c_i$. Since we take a uniform sample from each ring, in expectation, we maintain the relative proportions of $|X_{c_i}|$'s in the sample. Therefore, in expectation, the sample maintains the cost as well as capacity constraints. However, it is quite likely that we *over*- or *under-sample* clients from different $X_{c_i}$'s, and we need to bound the cost of a feasible assignment of the weighted sample. To this end, the authors Cohen-Addad and Li (2019) use an instance of the minimum cost flow problem. They use the ring-center $c$ to reroute flows between different over- and under-sampled $X_{c_i}$'s, and the cost of rerouting can be bounded using a generalization of the above argument involving triangle inequality.

**Our sampling and analysis.** Since we need to handle both capacities and outliers, we need to be even more careful in our analysis. In a solution for C$k$MO, unlike C$k$M, not all clients in a ring $R$ may be assigned to facilities – some can be outliers. In the presence of capacities, this issue cannot be handled by simply ignoring the contribution of $L$ farthest clients from $\mathcal{F}$, unlike in $k$MO. The optimal choice of weighted outliers from $R$, in fact, depends on the outcome of sampling outside $R$. Though, in expectation, we maintain relative proportions of $|X_{c_i}|$'s and of outliers in $R$, we may over- or under-sample clients from different $X_{c_i}$'s and over- or under-sample clients from the set of outliers. However, we are able to decouple the analysis of under/over-sampling in each $X_{c_i}$ from the analysis of under/over-sampling in the set of outliers in $R$. At this point, the error incurred by sampling from the subset of assigned clients can be bounded by essentially following the argument from Cohen-Addad and Li (2019). Next, we bound the error incurred while sampling from the set of outliers. Since we treat each ring independently in the analysis, the subset of outliers from $R$ is essentially arbitrary. To this end, we generalize the argument from Agrawal et al. (2023), and prove that rerouting cost in case of over/under sampling from the set of outliers is negligible for *any* set of weighted clients of weight at most $L$, not just for those that are farthest from $\mathcal{F}$. To bound the costs of rerouting, we use the minimum cost flow problem with outliers (see Lemma 1.7).

In conclusion, we show that the ring sampling approach can be used to construct a weighted sample $W$ of size $O\left(\frac{(kL\log n)^2}{\epsilon^3}\right)$ that approximately preserves the C$k$MO cost w.r.t. any feasible solution $\mathcal{F} \subseteq F$ of size at most $k$, with high probability (Lemma 5.2).

Next, we discuss how to use $W$ to reduce C$k$MO to C$k$M. We enumerate all possible subsets of $W' \subseteq W$ of total *weight* $L$ that corresponds to the set of outliers corresponding to an optimal solution $(F^*, \sigma^*)$ for the given instance. Using the bound on $|W|$, it can be easily argued that the number of such guesses is bounded by $f(k, L, \epsilon) \cdot |I_L|^{O(1)}$. For each such guess $W' \subseteq W$ for outliers, we obtain an instance of weighted C$k$M by deleting $W'$, and it can be easily converted to unweighted C$k$M by creating multiple co-located copies of each weighted client. For each such instance of C$k$M, we use a $\gamma$-approximation, and we return the minimum-cost solution found over all guesses. Using Lemma 5.2, we argue in Lemma 5.4 that in the iteration corresponding to the "correct" guess of $W'$, a $\gamma$-approximate solution for the C$k$M instance is, in fact, a $(\gamma + \epsilon)$-approximation for the original instance of C$k$MO. See Figure 5.1 for an overview of the algorithm.

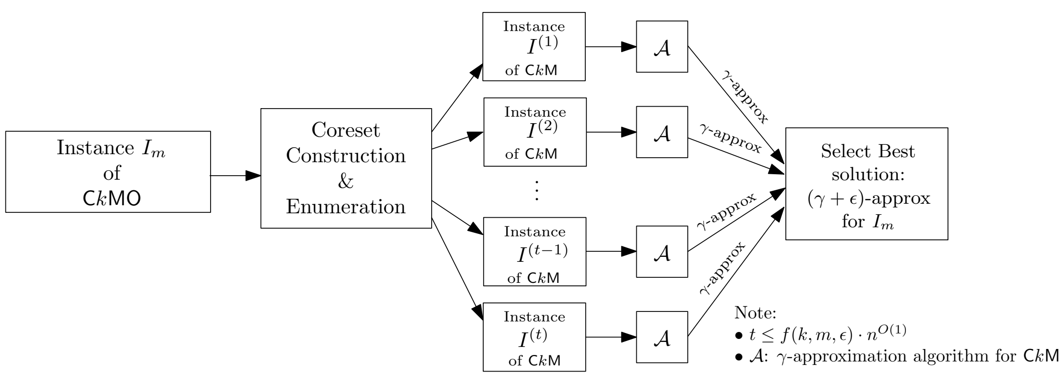



Figure 5.1: Overview of the Algorithm for C$k$MO.

## 5.3 Algorithm and Overview of Analysis

### 5.3.1 Preliminaries

Given an instance $I_L$ of C$k$MO and a set of facilities $\mathcal{F} \subseteq F$, we define $\mathsf{cost}_L(X, \mathcal{F})$ as the cost of optimal assignment with $L$ outliers. If the sum of the capacities of the facilities in $\mathcal{F}$ is less than $n - L$, then we call such a set $\mathcal{F}$ to be infeasible and define

$\mathsf{cost}_L(X, \mathcal{F}) = \infty$. Note that, for a given feasible $\mathcal{F}$, the set $X'$ of outliers and the assignment of clients to facilities can easily be determined by solving a minimum cost flow with outliers problem (see Lemma 1.7). Hence, $\mathsf{cost}_L(X, \mathcal{F})$ can be computed for a given $\mathcal{F} \subseteq F$.

In this chapter, we will often have non-negative integer weights on clients. Essentially, the weight on a client can be thought of as multiple co-located copies of a client. A formal definition follows.

**Definition 5.2.** *(WC$k$MO). The input is $I_L = ((P, d), W, F, k, u, L)$ where $W \subseteq X \times \mathbb{N}$ is the set of pairs $\{(j, w(j)) : j \in X\}$ and $w : X \to \mathbb{N}$ is a weight function. The objective now is to find a subset $\mathcal{F} \subseteq F$ of size at most $k$ and an assignment $\sigma : X \times \mathcal{F} \to \mathbb{N}$ such that:*

$(i)$ *For each $j \in X$, $\sum_{i \in \mathcal{F}} \sigma(j, i) \leq w(j)$,*

$(ii)$ *Total unassigned weight is at most L, i.e., $\sum_{j \in X} w(j) - \sum_{j \in X, i \in \mathcal{F}} \sigma(j, i) \leq L$,*

$(iii)$ *Assignment must respect capacity constraints for each $i \in \mathcal{F}$, i.e., $\sum\limits_{j \in X} \sigma(j, i) \leq u_i$, and*

$(iv)$ *$\sum_{j \in X, i \in \mathcal{F}} \sigma(j, i) d(j, i)$ is minimized.*

Similar to C$k$MO, given an instance of WC$k$MO and a set of facilities $\mathcal{F} \subseteq F$, we define $\mathsf{wcost}_L(W, \mathcal{F})$ as the cost of optimal assignment where the total unassigned weight is at most $L$. If the sum of the capacities of the facilities in $\mathcal{F}$ is less than $\sum_{j \in X} w(j) - L$, then we call such a set $\mathcal{F}$ to be infeasible and define $\mathsf{wcost}_L(X, \mathcal{F}) = \infty$. Just like C$k$MO, for a given feasible $\mathcal{F}$, the assignment of clients to facilities for WC$k$MO can also be easily determined by solving minimum cost flow problem with outliers. Hence, $\mathsf{wcost}_L(W, \mathcal{F})$ can be obtained for a given $\mathcal{F} \subseteq F$.

### 5.3.2 The Algorithm

Given an instance $I_L = ((P, d), X, F, k, u, L)$ of C$k$MO, we first create an instance $I_0^w = ((P, d), X, F \cup X, k + L)$ of $(k + L)$-Median (un-capacitated) where we have a facility co-located with every client in addition to the original set of facilities.

**Claim 5.1.** *Value of an optimal solution to $I_0^w$ is bounded by the value of an optimal solution to $I_L$, that is,* $\mathsf{OPT}(I_0^w) \leq \mathsf{OPT}(I_L)$.

**Proof.** Let $(\mathcal{F}, X', \sigma)$ be the optimal solution for instance $I_L$ with value $\mathsf{OPT}(I_L)$. A solution for instance $I_0^w$ can be created as follows: open a facility that is co-located with each outlier point in $X'$ along with set $\mathcal{F}$. Assign the outliers to the co-located facilities. Note that this can be done without incurring any additional cost. Remaining clients are served from where they were getting served in the solution of instance $I_L$ at the same cost. Hence, we have a feasible solution for the instance $I_0^w$ of cost no larger than $\mathsf{OPT}(I_L)$ and $\mathsf{OPT}(I_0^w)$ can only be lesser than this. Therefore the claim follows. □

Now, the instance $I_0^w$ is solved using any polynomial time algorithm (say Gowda et al. (2023)) to obtain a constant ($\zeta$) factor approximation. Let $\mathcal{F}_\zeta$ be the set of facilities opened by the algorithm. Note that $k_\zeta = |\mathcal{F}_\zeta| \leq k + L$ and

$$\mathsf{cost}_0(X, \mathcal{F}_\zeta) \leq \zeta \cdot \mathsf{OPT}(I_0^w) \leq \zeta \cdot \mathsf{OPT}(I_L) \tag{5.1}$$

For every facility $i \in \mathcal{F}_\zeta$, let $X_i \subseteq X$ be the clients assigned to $i$ by the algorithm. Note that the sets $X_i$ are disjoint. Let $\mathsf{Ball}(i, \mathcal{R}) \subseteq X_i$ denote the ball of radius $\mathcal{R}$ consisting of clients, in $X_i$, within distance $\mathcal{R}$ from $i$. Let $R = \frac{\mathsf{cost}_0(X, \mathcal{F}_\zeta)}{\zeta n}$, and $\psi = \lceil \log(\zeta n) \rceil$.

