Multivariate Analysis of Clustering Problems with Constraints

Nidhi Purohit

Thesis for the degree of Philosophiae Doctor (PhD)
University of Bergen, Norway
2023

UNIVERSITY OF BERGEN

Multivariate Analysis of Clustering Problems with Constraints

Nidhi Purohit

Thesis for the degree of Philosophiae Doctor (PhD)
at the University of Bergen

Date of defense: 14.12.2023

Year: 2023

Title: Multivariate Analysis of Clustering Problems with Constraints

Name: Nidhi Purohit

Print: Skipnes Kommunikasjon / University of Bergen

Acknowledgements

I would like to express my sincere gratitude to my advisor, Professor Petr A. Golovach, for being patient and calm throughout the whole time. Your encouragement and patience were essential, especially during the review and editing of manuscript drafts multiple times. Your mentorship, valuable insights throughout the research process, availability for discussions and guidance, encouragement of collaborations and support have been instrumental in completing this work. Also, Thank you for checking my progress and taking the necessary steps towards my growth as a researcher. I would also like to thank my co-advisor, Professor Fedor V. Fomin, for his time, valuable suggestions and for always being open for discussions.

I am incredibly grateful to Professor Saket Saurabh for playing a pivotal role in my scientific development by providing me with immense opportunities and resources, such as arranging research visits and courses that enhanced my learning. Your presence filled everything around you with energy and enthusiasm. I am deeply thankful to you for everything.

I express my gratitude to all my co-authors. This work would not have been possible without your contribution. Thanks, Sayan Bandyapadhyay, for being the cooperative office mate and great teacher I could hope for at the beginning of my PhD. Thanks, Kirill Simonov, for our scientific conversations over the coffee pantry. William Lochet, thanks for your valuable insight early in the morning. Thanks, Tanmay, for sharing your scientific insights and patiently answering my doubts. I would also like to thank the rest of my co-authors, Jayakrishnan Madathil, Édouard Bonnet, Lawqueen Kanesh, Madhumita Kundu, Komal Muluk, and Avinandan Das. I am fortunate to have worked alongside you all.

It has been an immense pleasure to be part of the Algorithmic group of the Department of Informatics. The seminars and workshops helped me learn about a variety of topics. I would like to acknowledge the financial support provided by the department, which enable me to pursue this research. I especially wish to thank those I have spent time with outside work. Matthias, your vibrant and helpful presence in our office made a huge

difference. Thanks for the suggestions, which helped me improve my thesis's quality. I am grateful to Jan Arne, and Kari for a fantastic host. Talking to Kari is always been a pleasure. My heartiest thanks to Paloma and Lars for being a support system and source of guidance ranging from research to practical information during the initial days in Bergen. Thanks to the department's administrative staff for their assistance, which greatly facilitated the administrative aspects of my thesis.

I am thankful to my friends whose friendship and the countless moments of laughter and shared experiences have provided a much-needed respite from the challenges of research and study. Thank you, Athira, for being a family away from home. Your delicious homemade meals provided sustenance, warmth, and moments of joy. Your conscious effort to be there for me, whether through a kind word or a comforting gesture, has meant the world to me. Thanks, Prithvi, for being an extremely cooperative and considerate flatmate and for the mutual respect that has allowed us to coexist harmoniously. My heartfelt thanks to Mohanapriya, with whom I had one of the most beautiful research experiences, engaging scientific discussions, and a lot of fun. Thanks, Madhumita, for our dinners, study sessions, and impromptu trips. I cherish the memories we have created together. Thanks, Farhad, for the long, funny discussions in the corridor and common area. Thanks, Sakshi, for just being there for me. Thanks, Sushmita, for nourishing my soul with your culinary creations and thoughtful conversations during my visit to IMSc. I would also like to thank Megha and Neeraj for their consideration and friendly atmosphere. Thank you, Svein, for the last-minute help.