Henceforth, we fix $0 < \epsilon < 1$ to be a small enough constant that determines the quality of the approximation. For a facility $i \in \mathcal{F}_\zeta$ and set $X_i$, we further partition $X_i$ into smaller sets called as *rings* such that points in each ring have similar distances to $i$, i.e.,

$$X_{i,t} = \begin{cases} \mathsf{Ball}(i, R), & \text{if } t = 0. \\ \mathsf{Ball}(i, 2^t R) \setminus \mathsf{Ball}(i, 2^{t-1} R), & \text{if } 1 \leq t \leq \psi. \end{cases} \tag{5.2}$$

Let $s = \frac{a\zeta^2}{\epsilon^3}(L + k \ln n)$ where $a$ is a large enough constant. We sample weighted points $W^i$ from $X_i$ as follows: for every set $X_{i,t} \subseteq X_i$, if $|X_{i,t}| \leq s$ then for every client $j \in X_{i,t}$, add $(j, 1)$ to $W^i$. Else, sample $s$ random clients in $S_{i,t} \subseteq X_{i,t}$ without replacement. For every client $j \in S_{i,t}$, add $(j, \frac{|X_{i,t}|}{s})$ to $W^i$. See Figure 5.2.

Applying the above procedure on every cluster $i \in \mathcal{F}_\zeta$, we get $W = \cup_{i \in \mathcal{F}_\zeta} W^i$ as the weighted set of points. As the number of rings is $\log n$ and the number of points in each

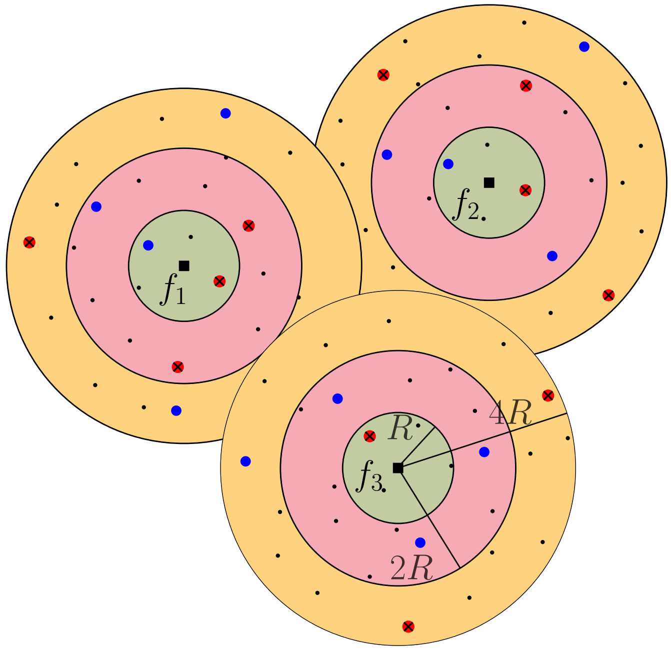



Figure 5.2: An illustration of the formation of rings and sampling from each ring. Black squares represent the cluster centers. Three rings of radius $R$, $2R$, $4R$ are represented in green, pink and orange color respectively. Dots/small circles represent the points in the ring. Small blue circles represent the non-outlier sampled points, whereas red points (marked crossed) represent the guessed points that will become outliers. We further guess the weight for each of these points that will be the outlier.

ring is at most $\frac{a\zeta^2}{\epsilon^3}(L+k\ln n)$, the total number of weighted points, $|W| = O\left(\frac{(kL\log n)^2}{\epsilon^3}\right)$ as $a$ and $\zeta$ are constants. Without loss of generality, we assume the weights are integral, by a slight modification of the construction, which can increase the number of points in $W$ by at most a factor of 2. See, Chen (2009) or Agrawal et al. (2023) for details.

Our key technical contribution is to show the following lemma (Lemma 5.2), which states that for all feasible sets $\mathcal{F} \subseteq F$, the cost of assignment of the weighted sample to $\mathcal{F}$, after excluding outliers with total weight $L$, is close to the cost of assigning original set of clients, after excluding $L$ outliers from $\mathcal{F}$. This is precisely the definition of $\epsilon$-coresets, which have been extensively studied, especially in the context of (uncapacitated) $k$-Median (see Cohen-Addad et al. (2021) and references therein). Thus, Lemma 5.2 implies the existence of small-sized coreset that handles both capacities and outlier constraints; however, unlike the coreset literature, in this work, our focus is not on optimizing the size of the coreset. The proof of the lemma is deferred to Section 5.4 and

Section 5.5 for ease of understanding.

**Lemma 5.2.** *For all feasible sets $\mathcal{F} \subseteq F$ of size $k$, $|\mathsf{wcost}_L(W, \mathcal{F}) - \mathsf{cost}_L(X, \mathcal{F})| \leq \epsilon \mathsf{cost}_L(X, \mathcal{F})$ with probability at least $1 - 1/n$.*

Assuming Lemma 5.2 holds, let us return to our algorithm to see how it can be used to reduce C$k$MO to C$k$M. Recall that $W$ is the set of weighted clients returned by our sampling algorithm and let $S$ be the set of clients (i.e., the set of first elements from each pair $(j, w(j)) \in W$, denoting a weighted client). For any subset $T \subseteq S$, let $w(T) = \sum_{j_x \in T} w(j_x)$. The algorithm then proceeds as follows: we iterate over each guess $T \subseteq S$ of size at most $L$. First, we check whether $w(T) \geq L$ – if not, we continue to the next guess. Now, suppose $w(T) \geq L$. Then, let us order the points in $T$ as $j_1, j_2, \ldots, j_{L'}$, where $L' = |T|$. For each $j_x \in T$, we guess an integer $0 < v(j_x) \leq \min\{L, w(j_x)\}$, such that $\sum_{j_x \in T} v(j_x) = L$. Note that the number of guesses is at most $L^L$. Now, fix one such guess $\mathbf{v} = (v(j_1), v(j_2), \ldots, v(j_L'))$. Now, we define a new weight function $w_{T,\mathbf{v}} : S \to \mathbb{N}$ as follows.

$$w_{T,\mathbf{v}}(j) = \begin{cases} w(j) & \text{if } j \notin T \\ w(j_x) - z(j_x) & \text{if } j = j_x \in T \end{cases}$$

Let $W_{T,\mathbf{v}} = \{(j, w_{T,\mathbf{v}}(j)) : j \in S\}$ denote the resulting set of weighted points, where the set of points is the same as that in $W$, but the weight function is $w_{T,\mathbf{v}}$ instead of $w$. Now, the algorithm uses a $\gamma$-approximation algorithm for the $\mathsf{WC}k\mathsf{M}$[3] instance $I_0^w = ((P, d), W_{T,\mathbf{v}}, F, k, L' := 0)$ to find an $\mathcal{F} \subseteq F$ of size at most $k$ and an assignment $\sigma : X \times \mathcal{F} \to \mathbb{N}$ satisfying the first three properties in Definition 5.2. Then, we use min cost flow with outliers to compute the optimal assignment cost from $X$ to $\mathcal{F}$ with $L$ outliers, and the corresponding assignment $\sigma$. Finally, the algorithm returns a solution $(\mathcal{F}^*, \sigma^*)$ of the minimum cost, over all guesses (i.e., guesses for $T$ as well as $\mathbf{v}$).

[3]Existing $\gamma$-approximation algorithms for unweighted C$k$M can be used to solve weighted instances by creating multiple co-located copies of each weighted client.

### 5.3.3 Analysis Overview

First, in Lemma 5.3, we analyze the running time of the algorithm, and then in Lemma 5.4, we analyze the approximation guarantee.

**Lemma 5.3.** *Let $\mathcal{T}(|I'|, k')$ denote the running time of the $\gamma$-approximation used to solve an instance $I'$ of* CAPACITATED $k'$-MEDIAN. *Then, the running time of our algorithm is upper bounded by $f(k, L, \epsilon) \cdot \mathcal{T}(|I_L|, k) \cdot |I_L|^{O(1)}$.*

**Proof.** Note that construction of the set $W$ takes polynomial time since we use $\zeta$-approximation for $k + L$-MEDIAN as a starting point. Next, we bound the total number of guesses tried by the algorithm. First, we guess a subset $T \subseteq W$ of size at most $L$, which takes time $\binom{|W|}{L} \leq \left(\frac{kL\log n}{\epsilon}\right)^{O(L)}$, which can be upper bounded by $\left(\frac{kL}{\epsilon}\right)^{O(L)} \cdot n^{O(1)}$ via a standard case analysis on whether $L \leq \frac{\log n}{\log\log n}$. Next, for each such guess $T$, the algorithm guesses the vector $\mathbf{v} = (v(p_1), v(p_2), \dots, v(p_{L'}))$. As remarked earlier, the number of such guesses is upper bounded by $L^L$. Finally, for each such guess, we use a $\gamma$-approximation for C$k$M, which takes time $\mathcal{T}(|I_L|, k)$, and for the solution $\mathcal{F} \subseteq F$ returned by the algorithm, we use min cost flow with outliers to compute the cost w.r.t. the original set $X$ which runs in polynomial time due to Lemma 1.7. Thus, the claimed bound on the running time follows. □

**Lemma 5.4.** *Let $\mathcal{F}^* \subseteq F$ be the set of at most $k$ facilities returned by the algorithm. Then, with probability at least $1 - 1/n$, it holds that for any feasible set $\mathcal{F} \subseteq F$ of size at most $k$,*

$$\mathsf{cost}_L(X, \mathcal{F}^*) \leq \gamma \cdot \frac{1+\epsilon}{1-\epsilon} \cdot \mathsf{cost}_L(X, \mathcal{F}).$$

**Proof.** The statement in Lemma 5.2 holds with probability at least $1 - 1/n$. We condition on this good event and show that the current lemma holds (with probability 1). Fix a feasible set $\mathcal{F} \subseteq F$ of size at most $k$, as in the statement of the lemma. Then, by Lemma 5.2, we have the following inequality:

$$(1-\epsilon) \cdot \mathsf{cost}_L(X, \mathcal{F}) \leq \mathsf{wcost}_L(W, \mathcal{F}) \leq (1+\epsilon) \cdot \mathsf{cost}_L(X, \mathcal{F}). \tag{5.3}$$

Now, consider the assignment $\sigma$ realizing $\mathsf{wcost}_L(W, \mathcal{F})$, and let $T \subseteq S$ be the set of clients $j$, such that $\sum_{i \in F} \sigma(j, i) < w(j)$. Note that $v(j) \coloneqq w(j) - \sum_{i \in \mathcal{F}} \sigma(j, i)$ is an integer for every $j \in S$. Let $\mathbf{v} = (v(j_1), v(j_2), \ldots, v(j'_L))$, where $T = \{j_1, j_2, \ldots, j_{L'}\}$ is indexed arbitrarily.

It is easy to verify that $(T, \mathbf{v})$ as defined here, satisfy the conditions of a "guess" as in the algorithm. Thus, consider the iteration of the algorithm corresponding to $T$ and $\mathbf{v}$. It follows that $\mathsf{wcost}_L(W, \mathcal{F}) = \mathsf{wcost}_0(W_{T,\mathbf{v}}, \mathcal{F})$. Let $\mathcal{F}' \subseteq F$ be the $\gamma$-approximation found for the $\mathsf{WC}k\mathsf{M}$ instance $I_0^w = ((P,d), W_{T,\mathbf{v}}, F, k, 0)$ in this iteration. It follows that,

$$\mathsf{wcost}_L(W, \mathcal{F}') = \mathsf{wcost}_0(W_{T,\mathbf{v}}, \mathcal{F}') \leq \gamma \cdot \mathsf{OPT}(I_L) \leq \gamma \cdot \mathsf{wcost}_0(W_{T,\mathbf{v}}, \mathcal{F}) \tag{5.4}$$

Here, $\mathsf{OPT}(I_L)$ denotes the cost of an optimal solution for the C$k$MO instance $I_L$, and since $\mathcal{F}$ is a feasible set of at most $k$ facilities, the inequality $\mathsf{OPT}(I_L) \leq \mathsf{wcost}_0(W_{T,\mathbf{v}}, \mathcal{F})$ follows.

Now, again from Lemma 5.2, we have the following inequality

$$(1-\epsilon) \cdot \mathsf{cost}_L(X, \mathcal{F}') \leq \mathsf{wcost}_L(W, \mathcal{F}') \leq (1+\epsilon) \cdot \mathsf{cost}_L(X, \mathcal{F}') \tag{5.5}$$

Finally, $\mathcal{F}^* \subseteq F$ is the best solution found over all iterations, it follows that

$$\begin{aligned}
\mathsf{cost}_L(X, \mathcal{F}^*) \leq \mathsf{cost}_L(X, \mathcal{F}') &\leq \frac{1}{1-\epsilon} \cdot \gamma \cdot \mathsf{wcost}_0(W_{T,\mathbf{v}}, \mathcal{F}) && \text{(From (5.4) and (5.5)} \\
&= \frac{1}{1-\epsilon} \cdot \gamma \cdot \mathsf{wcost}_L(W, \mathcal{F}) && \text{(By definition of } T, \mathbf{v}) \\
&\leq \gamma \cdot \frac{1+\epsilon}{1-\epsilon} \cdot \mathsf{cost}_L(X, \mathcal{F}). && \text{(From (5.3))}
\end{aligned}$$

□

Lemma 5.3 and Lemma 5.4 together yield a proof of the following theorem, which is the formal version of Theorem 1.5.

**Theorem 5.5.** *Given a $\gamma$-approximation for* Capacitated $k'$-Median *that runs in time $\mathcal{T}(|I'|, k')$ on an input instance $I'$, we give a randomized algorithm that runs in time $f(k, L, \epsilon) \cdot \mathcal{T}(|I_L|, k) \cdot |I_L|^{O(1)}$ for an instance $I_L$ of CkMO and achieves $(\gamma+\epsilon)$-approximation with high probability.*

To complete the proof of Theorem 5.5, it remains to prove Lemma 5.2. First, in Section 5.4, we prove Lemma 5.2 in a special case. Then, in Section 5.5, we show how this analysis can be extended to the more general case.

## 5.4 Single Ring Case

In this section, we will prove a version of Lemma 5.2 in the following special case that already illustrates most of the technical details, which will be later used to prove the more general case.

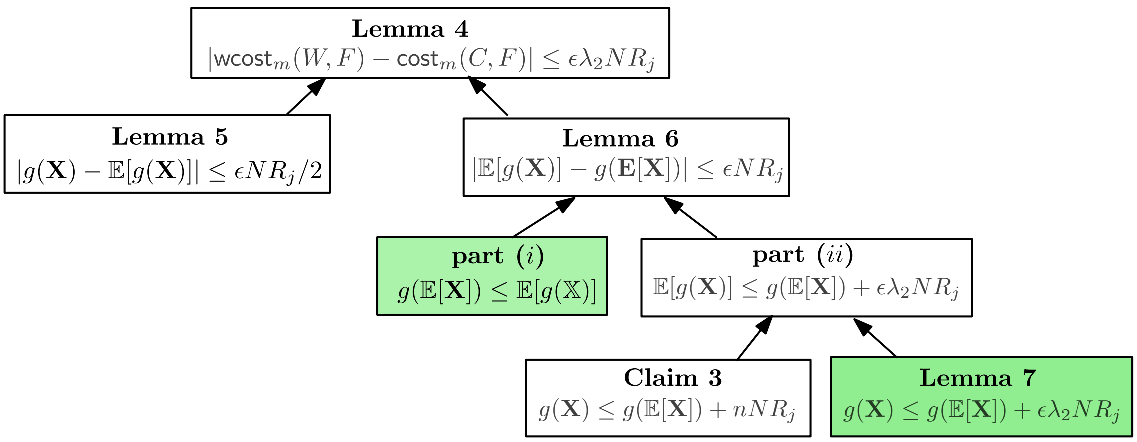



Figure 5.3: Relationship among different lemmas and claims that culminate in the proof of Lemma 5.6. The green-colored rectangles contain the parts of the proof that are significantly different from similar previous work in the literature.

Consider an arbitrary ring $X_{i,t}$, in a cluster $X_i$ centered at a facility $i \in \mathcal{F}_\zeta$, that will remain fixed throughout this section. We assume that the algorithm performs sampling only inside $X_{i,j}$; whereas all the clients in rings different from $X_{i,t}$ are all included into $W$ with weight 1. Note that this case can occur in the actual algorithm if all rings other than $X_{i,t}$ contain fewer than $s$ points. For simplicity of notation, let $R_t = 2^t R$ be the radius of this ring and let $X_t = X_{i,t}$ be the set of clients in the ring. Let $N = |X_t|$ be the number of clients in the ring. Let $W = \cup_i W^i$ be the weighted clients returned by our algorithm. In the remaining section, we will prove the Lemma 5.6. We refer the reader to Figure 5.3 for an overview of how different lemmas are related to each other to yield the proof of Lemma 5.6.

**Lemma 5.6.** *For any feasible $\mathcal{F} \subseteq F$ of size $k$, $|\mathsf{wcost}_L(W, \mathcal{F}) - \mathsf{cost}_L(X, \mathcal{F})| \leq \epsilon\lambda_2 N R_t$ with probability $1 - n^{-(k+\lambda_1)}$ where $\lambda_1$ and $\lambda_2$ are constants.*

As stated in the lemma, we fix a feasible set $\mathcal{F} \subseteq F$. For the rest of the section, we will assume $N > s = O\left(\frac{kL \log n}{\epsilon^3}\right)$, because otherwise the sampling does not change anything (recall that all clients outside $X_{i,t}$ already belong to $W$ with weight 1). For our analysis, we will define a random vector $\mathbf{X}$ and a function $g$ such that $g(\mathbf{X})$ and $\mathsf{wcost}_L(W, \mathcal{F})$ are identically distributed.

**Defining X:** a random vector $\mathbf{X} \in \mathbb{R}_+^N$ is defined as follows: for each coordinate pick value $N/s$ with probability $s/N$ and 0 otherwise such that $\mathbb{E}[\mathbf{X}] = \mathbf{1}$. One can show that with sufficient probability, this sampling scheme selects exactly $s$ points using the Claim 5.7.

**Claim 5.7** (Cohen-Addad and Li (2019)). *Let $b$ be a positive integer, and let $p \in (0, 1)$ such that $pb$ is an integer. The probability that Binomial($b, p$)= $pb$ is $\Omega(1/\sqrt{b})$.*

Setting $b = N$ and $p = s/N$ in Claim 5.7, it follows that $\mathbf{X}$ has exactly $N \cdot (s/N) = s$ non-zero entries with probability $\Omega(1/\sqrt{N})$. Conditioned on this event, then, $\mathbf{X}$ and $W$ are identically distributed, i.e., the $\mathbf{X}$ represents the outcome of our sampling algorithm. In the rest of the section, we analyze the unconditioned behavior of $\mathbf{X}$, and show that the desired concentration (as in Lemma 5.6) holds with high probability. Then, a standard argument shows that Lemma 5.6 also holds with high probability, even when we condition on the event that $\mathbf{X}$ has exactly $s$ non-zero entries.

**Defining Function $g$:** to define $g$, we first create an instance of min cost flow with outliers. Given a vector $\mathbf{v}$ of size $N$ where each entry in $\mathbf{v}$ corresponds to a client in $X_t$, a flow instance $\mathsf{FI}(\mathbf{v})$ is created as follows: every client $j \in X_t$ has $\mathbf{v}_j$ units of demand, every client in $X \setminus X_t$ has 1 unit of demand, cluster center $i$ has $N - \sum_{j \in X_t} \mathbf{v}_j$ (possibly negative[4]) demand. Every facility $i' \in \mathcal{F}$ has $u_{i'}$ units of supply. The number of outliers is $L$. For $\mathbf{v} \in \mathbb{R}_+^N$, let $g(\mathbf{v})$ denote the cost of optimal flow of $\mathsf{FI}(\mathbf{v})$. The flow instance we defined is feasible because the sum of demands is $|X| = n$ and the number of outliers is $L$, making the demand to be served to be $n - L$ which is feasible by assumption on $\mathcal{F}$.