Finally, I would like to express my gratitude towards the Divine God, my grandparents, parents, in-laws, and siblings for their unconditional love, constant belief in my abilities and enduring support that helped me reach this significant milestone. Thanks, Abhishek, Kavita, Aakanksha, and Arvind, for providing affection in every possible way. Thanks, Aria, Vedica, Nyra, and Rudransh, for being a constant bundle of joy. Thanks, Nisha, Shubham, Nikita, and Karan, for our fun time during my visits home. Last but certainly not least, I express my profound appreciation to Jitu, my husband, for his love, understanding, and a listening ear whenever I needed it most. This academic pursuit would not have been possible without you. I am forever grateful for his presence in my life and his role in making this journey beautiful. This thesis represents not only my academic achievements but also a result of your love, affection and support. I dedicate this thesis to you all.

Abstract

The k-median clustering problem is one of the most well-studied clustering problems. In this problem, we are given a set $\mathbf{X}$ of n points in a space $\mathcal{M}$ with a distance measure function $\mathsf{dist} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_{\geq 0}$, a description of a set $\mathbf{F} \subseteq \mathcal{M}$ of possible centers, and an integer k, and the task is to find a pair $(\mathcal{X}, C)$, where $\mathcal{X}$ is a partition of $\mathbf{X}$ into k subsets $\{X_1, \ldots, X_k\}$ called clusters, and $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\} \subseteq \mathbf{F}$ is a set of k centers. Here, X_i is the cluster corresponding to the cluster center $\mathbf{c}_i \in \mathcal{C}$. The goal is to minimize the following cost over all pairs $(C, \mathcal{X})$

$$\mathsf{cost}(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}(\mathbf{c}_i, \mathbf{x}).$$

In this thesis, we give a multivariate analysis of the problem subject to various constraints on cluster's size, metric spaces and choice of centers. First, we systematically study exact algorithms for the k-median clustering problem in the case of general metric, where the candidate center set is either the same as the point set or is selected from a prescribed finite set given as an input. Further, we study the variant of the k-median problem known as the categorical k-median clustering problem where metric space is Σ^m for a finite alphabet Σ and dist is defined by the Hamming measure. In particular, we provide fixed-parameter algorithm for the variant of the problem with size constraints on the clusters. Finally, we consider the k-median clustering problem with an additional equal-size constraint on the clusters from the approximate parameterized preprocessing perspective. The result includes the first 2-approximate polynomial kernel for this problem parameterized by the cost of clustering in the ℓ_p-norm. We also complement this result by establishing lower bounds for the problem that eliminates the existence of an exact kernel of polynomial size and a Polynomial-Time Approximation Scheme.

Abstract in Norwegian

k-median klyngeproblemet er et av best studerte klyngeproblemene. I dette problemet er vi gitt et sett $\mathbf{X}$ av n punkt i et rom $\mathcal{M}$ med en avstandsfunksjon $\mathsf{dist} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_{\geq 0}$, en beskrivelse av en mengde $\mathbf{F} \subseteq \mathcal{M}$ av mulige sentre, og et heltall k, og oppgaven er å finne et par $(\mathcal{X}, C)$, der $\mathcal{X}$ er en partisjon av $\mathbf{X}$ i k delsett $\{X_1, \ldots, X_k\}$ kalt klynger, og $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\} \subseteq \mathbf{F}$ er et sett med k-sentre. Her er X_i klyngen som tilsvarer klyngesenteret $\mathbf{c}_i \in \mathcal{C}$. Målet er å minimere følgende kostnad over alle parene $(C, \mathcal{X})$

$$\mathsf{cost}(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}(\mathbf{c}_i, \mathbf{x}).$$