Note that, $g(\mathbb{E}[\mathbf{X}]) = g(\mathbb{1})$ is exactly $\mathsf{cost}_L(X, \mathcal{F})$. Also, $g(\mathbf{X})$ and $\mathsf{wcost}_L(W, \mathcal{F})$ are identically distributed. Now, we will prove Lemma 5.6 in two steps: $(i)$ $g(\mathbf{X}) \approx \mathbb{E}(g(\mathbf{X}))$ with high probability (proven in Lemma 5.8) and $(ii)$ $\mathbb{E}(g(\mathbf{X})) \approx g(\mathbb{E}[\mathbf{X}])$ (proven in Lemma 5.10).

[4]Negative demand $d$ at a vertex $v$ requires that $d$ units of flow must enter $v$, whereas a positive demand requires that the specified units of demand must exit the vertex.

**Lemma 5.8.** $|g(\mathbf{X}) - \mathbb{E}[g(\mathbf{X})]| \leq \epsilon N R_t/2$ *with probability* $\geq 1 - n^{-(k+c)}$.

**Proof.** We will first show that $g(\mathbf{X})$ is $R_t$-Lipschitz with respect to the $\ell_1$ distance in $\mathbb{R}_+^N$ and then apply standard martingles tools to prove that $g(\mathbf{X})$ is concentrated around its mean. To prove $g(\mathbf{X})$ is $R_t$-Lipschitz, fix a client $j \in X_t$, and consider two vectors $\mathbf{v}, \mathbf{v}' \in \mathbb{R}_+^N$ with $\mathbf{v}' = \mathbf{v} + \delta \cdot \mathbb{1}_j$. Note that the flow instance $\mathbf{v}$ is the same as the flow instance of $\mathbf{v}'$ except the latter has $\delta$ more demand at client $j$ and $\delta$ less demand at cluster center $i$.

We will first construct a feasible flow for $FI(\mathbf{v}')$ from optimal flow $\phi$ of $FI(\mathbf{v})$ as follows: create a dummy facility $d_f$ with $L$ units of supply and connect it to all the demand points by introducing edges with cost $0$. For every demand point, the amount of demand that was outlier in $\phi$ of $FI(\mathbf{v})$ is sent to $d_f$ at $0$ cost. To construct a feasible flow for $\mathbf{v}'$, add $\delta$ units of flow from $j$ to $i$. Make all the demand coming on to $d_f$ facility outlier. It is easy to see that the resulting flow is a feasible flow for $\mathbf{v}'$, and the cost of solution increases by at most $\delta R_t$. Therefore, $g(\mathbf{v}') \leq g(\mathbf{v}) + \delta R_t$.

We next construct a feasible flow for $FI(\mathbf{v})$ from optimal flow $\phi$ of $FI(\mathbf{v}')$ in a similar way: create a dummy facility $d_f$ with $L$ units of supply and connect it to all the demand points by introducing edges with cost $0$. For every demand point, the amount of demand that was outlier in $\phi$ of $FI(\mathbf{v}')$ is sent to $d_f$ at $0$ cost. To construct feasible flow for $\mathbf{v}$, add $\delta$ units of flow from $i$ to $j$. Make all the demand coming on to $d_f$ facility outlier. Again it is easy to see that this is a feasible flow for $\mathbf{v}$ and the cost of solution increases by at most $\delta R_t$. Therefore, $g(\mathbf{v}) \leq g(\mathbf{v}') + \delta R_t$.

The proof now follows in a similar way as done in Cohen-Addad and Li (2019), that is, the desired bound is obtained by applying the Chernoff bound for Lipschitz functions (stated in the following Proposition 5.9).

**Proposition 5.9** (Cohen-Addad and Li (2019))**.** *Let* $x_1, \ldots, x_n$ *be independent random variables taking value* $b$ *with probability* $p$ *and value* $0$ *with probability* $1-p$*, and let* $g : [0,1]^n \to \mathbb{R}$ *be a L-Lipschitz function in* $\ell_1$ *norm. Define* $X := (x_1, \ldots, x_n)$ *and* $\mu := \mathbb{E}[g(X)]$*. Then, for* $0 \leq \epsilon \leq 1$*,* $Pr[|g(X) - \mathbb{E}[g(X)|] \geq \epsilon pnbL] \leq 2e^{-\epsilon^2 pn/3}$.

We apply Proposition 5.9 on function $g$ with $X := \mathbf{X}$, $p := s/N$, $n := N$, $b := 1/p$, and $L := R_t$ to obtain the following:

$$\begin{aligned}
\Pr[|g(\mathbf{X}) - \mathbb{E}[g(\mathbf{X})]| \geq (\epsilon/2)NR_t] &= \Pr[|g(\mathbf{X}) - \mathbb{E}[g(\mathbf{X})]| \geq (\epsilon/2)pnbL] \\
&\leq 2 \cdot \exp\left(\frac{-(\epsilon/2)^2 pn}{3}\right) \\
&= 2 \cdot \exp\left(\frac{-(\epsilon/2)^2 (s/N)N}{3}\right) \\
&= \exp\left(-\Theta\left(\epsilon^2 s\right)\right) \\
&= \exp\left(-\Theta\left(\epsilon^2 \cdot \frac{L + k\log n}{\epsilon^2}\right)\right) \\
&\leq n^{-(k+\lambda_1)}
\end{aligned} \tag{5.6}$$

where the last equality follows by definition of $s$ and $\lambda_1$ is a constant in last inequality. This concludes the proof of Lemma 5.8. $\square$

**Lemma 5.10.** $|\mathbb{E}[g(\mathbf{X})] - g(\mathbb{E}[\mathbf{X}])| \leq \epsilon\lambda_2 NR_t$ *where* $\lambda_2$ *is a constant.*

**Proof.** We prove this in two parts $(i)$ $g(\mathbb{E}[\mathbf{X}]) \leq \mathbb{E}[g(\mathbf{X})]$ (proven in Section 5.4.1), and $(ii)$ $\mathbb{E}[g(\mathbf{X})] \leq g(\mathbb{E}[\mathbf{X}]) + \epsilon\lambda_2 NR_t$ (proven in Section 5.4.2). $\square$

Lemma 5.6 follows by adding Lemma 5.8 and Lemma 5.10 and modifying $\lambda_2 \leftarrow \lambda_2 + 1/2$.

### 5.4.1 Proof of Lemma 5.10, part $(i)$ $g(\mathbb{E}[\mathbf{X}]) \leq \mathbb{E}[g(\mathbf{X})]$

To prove, $g(\mathbb{E}[\mathbf{X}]) \leq \mathbb{E}[g(\mathbf{X})]$, we construct a feasible solution for $\mathsf{FI}(\mathbb{E}[\mathbf{X}])$ of cost no more than $\mathbb{E}[g(\mathbf{X})]$. Since the min-cost flow can only be lower, we get the desired result.

Let the outcomes of vector $\mathbf{X}$ be $\mathbf{v}^{(1)}, \mathbf{v}^{(2)}, \ldots$, with probability $p^{(1)}, p^{(2)}, \ldots$ respectively. We have, $\mathbb{E}[g(\mathbf{X})] = \sum_x p^{(x)} g(\mathbf{v}_x)$. Let $\phi_i$ be the flow obtained for $\mathsf{FI}(\mathbf{v}^{(x)})$. Now, consider the flow $\phi$ obtained by summing up over $x$, $\phi^{(x)}$ scaled by $p^{(x)}$. Observe that the cost of $\phi$ is at most $\sum_x p^{(x)} g(\mathbf{v}^{(x)})$, which is $= \mathbb{E}[g(\mathbf{X})]$.

Next, we will show that $\phi$ is a feasible flow for $\mathsf{FI}(\sum_x p^{(x)}\mathbf{v}^{(x)}) = \mathsf{FI}(\mathbb{E}(\mathbf{X}))$. For a client $j$, let $y^{(j,x)}$ be the demand in $\mathsf{FI}(\mathbf{v}^{(x)})$ and let $o_{j,x}$ be the demand that is left as outlier, i.e., $(y^{(j,x)} - o^{(j,x)})$ demand is satisfied in $\phi^{(x)}$. Therefore, in $\phi$, total $\sum_{j\in X}\sum_x p^{(x)} \cdot$

$(y^{(j,x)} - o^{(j,x)})$ demand is satisfied. And, $\sum_{j \in X} \sum_x p^{(x)} \cdot (y^{(j,x)} - o^{(j,x)}) = \sum_x p^{(x)} \cdot \sum_{j \in X} (y^{(j,x)} - o^{(j,x)}) = \sum_x p^{(x)}(|X| - L) = |X| - L$. The second last equality follows as $\phi^{(x)}$ is a feasible flow of $\mathsf{FI}(\mathbf{v}^{(x)})$ and the last equality follows because the sum of probabilities is 1.

Next, we show that the capacities are respected on every facility in flow $\phi$. For any facility $i \in \mathcal{F}$, let $i^{(x)}$ be the total flow coming onto $i$ in $\phi^{(x)}$. Therefore, total flow coming into $i$ in $\phi$ is $\sum_x j^{(x)} p^{(x)} \leq (\sum_x p^{(x)}) \cdot \max_x i^{(x)} = \max_x i^{(x)}$, which is at most the capacity of $i$ as $\phi^*$ is a feasible flow of $\mathsf{FI}(\mathbf{v}^{(x)})$.

### 5.4.2 Proof of Lemma 5.10, part $(ii)$ $\mathbb{E}[g(\mathbf{X})] \leq g(\mathbb{E}[\mathbf{X}]) + \epsilon\lambda_2 N R_t$

To prove this we first prove Claim 5.11 followed by Lemma 5.12.