I denne oppgaven gir vi en multivariat-analyse av problemet underlagt ulike begrensninger på klyngens størrelse, metriske rom og valg av sentre. Først studerer vi systematisk eksakte algoritmer for k-median klynge-problemet med vilkårlig metrikk, der mengden av kandidater for senter enten er det samme som punktmengden eller velges fra en gitt, endelig mengde gitt som input. Videre studerer vi varianten av k-median-problemet kjent som det kategoriske k-median klyngeproblemet, der det metriske rommet er Σ^m for et endelig alfabet Σ og dist er definert ved Hamming-avstand. Spesielt gir vi parameteriserte algoritmer for varianten av problemet med begrensninger på størrelsen på klyngene. Til slutt ser vi på k-median klyngeproblemet med tilleggsbegrensningen at alle klyngene har lik størrelse, fra perspektivet av tilnærmelig parametrisert preprosessering. Resultater inkluderer den første 2-tilnærmede polynomiske kernelen for dette problemet parametrisert av kostnadene ved klynging i ℓ_p-normen. Vi utfyller også dette resultatet ved å etablere nedre grenser for problemet som viser at en eksakt kjerne av polynomisk størrelse og et tilnærmingsprogram i polynomisk tid ikke finnes.

List of Publications

The results included in the thesis have been published in the papers numbered with 1, 2, and 3. We refer to these papers in the thesis as the Article 1, Article 2 and Article 3. Note that the authors are listed in alphabetical order as is customary in Theoretical Computer Science.

1. Fedor V. Fomin, Petr A. Golovach, Tanmay Inamdar, **Nidhi Purohit**, Saket Saurabh. *Exact Exponential Algorithms for Clustering Problems.* The International Symposium on Parameterized and Exact Computation (IPEC). $13 : 1 - 13 : 14, 2022$.

2. Fedor V. Fomin, Petr A. Golovach, **Nidhi Purohit**. *Parameterized Complexity of Categorical Clustering with Size Constraints.* Journal of Computer and System Sciences (JCSS). $136 : 171 - 194, 2023$.

3. Sayan Bandyapadhyay, Fedor V. Fomin, Petr A. Golovach, **Nidhi Purohit**, Kirill Simonov. *Lossy Kernelization of Same-Size Clustering.* The Theory of Computing Systems (TOCS). $67 : 785 - 824, 2023$.

4. Sayan Bandyapadhyay, Fedor V. Fomin, Petr A. Golovach, William Lochet, **Nidhi Purohit**, Kirill Simonov. *How to Find a Good Explanation for Clustering?.* Artificial Intelligence, Volume $322 : 103948, 2023$.

5. Tanmay Inamdar, Lawqueen Kanesh, Madhumita Kundu, **Nidhi Purohit**, Saket Saurabh. *Fixed-Parameter Algorithms for Fair Hitting Set Problems.* Mathematical Foundations of Computer Science (MFCS). $55 : 1 - 55 : 14$. 2023.

6. Sayan Bandyapadhyay, Fedor V. Fomin, Petr A. Golovach, **Nidhi Purohit**, Kirill Simonov. *FPT Approximation for Fair Minimum-Load Clustering.* The International Symposium on Parameterized and Exact Computation (IPEC). $4 : 1 - 4 : 14, 2022$.

7. Avinandan Das, Lawqueen Kanesh, Jayakrishnan Madathil, Komal Muluk, **Nidhi Purohit**, Saket Saurabh. *On the complexity of singly connected vertex deletion.* Theoretical Computer Science (TCS). $934 : 47 - 64, 2022$.

8. Édouard Bonnet, **Nidhi Purohit**. *Metric Dimension Parameterized By Treewidth.* Algorithmica. $83(8) : 2606 - 2633, 2021$.

Contents

Chapter 1

Introduction

Data analysis with no prior knowledge is indispensable in understanding various phenomena. The data could be database records, graph nodes, texts, words, images, or any collection where a set of features describes individuals. One of the means to organise data is to classify or group them into subsets of similar objects known as *clusters*. For example, clustering is an unsupervised machine-learning tool that plays a significant role in analyzing data and making decisions [3, 13, 42].

The k-median clustering problem is one of the most fundamental and well-studied clustering problems [6, 17, 53, 63]. In its most general form, the problem is defined as follows. Given a set $\mathbf{X}$ of n points in a space $\mathcal{M}$ with a distance measure function $\mathsf{dist} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_{\geq 0}$, a description of a set $\mathbf{F} \subseteq \mathcal{M}$ of possible centers and an integer k, the task is to find a pair $(\mathcal{X}, C)$, where $\mathcal{X}$ is a partition of $\mathbf{X}$ into k subsets $\{X_1, \ldots, X_k\}$, called clusters, and $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\} \subseteq \mathbf{F}$ is a set of k centers. Here, X_i is the cluster corresponding to the cluster center $\mathbf{c}_i \in \mathcal{C}$. The goal is to minimize the following cost over all pairs $(C, \mathcal{X})$

$$\mathsf{cost}(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}(\mathbf{c}_i, \mathbf{x}).$$

When $\mathbf{F} = \mathcal{M}$, that is, we allow picking centers anywhere in the metric space, we call the problem k-Median Clustering. In the literature, this problem is also known as Continuous k-Median Clustering. For example, if $\mathcal{M}$ is a Euclidean space, one can pick any point in the space which is infinite as a potential center. When $\mathbf{F}$ is a finite set given as part of the input, we call the problem Discrete k-Median Clustering. This variant could be seen as a special case of k-Median Facility Location. Here each center corresponds to a facility to be built from the set of possible facility locations $\mathbf{F}$, and the input $\mathbf{X}$ of points corresponds to the set of clients that need to be served by these facilities. The cost of establishing a facility is zero, but we have

an upper bound on the number of facilities and one wishes to minimize the total cost of serving the clients. These problems have many applications in operational research and network design problems such as placing warehouses and hospitals [46]. When $\mathcal{M}$ is a d-dimensional Euclidean space $\mathbb{R}^d$ and dist is the Euclidean distance, then we call the problem Euclidean k-Median Clustering. The problem has gained much attention from the theory community [1, 9, 31, 60].

There is a large class of problems about learning from categorical data. The term categorical data refers to data type whose values are discrete and belong to a specific finite set of categories. It could be text, some numeric values, or even unstructured data like images. A prominent example of categorical data is binary data, where the points are binary vectors, each of whose coordinates can take a value of either 0 or 1. For example, in electronic commerce, each transaction can be modelled as a binary vector (known as market-based data), each of whose coordinates denotes whether a particular item is purchased or not [62, 83]. In document clustering, each document can be modelled as a binary vector, each of whose coordinates denotes whether a specific word is present or not in the document [62, 83]. For categorical data, Hamming distance is believed to be more useful. When $\mathcal{M}$ is the set of strings of length m over a finite alphabet Σ equipped with the Hamming distance, we call the problem Categorical k-Median Clustering.

Sometimes we have access to "some" information about the data, for example, class labels of the object, whether the two points must or cannot be placed together, the preference of the users about how the data must be grouped, or information about the minimum and maximum sizes of the clusters. The constraint clustering problem is an approach to cluster data while incorporating such domain knowledge (when available). In many applications of clustering, constraints come naturally. For example, the lower bound on the size of a cluster ensures certain anonymity of data and is often required for data privacy [74]. Moreover, the survey of Banerjee and Ghosh [12] contains various examples of clustering with balancing constraints in Direct Marketing [82], Category Management, Clustering of Documents [67], and Energy Aware Sensor Networks [44, 47] among others. It can be possible that the solution of unconstrained clustering algorithms is consistent with the given information. However, studying what happens if the answer is not aligned with existing knowledge is fascinating. We refer to the book by Basu et al. [13] for an overview. Here, we consider a variant of the constrained version of k-Median Clustering called Capacitated k-Median clustering, where the size of each cluster is specified, that is, required to lie within a given interval.

1.1 Known and Related Results

The problems are well-known to be NP-hard [32, 70]. Researchers have therefore investigated these in terms of approximation algorithms, and lots of work has been done on producing good approximation algorithms for these problems [7, 8, 15, 18, 19, 45, 54, 55, 61, 75].