**Claim 5.11.** *With probability* $1$*,* $g(\mathbf{X}) \leq g(\mathbb{E}[\mathbf{X}]) + nNR_t$*.*

**Proof.** The value of $g(\mathbf{X})$ lies in an interval of length $N/s \cdot N \cdot R_t \leq N^2 R_t$ because $\mathbf{X} \in [0, N/s]^N$ and function $g$ is $R_t$-Lipschitz. $g(\mathbb{E}[\mathbf{X}])$ also lies in the same interval because $\mathbb{E}[\mathbf{X}] = \mathbb{1} \in [0, N/s]^N$. Therefore, the claim follows. □

**Lemma 5.12.** *With probability* $1 - n^{-10}$*,* $g(\mathbf{X}) \leq g(\mathbb{E}[\mathbf{X}]) + \epsilon\lambda_2 N R_t$ *where* $\lambda_2$ *is a constant.*

**Proof.** To prove this, we construct a feasible flow $\phi$ for $FI(\mathbf{X})$ from the min cost flow with outliers of $FI(\mathbb{1})$ such that cost of flow $\phi$ is bounded by $g(\mathbb{E}[\mathbf{X}]) + \epsilon\lambda_2 N R_t$ (see Figure 5.4) for an illustration.

Consider the optimal min-cost flow with outliers $\phi'$ of $\mathsf{FI}(\mathbb{1})$. Let $o$ be the set of clients that are made outliers in this flow. Note that, $|o| \leq L$. Create a dummy facility $d_f$ and connect it to all the clients in $o$ at 0 cost. For every client $j \in o$, add 1 unit of flow from $j$ to $d_f$ in $\phi'$. Also, set the cost from ring center $i'$, to $d_f$ as 0.

For all the clients that are not in $X_t$, we route their 1 unit of demand in the same way as in $\phi'$. We are left with demands in $X_t$ and the extra $N - \sum_{j \in X_t} \mathbf{v}_j$ demand at the ring center $i'$.

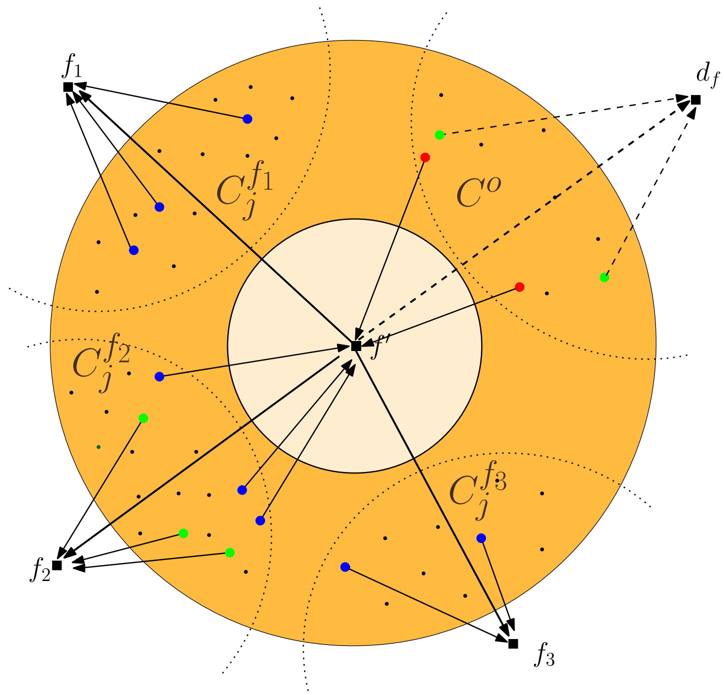



Figure 5.4: Illustration of flow rerouting to create a feasible flow $\phi$ for $\mathsf{FI}(\mathbf{X})$ from the optimal flow $\phi'$ of $\mathsf{FI}(\mathbf{1})$. $f'$ is the cluster center, $f_1$, $f_2$, $f_3$ are three other facilities. $d_f$ is the dummy facility to which outlier flow is directed at 0 cost. Dashed lines represent the 0 cost edges added for rerouting. Blue and red circles represent the sampled points from non-outlier and outlier flow in $\phi'$, respectively. Let $N/s$ be 4. $S^{f_1}$ and $S^{f_3}$ are under-sampled whereas $S^{f_2}$ and $S^{d_f}$ are over-sampled. Green circles represent the sub-sample picked from the over-sampled points.

For every facility $i \in F \cup d_f$, let $X_t^i \subseteq X_t$ be the set of clients served by $i$ in $\phi'$ and let $S^i \subseteq X_t^i$ be the sampled clients in $X_t^i$. If $S^i$ is under-sampled, i.e., $|S^i| \cdot \frac{N}{s} \leq |X_t^i|$, then in $\phi$ we route :

$(i)$ $\frac{N}{s}$ units of flow from every $j \in S^i$ to facility $i$ and,

$(ii)$ send $|X_t^i| - |S^i| \cdot \frac{N}{s}$ units of flow from ring center $i'$ to facility $i$.

Whereas if $S^i$ is over-sampled, i.e., $|S^i| \cdot \frac{N}{s} > |X_t^i|$, then we first pick a sub sample randomly from $S^i$, say $S_s^i$ of size $\lfloor |X_t^i| \cdot \frac{s}{N} \rfloor$. Now in $\phi$ we route:

$(i)$ $\frac{N}{s}$ units of flow from every $j \in S_s^i$ to facility $i$,

$(ii)$ send $|X_t^i| - |S_s^i| \cdot \frac{N}{s}$ units of flow from ring center $i'$ to facility $i$ and,

$(iii)$ $\frac{N}{s}$ units of flow from every $j \in S^i \setminus S^i_s$ to ring center $i'$.

Observe that the total amount of incoming flow on $d_f$ is $|o| \leq L$. Make all the demand coming on to $d_f$ outlier in $\phi$. It can be easily verified that the resulting flow $\phi$ is a feasible flow, for instance $\mathsf{Fl}(\mathbf{X})$.

The cost of flow of the clients in $X_t \setminus o$ is same as in Cohen-Addad and Li (2019) as stated in following lemma:

**Lemma 5.13** (Cohen-Addad and Li (2019)). *With probability at least $1 - n^{-10}$, the cost of clients in $X_t \setminus o$ in flow $\phi$ is bounded by* $\sum_{i \in \mathcal{F} \setminus d_f} \sum_{j \in X^i_t \setminus o} d(j, i) + 0.48\epsilon N R_t$.

Now, for clients in $o$, the cost paid from client to $d_f$ or from ring center to $d_f$ is $0$. Therefore, the only additional cost we paid is from a client $j \in o$ to the ring center $i'$ which is at most $R_t$. Since $|o| \leq L$, the total additional cost is at most $L \cdot R_t \leq \frac{\epsilon}{\lambda_2} N \cdot R_t$, where $\lambda_2$ is a constant. Adding this to Lemma 5.13, we obtain a total cost of, at most,

$$\sum_{i \in \mathcal{F}} \sum_{j \in X_t} d(j, i) + 0.48\epsilon N R_t + \epsilon \lambda_2 N R_t. \tag{5.7}$$

Modifying constant $\lambda_2 \coloneqq 0.48 + \lambda_2$ gives us Lemma 5.12. $\square$

From Claim 5.11 and Lemma 5.12, we have

$$\begin{aligned}
\mathbb{E}[g(\mathbf{X})] &\leq n^{-10} \cdot (g(\mathbb{E}[\mathbf{X}]) + nNR_t) + (1 - n^{-10})(g(\mathbb{E}[\mathbf{X}]) + \lambda_2 \epsilon N R_t) \\
&= g(\mathbb{E}[\mathbf{X}]) + (n^{-10} \cdot n + (1 - n^{-10}) \cdot \epsilon \lambda_2) N R_t \\
&\leq g(\mathbb{E}[\mathbf{X}]) + (n^{-9} + \epsilon \lambda_2) N R_t, \\
&\leq g(\mathbb{E}[\mathbf{X}]) + 2\epsilon \lambda_2 N R_t,
\end{aligned} \tag{5.8}$$

where we use the assumption that $\epsilon > 0$ is a constant, and hence $\epsilon \lambda_2 \geq n^{-9}$. By redefining $\lambda_2 \coloneqq 2\lambda_2$, this finishes the proof that $\mathbb{E}[g(\mathbf{X})] \leq g(\mathbb{E}[\mathbf{X}]) + \epsilon \lambda_2 N R_t$.

## 5.5 Analysis of Multiple Rings Case

The argument from Section 5.4 can be generalized to handle the more general situation, where we perform sampling in multiple rings. At a high level, the strategy is similar to

(2019), in that we process the rings in an arbitrary order, and while analyzing the error incurred by sampling in a particular ring, we condition on a fixed outcome of sampling in the rings that occur prior to the current ring; and this situation is, in spirit, similar to the single ring case as in Section 5.4. We show that this error is small when conditioned on the prior outcomes, with high probability. Finally, by taking a union bound over all rings, the overall error is shown to be small with high probability. We next formalize this notion of conditional expectations which completes a formal proof of Lemma 5.2.

Consider the rings in any arbitrary order $\sigma$. For any two rings $X_{i,t}$ and $X_{i',t'}$, we say $(i,t) < (i',t')$ if $X_{i',t'}$ comes after $X_{i,t}$ in $\sigma$. Fix a ring $X_{i,t}$. Now we define a function $g_{i,t}$ for each ring similar to the function $g$ defined in Section 5.4. As done in Section 5.4, to define $g_{i,t}$, we first create an instance $\mathsf{FI}(\mathbb{Y})$ of min cost flow with outliers corresponding to a vector $\mathbb{Y}$ of size $n$. To create this instance, we define a random vector $\mathbf{X} \in \mathbb{R}_+^{|X_{i,t}|}$ where each coordinate pick value $\frac{|X_{i,t}|}{s}$ with probability $\frac{s}{|X_{i,t}|}$ and 0 otherwise. In $\mathsf{FI}(\mathbb{Y})$, every client $j \in X_{i,t}$ has $\mathbf{X}_j$ demand and the cluster center $i$ has $|X_{i,t}| - \sum_{j \in X_{i,t}} \mathbf{X}_j$ demand. But, as we consider the sample from the other rings too, $g_{i,t}$ will also depend on these samples. Suppose we fix samples $S_{i',t'}$ for every ring $X_{i',t'}$, $(i',t') \neq (i,t)$. In $\mathsf{FI}(\mathbb{Y})$, set demand $\frac{|X_{i',t'}|}{s}$ at each client $j \in S_{i',t'}$. $g_{i,t}(\mathbb{Y})$ is the optimal cost of $\mathsf{FI}(\mathbb{Y})$.