There are other algorithmic paradigms to cope with the NP-hardness of the problem; one such is *parameterized complexity*. Here, the input comes with an additional parameter $k \in \mathbb{N}$, which describes some property of the input I and is believed to be small in practical applications. The aim is to restrict the exponential part of the running time to this parameter and have only a polynomial dependence on the input size $|I|$. A parameterized problem is said to be fixed-parameter tractable (FPT) if it admits an algorithm computing an optimal solution in time $f(k) \cdot |I|^{O(1)}$, where f is some computable function that depends solely on k, and the algorithm is correspondingly referred as an FPT algorithm. Naturally, the k-median problem is "multivariate" in the sense that in addition to the input size n, there are also parameters like the number of clusters k or the cost of clustering B and the dimension of space d. The choice of k as a parameter is very natural because, in many real-world applications, the problem requires a small number of clusters.

Over the years, the researchers studied the k-median clustering problem in the domain of approximation algorithms and parameterized complexity in parallel. It naturally gave rise to the study of the problem in the recently developed field of FPT-*approximation* where the above two paradigms are combined. This allowed for intriguing discoveries in the intersection of the two worlds. We refer to the survey by Feldmann et al. [35] for an overview of the area.

The complexity of the above problems heavily depends on the underlying metric space and the considered version of the k-median problem. Discrete k-Median Clustering in general metric space, when distance only needs to satisfy triangle inequality is known to be NP-hard for $k = 2$ [32] and for Euclidean norm even for the dimension $d = 2$ (k is large) [70]. The best-known approximation factor in polynomial time for the problem in the general metric is 2.6705 [22]. In the Euclidean metric, a 2.406-factor approximation is known [21]. In this result, the analysis heavily relies on the structure of the Euclidean space. It is therefore not believed to extendable to any other metric space. Moreover, for both of the above metric spaces, the factor cannot be approximated better than $(1 + \frac{2}{e})$ [1] unless P $\neq$ NP, that is, the best lower bound is still the

[1] Here, the value of e is 2.71828. It is also known as Euler's number.

$(1+\frac{2}{e})$-hardness from Guha and Kuller [45]. For the algorithmic designer, the *continuous* version of the k-median problem appears computationally more manageable than the discrete case, as it allows to place centers anywhere in the metric space. In the Euclidean metric space, it is shown that an α-approximation to the discrete case can be used to obtain $(1+\varepsilon)\cdot\alpha$-approximation for the continuous case under the Euclidean distance for any $\varepsilon > 0$ [69]. However, in the general metric space, Continuous k-Median Clustering admits a 2-factor approximation and it is NP-hard to approximate up to a factor of 2–$o(1)$ [24] improving the inapproximability $(1+\frac{1}{e})$-factor derived from the approach in [45]. Further, we know that the upper and lower bound for the discrete version are tight even if "more" time is allowed [23]. The polynomial time approximation schemes, that is, algorithms finding solutions very close to the optimal are known for the Euclidean k-Median Clustering when d is a constant [6, 25, 58].

The next question is whether we can do better than the approximation results mentioned above if we have more resources, that is, when we allow a running time of $f(k)\cdot n^{\mathcal{O}(1)}$ (i.e. in the fixed-parameter tractability setting) for arbitrary computable functions f. The reduction by Guha and Kuller [45] showed that in the general metric space Discrete k-Median Clustering is $W[2]$-hard when parameterized by k. In other words, it is unlikely to be solvable optimally in FPT time when parameterized by the number of clusters in the solution. For Euclidean space and $d = 2$, Cohen-Addad et al. in [20] showed that there does not exist a $n^{\mathcal{O}(\sqrt{k})}$-time algorithm unless the Exponential Time Hypothesis (ETH) fails. The authors in the same paper [20] showed that the problem is even harder when $d \geq 4$. That is, unless the ETH fails, there is no $f(k)\cdot n^{o(k)}$-time algorithm for any computable function f solving Euclidean k-Median Clustering strictly in the settings where the set of potential candidate centers is explicitly given as input. Moreover, approximating Discrete k-Median Clustering in FPT time when parameterized by k is studied by Cohen-Addad et al. [23]. In their paper, the authors give an FPT-time algorithm with approximation factor $(1+\frac{2}{e})$. However, in the same paper, the authors showed that even after allowing ourselves FPT time, one can not achieve a better approximation factor than $(1+\frac{2}{e})$ (assuming standard complexity-theoretic conjectures) concluding that in the setting of FPT, the upper and lower bounds are tight.