Let $\mathbb{E}_{S_{i',t'}:(i',t')>(i,t)}[g_{i,t}(\mathbb{Y}) | S_{i',t'} : (i',t') < (i,t)]$ or in short $\mathbb{E}_{>(i,t)}[g_{i,t}(\mathbb{Y})]$ be the expectation of $g_{i,t}$ over all samples $S_{i',t'}$ for $(i',t') > (i,t)$ given fixed samples $S_{i',t'}$ for all $(i',t') < (i,t)$. Similarly, define $\mathbb{E}_{S_{i',t'}:(i',t')\geq(i,t)}[g_{i,t}(\mathbb{Y}) | S_{i',t'} : (i',t') < (i,t)]$ or in short $\mathbb{E}_{\geq(i,t)}[g_{i,t}(\mathbb{Y})]$ be the expectation of $g_{i,t}$ over all samples $S_{i',t'}$ for $(i',t') \geq (i,t)$ given fixed samples $S_{i',t'}$ for all $(i',t') < (i,t)$.

Let $(i_1,t_1)$ and $(i_l,t_l)$ be the indexes of the first and last rings in this order, respectively. Then, $\mathbb{E}_{\geq(i_1,i_1)}[g_{i,t}(\mathbb{Y})] = \mathsf{cost}_L(X, \mathcal{F})$, and $\mathbb{E}_{>(i_l,t_l)}[g_{i,t}(\mathbb{Y})] = \mathsf{wcost}_L(W, \mathcal{F})$.

With these definitions, we get the following lemma analogous to Lemma 5.8 and Lemma 5.10 (combined) in the single ring case.

**Lemma 5.14.** *With probability at least $1 - n^{-(k+\lambda_1)}$, for any ring $X_{i,t}$, $|\mathbb{E}_{>(i,t)}[g_{i,t}(\mathbb{Y})] - \mathbb{E}_{\geq(i,t)}[g_{i,t}(\mathbb{Y})]| \leq \epsilon\lambda_2 |X_{i,t}| R_t$ where $R_t$ is the radius of ring $X_{i,t}$ and $\lambda_1, \lambda_2$ are constants.*

Combining over all rings and using respective definitions of $\mathbb{E}_{>(f,j)}[\mathsf{wcost}_L(W,F)]$

and $\mathbb{E}_{\geq(f,j)}[\mathsf{wcost}_L(W, F)]$, we get the following lemma:

**Lemma 5.15.** *For any feasible* $\mathcal{F} \subseteq F$, $|\mathbb{E}_{>(i,t)}[\mathsf{wcost}_L(W, \mathcal{F})] - \mathbb{E}_{\geq(i,t)}[\mathsf{wcost}_L(W, \mathcal{F})]| \leq \epsilon\lambda_2 |X_{i,t}| R_t$ *with probability* $1 - n^{-(k+\lambda_1)}$.

We take union bound over all possible sets of feasible solutions, which gives that inequality in Lemma 5.15 fails with probability $\leq n^{-c}$ and hence the lemma holds with high probability. Now consider the process of going through all the rings $X_{i,t}$ according to $\sigma$. Applying Lemma 5.15 on all $O((k + L)\log n) \leq n^2$ rings conditioned on the choices of $S_{i',t'}$ for $(i', t') < (i, t)$. We get the following with high probability,

$$
\begin{aligned}
|\mathsf{wcost}_L(W, \mathcal{F}) - \mathbb{E}[\mathsf{wcost}_L(W, \mathcal{F})]| &\leq \sum_{(i,t)} \epsilon\lambda_2 |X_{i,t}| R_t \\
&= 2\epsilon\lambda_2 \cdot \sum_{(i,t)} |X_{i,j}| \cdot R_t/2 \\
&\leq 2\epsilon\lambda_2 \cdot \mathsf{cost}_0(X, \mathcal{F}_\zeta) \\
&\leq 2\epsilon\lambda_2 \cdot \zeta \cdot \mathsf{OPT}(I) \\
&\leq O(\epsilon)\mathsf{cost}_m(C, \mathcal{F}),
\end{aligned} \tag{5.9}
$$

where the second last inequality follows from Equation (5.1). Note that, $\mathbb{E}[\mathsf{wcost}_L(W, \mathcal{F})] = \mathsf{cost}_L(X, \mathcal{F})$. Therefore, scaling down $\epsilon$ by a constant factor to get $\epsilon \cdot \mathsf{cost}_L(X, \mathcal{F})$ from $O(\epsilon)\mathsf{cost}_L(X, \mathcal{F})$ gives us Lemma 5.2.

## 5.6 Proof of Theorem 1.4

In this section, we explore the possibility of obtaining FPT approximation solution for CFLO. While a natural candidate for the FPT parameter is the number of outliers, it is not immediately clear whether an FPT approximation can be achieved based solely on this parameter. However, if the number of facilities in the optimal solution is known, it becomes possible to design an FPT algorithm that is parameterized by the number of outliers and the solution size. Since the optimal number of facilities is generally unknown, we adopt the standard approach of including a bound, $k$, on the solution size as part of the input. This transforms the problem into a generalization, called C$k$FLO, which combines aspects of both CFLO and C$k$MO. In particular, we prove Theorem 1.4.

**Theorem 1.4.** *There exists a $(3+\epsilon)$ approximation for C$k$FLO that runs in time FPT in $k$, the number of outliers and $\epsilon$ where $\epsilon > 0$ is a small constant.*

Our reduction from C$k$MO to C$k$M also works in the presence of opening costs – indeed, the weighted sample $W$ approximately preserves distances w.r.t. all sets $\mathcal{F} \subseteq F$ w.h.p., and the opening costs of the facilities are unaffected by the sampling process. Thus, Theorem 1.5, which gives a reduction from C$k$MO to C$k$M, in fact, generalizes to give an FPT reduction from C$k$FLO to C$k$FL. We also observe that the $(3+\epsilon)$ FPT approximation for C$k$M from Cohen et al. (2019) can be easily adapted for C$k$FL – in the enumeration part, one also has to guess the opening cost of the closest facility to each "leader" up to a factor of $(1+\epsilon)$. Therefore, we obtain $(3+\epsilon)$-approximation for C$k$FLO.

# Chapter 6

# Conclusion

This thesis presented several approximation algorithms for facility location problems, with a particular focus on generalizations of the capacitated facility location and facility location with outliers problems. These problems combine the challenges of handling capacity constraints and the flexibility to exclude certain clients.

We first studied the capacitated facility location with outliers (CFLO) problem. In the case of uniform facility costs, a novel $(6.373 + \epsilon)$-factor approximation algorithm was developed using a two-operation local search approach. To the best of our knowledge, this is the first approximation algorithm for the problem. Additionally, we presented a $(3.733 + \epsilon)$-factor approximation for the non-outlier variant of the problem which is also the current best approximation for the capacitated facility location with uniform facility costs.

For non-uniform facility costs, we conjectured that the locality gap for (uncapacitated) facility location with outliers is unbounded, even with constant factor violation in outliers. To support this conjecture, we provided an example where escaping the unbounded locality gap involves solving another instance of facility location with outliers problem. The locality gap example illustrates that obtaining a constant-factor approximation is challenging for CFLO even for uniform capacities despite allowing violations in capacities and outliers using the local search technique. We, therefore, used LP-based algorithms for CFLO with uniform capacities. We made some progress by presenting a tri-criteria approximation based on rounding the solution to the standard LP relaxation.

Specifically, we provide an $O(1/\epsilon^2)$-factor approximation, with $(1+\epsilon)$-factor violations in both capacity and outlier constraints. These violations are inevitable because both CFL and FLO have unbounded integrality gaps for standard LP formulations. Furthermore, the tri-criteria approach could be useful in the future for eliminating violations in capacities, outliers, or both.

Note that, when the facility opening costs are uniform, the algorithm in Chapter 4 can be modified to get rid of violations in capacities. Although the overall result is weaker compared to the result in Chapter 3, it remains noteworthy because it is derived using a straightforward LP-rounding approach and hence may be useful in integration with other LP-based algorithms.

Building on recent progress in fixed-parameter tractability (FPT), we also gave a $(3+\epsilon)$ FPT approximation for CFLO (with general opening cost and general capacities), FPT in $k$, $L$ and $\epsilon$ where $k$ is the bound on the solution size given as an additional input parameter.

Furthermore, the thesis explored the Capacitated $k$-Median with Outliers problem. We introduced an approximation-preserving reduction from this problem to the standard capacitated $k$-Median problem. This reduction runs in time FPT in the number of facilities, the number of outliers, and a small constant, yielding a $(3+\epsilon)$-factor approximation for the problem.

**Open problems and future direction:** Obtaining a true constant-factor approximation for the capacitated facility location with outliers and general opening costs (even when capacities are uniform) remains unsolved. We believe that ideas based on the rounding of strengthened LPs for the capacitated facility location problem may hold the potential to give us some success for the outlier variant. Additionally, the hardness lower bound for the problem remains at $1.463$, the same lower bound as for the classical facility location problem. Improving this bound is another interesting question. For the capacitated $k$-Median with outliers problem, an interesting open problem is to develop a polynomial-time constant-factor approximation, even when violations in capacities or outliers is allowed.

# Bibliography


Aardal, K. I., van den Berg, P. L., Gijswijt, D., and Li, S. (2015). Approximation algorithms for hard capacitated $k$-facility location problems. *Eur. J. Oper. Res.*, 242(2):358–368.

Adamczyk, M., Byrka, J., Marcinkowski, J., Meesum, S. M., and Wlodarczyk, M. (2019). Constant-factor FPT approximation for capacitated $k$-median. In Bender, M. A., Svensson, O., and Herman, G., editors, *27th Annual European Symposium on Algorithms, ESA 2019, September 9-11, 2019, Munich/Garching, Germany*, volume 144 of *LIPIcs*, pages 1:1–1:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Aggarwal, A., Louis, A., Bansal, M., Garg, N., Gupta, N., Gupta, S., and Jain, S. (2010). A 3-approximation for facility location with uniform capacities. In Eisenbrand, F. and Shepherd, F. B., editors, *Integer Programming and Combinatorial Optimization, 14th International Conference, IPCO 2010, Lausanne, Switzerland, June 9-11, 2010. Proceedings*, volume 6080 of *Lecture Notes in Computer Science*, pages 149–162. Springer.

Agrawal, A., Inamdar, T., Saurabh, S., and Xue, J. (2023). Clustering what matters: Optimal approximation for clustering with outliers. In Williams, B., Chen, Y., and Neville, J., editors, *AAAI*, pages 6666–6674. AAAI Press.

Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993). *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall.

An, H., Bhaskara, A., Chekuri, C., Gupta, S., Madan, V., and Svensson, O. (2015). Centrality of trees for capacitated k-center. *Math. Program.*, 154(1-2):29–53.

An, H., Singh, M., and Svensson, O. (2014). Lp-based algorithms for capacitated facility location. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 256–265. IEEE Computer Society.

Arya, V., Garg, N., Khandekar, R., Meyerson, A., Munagala, K., and Pandit, V. (2001). Local search heuristic for $k$-median and facility location problems. In Vitter, J. S., Spirakis, P. G., and Yannakakis, M., editors, *Proceedings on 33rd Annual ACM Symposium on Theory of Computing, July 6-8, 2001, Heraklion, Crete, Greece*, pages 21–29. ACM.

Arya, V., Garg, N., Khandekar, R., Meyerson, A., Munagala, K., and Pandit, V. (2004). Local search heuristics for k-median and facility location problems. *SIAM Journal on Computing*, 33(3):544–562.

Backurs, A., Indyk, P., Onak, K., Schieber, B., Vakilian, A., and Wagner, T. (2019). Scalable fair clustering. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 405–413. PMLR.

Bandyapadhyay, S., Fomin, F. V., and Simonov, K. (2021). On coresets for fair clustering in metric and euclidean spaces and their applications. In Bansal, N., Merelli, E., and Worrell, J., editors, *ICALP*, volume 198 of *LIPIcs*, pages 23:1–23:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Bansal, M., Garg, N., and Gupta, N. (2012). A 5-approximation for capacitated facility location. In Epstein, L. and Ferragina, P., editors, *Algorithms - ESA 2012 - 20th Annual European Symposium, Ljubljana, Slovenia, September 10-12, 2012. Proceedings*, volume 7501 of *Lecture Notes in Computer Science*, pages 133–144. Springer.

Braverman, V., Cohen-Addad, V., Jiang, S. H., Krauthgamer, R., Schwiegelshohn, C., Toftrup, M. B., and Wu, X. (2022). The power of uniform sampling for coresets. In *63rd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2022, Denver, CO, USA, October 31 - November 3, 2022*, pages 462–473. IEEE.

Byrka, J. (2007). An optimal bifactor approximation algorithm for the metric uncapacitated facility location problem. In Charikar, M., Jansen, K., Reingold, O., and Rolim, J. D. P., editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 10th International Workshop, APPROX 2007, and 11th International Workshop, RANDOM 2007, Princeton, NJ, USA, August 20-22, 2007, Proceedings*, volume 4627 of *Lecture Notes in Computer Science*, pages 29–43. Springer.

Byrka, J., Fleszar, K., Rybicki, B., and Spoerhase, J. (2015a). Bi-factor approximation algorithms for hard capacitated $k$-median problems. In Indyk, P., editor, *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. SIAM.

Byrka, J., Pensyl, T. W., Rybicki, B., Srinivasan, A., and Trinh, K. (2015b). An improved approximation for $k$-median, and positive correlation in budgeted optimization. In Indyk, P., editor, *Proceedings of the*

*Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 737–756. SIAM.

Byrka, J., Rybicki, B., and Uniyal, S. (2016). An approximation algorithm for uniform capacitated $k$-median problem with (1+$\epsilon$) capacity violation. In Louveaux, Q. and Skutella, M., editors, *Integer Programming and Combinatorial Optimization - 18th International Conference, IPCO 2016, Liège, Belgium, June 1-3, 2016, Proceedings*, volume 9682 of *Lecture Notes in Computer Science*, pages 262–274. Springer.

Chakrabarty, D., Goyal, P., and Krishnaswamy, R. (2016). The non-uniform $k$-center problem. In Chatzigiannakis, I., Mitzenmacher, M., Rabani, Y., and Sangiorgi, D., editors, *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, volume 55 of *LIPIcs*, pages 67:1–67:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Charikar, M. and Guha, S. (1999). Improved combinatorial algorithms for the facility location and $k$-median problems. In *40th Annual Symposium on Foundations of Computer Science, FOCS '99, 17-18 October, 1999, New York, NY, USA*, pages 378–388. IEEE Computer Society.

Charikar, M. and Guha, S. (2005). Improved combinatorial algorithms for facility location problems. *SIAM J. Comput.*, 34(4):803–824.

Charikar, M., Guha, S., Tardos, É., and Shmoys, D. B. (1999). A constant-factor approximation algorithm for the $k$-median problem (extended abstract). In Vitter, J. S., Larmore, L. L., and Leighton, F. T., editors, *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, May 1-4, 1999, Atlanta, Georgia, USA*, pages 1–10. ACM.

Charikar, M., Khuller, S., Mount, D. M., and Narasimhan, G. (2001). Algorithms for facility location problems with outliers. In Kosaraju, S. R., editor, *SODA*, pages 642–651. ACM/SIAM.

Charikar, M. and Li, S. (2012). A dependent lp-rounding approach for the $k$-median problem. In Czumaj, A., Mehlhorn, K., Pitts, A. M., and Wattenhofer, R., editors, *Automata, Languages, and Programming - 39th International Colloquium, ICALP 2012, Warwick, UK, July 9-13, 2012, Proceedings, Part I*, volume 7391 of *Lecture Notes in Computer Science*, pages 194–205. Springer.

Chen, K. (2008). A constant factor approximation algorithm for $k$-median clustering with outliers. In Teng, S., editor, *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2008, San Francisco, California, USA, January 20-22, 2008*, pages 826–835. SIAM.

Chen, K. (2009). On coresets for $k$-median and $k$-means clustering in metric and euclidean spaces and their applications. *SIAM J. Comput.*, 39(3):923–947.

Chudak, F. A. (1998). Improved approximation algorithms for uncapitated facility location. In Bixby, R. E., Boyd, E. A., and Ríos-Mercado, R. Z., editors, *Integer Programming and Combinatorial Optimization, 6th International IPCO Conference, Houston, Texas, USA, June 22-24, 1998, Proceedings*, volume 1412 of *Lecture Notes in Computer Science*, pages 180–194. Springer.

Chudak, F. A. and Shmoys, D. B. (2003). Improved approximation algorithms for the uncapacitated facility location problem. *SIAM J. Comput.*, 33(1):1–25.

Chudak, F. A. and Williamson, D. P. (1999). Improved approximation algorithms for capacitated facility location problems. In Cornuéjols, G., Burkard, R. E., and Woeginger, G. J., editors, *Integer Programming and Combinatorial Optimization, 7th International IPCO Conference, Graz, Austria, June 9-11, 1999, Proceedings*, volume 1610 of *Lecture Notes in Computer Science*, pages 99–113. Springer.

Chuzhoy, J. and Rabani, Y. (2005). Approximating $k$-median with non-uniform capacities. In *SODA*, pages 952–958. SIAM.

Cohen, M. B., Lee, Y. T., and Song, Z. (2021). Solving linear programs in the current matrix multiplication time. *J. ACM*, 68(1):3:1–3:39.

Cohen-Addad, V., Grandoni, F., Lee, E., and Schwiegelshohn, C. (2023). Breaching the 2 LMP approximation barrier for facility location with applications to *k*-median. In Bansal, N. and Nagarajan, V., editors, *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, pages 940–986. SIAM.

Cohen-Addad, V., Gupta, A., Hu, L., Oh, H., and Saulpic, D. (2022). An improved local search algorithm for $k$-median. In Naor, J. S. and Buchbinder, N., editors, *Proceedings of the 2022 ACM-SIAM Symposium on Discrete Algorithms, SODA 2022, Virtual Conference / Alexandria, VA, USA, January 9 - 12, 2022*, pages 1556–1612. SIAM.

Cohen-Addad, V., Gupta, A., Kumar, A., Lee, E., and Li, J. (2019). Tight FPT approximations for $k$-median and $k$-means. In Baier, C., Chatzigiannakis, I., Flocchini, P., and Leonardi, S., editors, *ICALP*, volume 132 of *LIPIcs*, pages 42:1–42:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Cohen-Addad, V. and Li, J. (2019). On the fixed-parameter tractability of capacitated clustering. In Baier, C., Chatzigiannakis, I., Flocchini, P., and Leonardi, S., editors, *ICALP*, volume 132 of *LIPIcs*, pages 41:1–41:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Cohen-Addad, V., Saulpic, D., and Schwiegelshohn, C. (2021). A new coreset framework for clustering. In Khuller, S. and Williams, V. V., editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 169–182. ACM.

Cygan, M. and Kociumaka, T. (2014). Constant factor approximation for capacitated k-center with outliers. In Mayr, E. W. and Portier, N., editors, *31st International Symposium on Theoretical Aspects of Computer Science (STACS 2014), STACS 2014, March 5-8, 2014, Lyon, France*, volume 25 of *LIPIcs*, pages 251–262. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Dabas, R., Garg, N., and Gupta, N. (2024). Capacitated facility location with outliers and uniform facility costs. In Vygen, J. and Byrka, J., editors, *Integer Programming and Combinatorial Optimization - 25th International Conference, IPCO 2024, Wrocław, Poland, July 3-5, 2024, Proceedings*, volume 14679 of *Lecture Notes in Computer Science*, pages 85–98. Springer.

Dabas, R. and Gupta, N. (2022). Capacitated facility location with outliers/penalties. In Zhang, Y., Miao, D., and Möhring, R. H., editors, *Computing and Combinatorics - 28th International Conference, COCOON 2022, Shenzhen, China, October 22-24, 2022, Proceedings*, volume 13595 of *Lecture Notes in Computer Science*, pages 549–560. Springer.

Dabas, R. and Gupta, N. (2025). Uniform capacitated facility location with outliers/penalties. *Discrete Optimization*, 55:100878.

Dabas, R., Gupta, N., and Inamdar, T. (2025). FPT approximation for capacitated clustering with outliers. *Theor. Comput. Sci.*, 1027:115026.