Coreset constructions is one of the essential advances in FPT-approximation concerning clustering problems. It is an approach for data compression for obtaining FPT-approximation for clustering. The notion of coresets originated from computational geometry. In the language of parameterized complexity, a coreset is essentially an approximate kernel. Informally, a coreset summarises the data that for every set of k centers, approximately (within $(1\pm\varepsilon)$ factor) preserves the optimal clustering cost. Feng et al. in [36] gave a unified framework to design FPT approximation algorithms for clus-

tering problems. Har-Peled and Mazumdar gave a $(1+\varepsilon)$-approximation algorithm for the k-median clustering problem using coreset constructions [48]. After a series of interesting works, the best-known upper bound on coreset size in general metric space is $\mathcal{O}((k\log n)/\varepsilon^2)$ [33] and the lower bound is known to be $\Omega((k\log n)/\varepsilon)$ [10].

For the Euclidean space of dimension d, it is possible to construct coresets of size $(k/\varepsilon)^{\mathcal{O}(1)}$ [34, 76]. Remarkably, the size of the coresets does not depend on n and d in this case. Hence, they can be used to obtain an $(1+\varepsilon)$-factor approximate scheme parameterized by k in time $f(k,\varepsilon)\cdot nd$. One can obtain an approximation scheme by enumerating all possible partitions of the coreset points into k parts, evaluating the cost of each of them and outputting the one of minimum cost.

Another important tool for constructing an FPT randomized algorithm is *sampling*. Kumar et al. gave a $(1+\varepsilon)$-approximation scheme in $f(k,\varepsilon)\cdot nd$-time, with an exponential dependence on k [60]. When the dimension d is arbitrary, one can obtain a $(1+\varepsilon)$-approximation in FPT time when parameterized by k where the dependency on n and d are only linear. We give a brief explanation of this result in Section 1.2.

Feige in [32] proved that the Categorical k-Median Clustering (for binary points) is NP-hard for $k=2$. However, in the case of categorical data, we have more possibilities for parameterization. In particular, it makes sense to consider parameterization by the budget B. In the domain of parameterized algorithms, Fomin, Golovach, and Panolan [37] gave two parameterized algorithms for the binary case of Categorical k-Median Clustering with running times $2^{\mathcal{O}(B\log B)}\cdot(nm)^{\mathcal{O}(1)}$ and $2^{\mathcal{O}(\sqrt{kB\log(k+B)\log k})}\cdot(nm)^{\mathcal{O}(1)}$, respectively. Fomin, Golovach and Simonov in [38] studied k-clusterings with various distance norms in the categorical clustering problem. They showed that the problem is W[1]-hard parameterized by $d+B$ under the ℓ_0-norm (but the size of the alphabet Σ is unbounded), where d is the dimension of input points and B is the cost of clustering. They also showed that for the ℓ_p norm, the problem admits FPT algorithms if $0<p\leq 1$ or $p=2$, and the problem is W[1]-hard for $p=\infty$, when parameterized by B.

The capacitated variants of the k-median clustering problem are generally more difficult. In particular, all hardness results of uncapacitated hold for the capacitated variant of the k-median problem. On the positive side, Capacitated k-Median clustering admits a $\mathcal{O}(\log(k))$ approximation in general metric space and high dimensional Euclidean space [17] . From the negative side, similar to the uncapacitated variant of the problem, it is hard to obtain an approximation factor better than $(1+\frac{2}{e})$ [45]. For the capacitated k-median clustering problem, Cohen-Addad et al. gave a $(3+\varepsilon)$-approximation scheme in general metric with general capacities using coresets constructions [26]. The

same paper also showed a $(1+\varepsilon)$-approximation for the Euclidean metrics of arbitrary dimensions, surprisingly obtaining a better approximation factor than $(1+\frac{2}{e})$. However, in general metric spaces, obtaining an FPT approximation algorithm for the uncapacitated k-median clustering problem with approximation guarantee less than $(1+\frac{2}{e})$ is impossible assuming the GAP-ETH [2] [23].