Demirci, H. G. and Li, S. (2016). Constant approximation for capacitated $k$-median with $(1 + \epsilon)$-capacity violation. In Chatzigiannakis, I., Mitzenmacher, M., Rabani, Y., and Sangiorgi, D., editors, *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, volume 55 of *LIPIcs*, pages 73:1–73:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Friggstad, Z., Khodamoradi, K., Rezapour, M., and Salavatipour, M. R. (2018). Approximation schemes for clustering with outliers. pages 398–414.

Friggstad, Z., Khodamoradi, K., Rezapour, M., and Salavatipour, M. R. (2019). Approximation schemes for clustering with outliers. *ACM Trans. Algorithms*, 15(2):26:1–26:26.

Gowda, K. N., Pensyl, T. W., Srinivasan, A., and Trinh, K. (2023). Improved bi-point rounding algorithms and a golden barrier for *k*-median. In Bansal, N. and Nagarajan, V., editors, *SODA*, pages 987–1011. SIAM.

Goyal, D. and Jaiswal, R. (2023). Tight FPT approximation for constrained *k*-center and *k*-supplier. *Theor. Comput. Sci.*, 940(Part):190–208.

Goyal, D., Jaiswal, R., and Kumar, A. (2020). FPT approximation for constrained metric k-median/means. In Cao, Y. and Pilipczuk, M., editors, *15th International Symposium on Parameterized and Exact*

*Computation, IPEC 2020, December 14-18, 2020, Hong Kong, China (Virtual Conference)*, volume 180 of *LIPIcs*, pages 14:1–14:19. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Grover, S., Gupta, N., and Khuller, S. (2022). Lp-based approximation for uniform capacitated facility location problem. *Discret. Optim.*, 45:100723.

Grover, S., Gupta, N., Khuller, S., and Pancholi, A. (2018a). Constant factor approximation algorithm for uniform hard capacitated knapsack median problem. In Ganguly, S. and Pandya, P. K., editors, *FSTTCS*, volume 122 of *LIPIcs*, pages 23:1–23:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Grover, S., Gupta, N., and Pancholi, A. (2018b). Improved local search based approximation algorithm for hard uniform capacitated k-median problem. *Informatica (Slovenia)*, 42(3).

Guha, S. and Khuller, S. (1998). Greedy strikes back: Improved facility location algorithms. In Karloff, H. J., editor, *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, 25-27 January 1998, San Francisco, California, USA*, pages 649–657. ACM/SIAM.

Guha, S. and Khuller, S. (1999). Greedy strikes back: Improved facility location algorithms. *J. Algorithms*, 31(1):228–248.

Gupta, A., Moseley, B., and Zhou, R. (2021). Structural iterative rounding for generalized k-median problems. In Bansal, N., Merelli, E., and Worrell, J., editors, *ICALP*, volume 198 of *LIPIcs*, pages 77:1–77:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Gupta, A. and Tangwongsan, K. (2008). Simpler analyses of local search algorithms for facility location. *CoRR*, abs/0809.2554.

Hochbaum, D. S. (1982). Heuristics for the fixed cost median problem. *Math. Program.*, 22(1):148–162.

Hochbaum, D. S. and Shmoys, D. B. (1985). A best possible heuristic for the $k$-center problem. *Math. Oper. Res.*, 10(2):180–184.

Huang, L., Jiang, S. H., and Vishnoi, N. K. (2019). Coresets for clustering with fairness constraints. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R., editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 7587–7598.

Jain, K., Mahdian, M., Markakis, E., Saberi, A., and Vazirani, V. V. (2003). Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. *J. ACM*, 50(6):795–824.

Jain, K., Mahdian, M., and Saberi, A. (2002). A new greedy approach for facility location problems. In Reif, J. H., editor, *Proceedings on 34th Annual ACM Symposium on Theory of Computing, May 19-21, 2002, Montréal, Québec, Canada*, pages 731–740. ACM.

Jain, K. and Vazirani, V. V. (1999). Primal-dual approximation algorithms for metric facility location and k-median problems. In *40th Annual Symposium on Foundations of Computer Science, FOCS '99, 17-18 October, 1999, New York, NY, USA*, pages 2–13. IEEE Computer Society.

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

Jaiswal, R. and Kumar, A. (2023). Clustering what matters in constrained settings: Improved outlier to outlier-free reductions. In Iwata, S. and Kakimura, N., editors, *34th International Symposium on Algorithms and Computation, ISAAC 2023, December 3-6, 2023, Kyoto, Japan*, volume 283 of *LIPIcs*, pages 41:1–41:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Kao, M. (2023a). Improved approximation algorithm for capacitated facility location with uniform facility cost. In Iwata, S. and Kakimura, N., editors, *34th International Symposium on Algorithms and Computation, ISAAC 2023, December 3-6, 2023, Kyoto, Japan*, volume 283 of *LIPIcs*, pages 45:1–45:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Kao, M. (2023b). On the integrality gap of MFN relaxation for the capacitated facility location problem. In Bansal, N. and Nagarajan, V., editors, *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, pages 1071–1089. SIAM.

Korupolu, M. R., Plaxton, C. G., and Rajaraman, R. (1998). Analysis of a local search heuristic for facility location problems. In Karloff, H. J., editor, *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, 25-27 January 1998, San Francisco, California, USA*, pages 1–10. ACM/SIAM.

Korupolu, M. R., Plaxton, C. G., and Rajaraman, R. (2000). Analysis of a local search heuristic for facility location problems. *J. Algorithms*, 37(1):146–188.

Krishnaswamy, R., Kumar, A., Nagarajan, V., Sabharwal, Y., and Saha, B. (2011). The matroid median problem. In Randall, D., editor, *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. SIAM.

Krishnaswamy, R., Li, S., and Sandeep, S. (2018). Constant approximation for $k$-median and $k$-means with outliers via iterative rounding. In Diakonikolas, I., Kempe, D., and Henzinger, M., editors, *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 646–659. ACM.

Levi, R., Shmoys, D. B., and Swamy, C. (2012). Lp-based approximation algorithms for capacitated facility location. *Math. Program.*, 131(1-2):365–379.

Li, S. (2011). A 1.488 approximation algorithm for the uncapacitated facility location problem. In Aceto, L., Henzinger, M., and Sgall, J., editors, *Automata, Languages and Programming - 38th International Colloquium, ICALP 2011, Zurich, Switzerland, July 4-8, 2011, Proceedings, Part II*, volume 6756 of *Lecture Notes in Computer Science*, pages 77–88. Springer.

Li, S. (2013). A 1.488 approximation algorithm for the uncapacitated facility location problem. *Inf. Comput.*, 222:45–58.

Li, S. (2014). An improved approximation algorithm for the hard uniform capacitated k-median problem. In Jansen, K., Rolim, J. D. P., Devanur, N. R., and Moore, C., editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2014, September 4-6, 2014, Barcelona, Spain*, volume 28 of *LIPIcs*, pages 325–338. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Li, S. (2015). On uniform capacitated $k$-median beyond the natural LP relaxation. In Indyk, P., editor, *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 696–707. SIAM.

Li, S. (2016). Approximating capacitated $k$-median with $(1+\epsilon)k$ open facilities. In Krauthgamer, R., editor, *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 786–796. SIAM.

Li, S. and Svensson, O. (2013). Approximating k-median via pseudo-approximation. In Boneh, D., Roughgarden, T., and Feigenbaum, J., editors, *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 901–910. ACM.

Lin, J. and Vitter, J. S. (1992). $\epsilon$-approximations with minimum packing constraint violation (extended abstract). In Kosaraju, S. R., Fellows, M., Wigderson, A., and Ellis, J. A., editors, *Proceedings of the 24th Annual ACM Symposium on Theory of Computing, May 4-6, 1992, Victoria, British Columbia, Canada*, pages 771–782. ACM.

Mahdian, M. and Pál, M. (2003). Universal facility location. In Battista, G. D. and Zwick, U., editors, *Algorithms - ESA 2003, 11th Annual European Symposium, Budapest, Hungary, September 16-19, 2003, Proceedings*, volume 2832 of *Lecture Notes in Computer Science*, pages 409–421. Springer.

Mahdian, M., Ye, Y., and Zhang, J. (2002). Improved approximation algorithms for metric facility location problems. In Jansen, K., Leonardi, S., and Vazirani, V. V., editors, *Approximation Algorithms for*

*Combinatorial Optimization, 5th International Workshop, APPROX 2002, Rome, Italy, September 17-21, 2002, Proceedings*, volume 2462 of *Lecture Notes in Computer Science*, pages 229–242. Springer.

Pál, M., Tardos, É., and Wexler, T. (2001). Facility location with nonuniform hard capacities. In *42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14-17 October 2001, Las Vegas, Nevada, USA*, pages 329–338. IEEE Computer Society.

Schmidt, M., Schwiegelshohn, C., and Sohler, C. (2019). Fair coresets and streaming algorithms for fair $k$-means. In Bampis, E. and Megow, N., editors, *Approximation and Online Algorithms - 17th International Workshop, WAOA 2019, Munich, Germany, September 12-13, 2019, Revised Selected Papers*, volume 11926 of *Lecture Notes in Computer Science*, pages 232–251. Springer.

Shmoys, D. B., Tardos, É., and Aardal, K. I. (1997). Approximation algorithms for facility location problems (extended abstract). In Leighton, F. T. and Shor, P. W., editors, *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, El Paso, Texas, USA, May 4-6, 1997*, pages 265–274. ACM.

Sviridenko, M. (2002). An improved approximation algorithm for the metric uncapacitated facility location problem. In Cook, W. J. and Schulz, A. S., editors, *Integer Programming and Combinatorial Optimization, 9th International IPCO Conference, Cambridge, MA, USA, May 27-29, 2002, Proceedings*, volume 2337 of *Lecture Notes in Computer Science*, pages 240–257. Springer.

Williamson, D. P. and Shmoys, D. B. (2011). *The Design of Approximation Algorithms*. Cambridge University Press.

Zhang, J., Chen, B., and Ye, Y. (2005). A multiexchange local search algorithm for the capacitated facility location problem. *Journal of Mathematics of Operations Research*, 30(2):389–403.