1.2 Our Results

The k-median clustering problem received much attention in terms of Euclidean metric space setting and parameterization by k [1, 9, 33, 60]. However, for Discrete k-Median Clustering the brute force approach of trying all the possible subsets of centers was the best exact algorithm (with running time $\binom{n}{k} \cdot n^{\mathcal{O}(1)}$) known in general metric space. The algorithm enumerates all sets of centers of size k, and the corresponding partition of $\mathbf{X}$ into clusters is obtained by assigning each point to its nearest center. Then, we simply return the solution with the minimum cost. However, note that when k belongs to the range $n/2 \pm o(n)$, then $\binom{n}{k} \simeq 2^n$. Thus, the naïve algorithm has running time $\mathcal{O}^*(2^n)$ [3] in the worst case. Hence, we ask whether the discrete k-median clustering in general metric space admit moderately exponential-time algorithms, i.e., algorithms with running time $c^n \cdot n^{\mathcal{O}(1)}$ for a constant $c < 2$.

In the first part of the thesis, we study exact algorithms for the discrete k-median clustering problem in general metric spaces. In particular, we study the two variants of Discrete k-Median Clustering. First, when $\mathbf{F} = \mathbf{X}$, that is, any point in the set of input points $\mathbf{X}$ is allowed to be picked as a center, we give an exact algorithm running in $\mathcal{O}(1.89)^n$-time, where n is the number of input points. Second, we consider the case where the set of candidate centers $\mathbf{F}$ is a finite set given as part of the input and distinct from the input set $\mathbf{X}$. We provide an algorithm with running $2^n \cdot (mn)^{\mathcal{O}(1)}$-time which solves the problem exactly, where n is the number of input points and m is the number of candidate centers. We also complement both results by showing that the running time of the algorithm is asymptotically optimal upto the base of the exponent. The results appeared in Article 1.

Motivated by Cohen-Addad et al. [20], Fomin et al. studied Categorical k-Median Clustering, mainly when the input points are binary and with Hamming distance [37]. They proved that the problem is fixed-parameter tractable by giving an algorithm that

[2] The GAP-ETH states that no $2^{o(n)}$-time algorithm can distinguish between a satisfiable 3-CNF formula and a 3-CNF formula in which each assignment satisfies at most $(1-\varepsilon)$ fraction of all clauses for some constant $\varepsilon > 0$.

[3] $\mathcal{O}^*(\cdot)$ hides polynomial factors in the instance size

solves the problem in time $f(B) \cdot |I|^{\mathcal{O}(1)}$, where B is the cost of clustering and $|I|$ is the size of the input instance. The natural question is to analyze the complexity of the problem on a significantly more general model of the capacitated clustering, where the sizes of the clusters should satisfy certain constraints. More precisely, in Categorical Capacitated k-Median Clustering, we are given two non-negative integers p and q and seek a k-median clustering with each cluster's size between the given numbers p and q.

In the next part of the thesis, we conclude that these additional constraints do not impact the problem's parameterized complexity. We give an algorithm that solves the problem in $f(B) \cdot |\Sigma|^B \cdot |I|^{\mathcal{O}(1)}$ time. Hence, the problem is fixed-parameter tractable with respect to the combined parameter $B + |\Sigma|$. In some applications, the cluster size is approximately equal; see, e.g. [78]. We consider two variants of Categorical Capacitated k-Median Clustering. In the first variant, the input consists of a set of points, positive integers k and B, and a non-negative integer δ. The task is to find a k-median clustering of cost at most B such that the sizes of the resulting clusters should differ by at most δ, we call the problem Balanced Categorical k-Median Clustering. In the second variant, we are given a set of points, positive integers k and B, and a real $\alpha \geq 1$, and the goal is to obtain a k-median clustering of cost at most B such that the ratio of the resulting cluster's sizes is upper bound by α, we call the problem Factor-Balanced Categorical k-Median Clustering. The NP-hardness and fixed-parameter tractability parameterized by $B + |\Sigma|$ of both the problems follow the hardness and FPT results for Categorical Capacitated k-Median Clustering. Moreover, we show that Balanced Categorical k-Median Clustering admits a polynomial kernel with respect to the combined parameter $k + B + \delta$. However, for the binary case, we conclude that unless some complexity theoretic hypothesis fails, neither of the considered problems admits a polynomial kernel parameterized by B. The results appeared in Article 2.

In many real-life scenarios, it is desirable to cluster data into clusters of exactly equal sizes. For example, to tailor teaching methods to meet the specific needs of various students, one would be interested in allocating k fair class sizes by grouping students with homogeneous abilities and skills [49]. In scheduling, the standard task is to distribute n jobs to k machines while keeping identical workloads on each machine and simultaneously reducing the configuration time. In designing a conference program, one might be interested in allocating n scientific papers according to their similarities to k "balanced" sessions [78].

The next part of the thesis is an attempt to capture such scenarios. Towards this, we study a variant of Capacitated k-Median clustering, where the input points are

in $\mathbb{Z}^d$, dist is the ℓ_p-norm for $p \geq 0$, and the goal is to find a k-median clustering of cost at most B such that the size of each cluster is equal. We call this problem ℓ_p-Equal k-Median Clustering. The results appeared in Article 3. We study the parameterized complexity of ℓ_p-Equal k-Median Clustering when parameterized by the cost of clustering B.

Before stating our results, let us first discuss some limitations and advantages of parameterization of the problem by the budget B. We believe that restricting the input to integral values is the most natural model for studying the complexity of the problem with respect to the parameter B. Moreover, considering B as a parameter only makes sense when input values are suitably discretized and not scaleable, which is quite common when the data is categorical. The most drastic effect of compression occurs when B is small. Intuitively, this means that many of the data points are the same. Such a condition is common in handling personal data that cannot be re-identified. For example, the k-anonymity property requires each person in the data set to be undistinguishable from at least k individuals whose information appears in the release [77].

ℓ_p-Equal k-Median Clustering is known to be NP-hard and, moreover, when it comes to approximation in polynomial time, we show that it is NP-hard to obtain a $(1+\varepsilon)$-approximation with ℓ_0 (or ℓ_1) distances for some $\varepsilon > 0$. However, parameterized by k and ε, standard techniques yield $(1+\varepsilon)$-approximation in FPT time. For the ℓ_2 norm, there is a general framework by Ding and Xu [31] for designing algorithms for the k-median clustering problem with an additional constraints on cluster sizes. The best-known improvements by Bhattacharya et al. [14] achieve a running time of $2^{\widetilde{\mathcal{O}}(k/\varepsilon^{\mathcal{O}(1)})} \cdot n^{\mathcal{O}(1)}d$ in the case of ℓ_2-Equal k-Median Clustering, where $\widetilde{\mathcal{O}}$ hides polylogarithmic factors.

A seminal work of Kumar et al. [60] achieves a $(1+\varepsilon)$-approximation for ℓ_2-Equal k-Median Clustering with the similar running time of $2^{\widetilde{\mathcal{O}}(k/\varepsilon^{\mathcal{O}(1)})} \cdot nd$. The algorithm proceeds as follows. First, take a small uniform sample of the input points, and by guessing assure that the sample is taken only from the largest cluster. Second, estimate the optimal center of this cluster from the sample. In the case of the equal k-median clustering problem, Theorem 5.4 of Kumar et al. [60] guarantees that from a sample of size $(1/\varepsilon)^{\mathcal{O}(1)}$ one can compute in time $2^{(1/\varepsilon)^{\mathcal{O}(1)}} \cdot d$ a set of candidate centers such that at least one of them provides a $(1+\varepsilon)$-approximation to the cost of the cluster. Finally, "prune" the set of points so that the next largest cluster contains at least a $\Omega(1/k)$ fraction of the remaining points and continue the same process with one less cluster. One can observe that in the case of ℓ_2-Equal k-Median Clustering, a simplification of the above algorithm suffices. One does not need to perform the "pruning" step, as we are only interested in clusterings where all the clusters have size exactly n/k. Thus,

$(1/\varepsilon)^{\mathcal{O}(1)}$-sized uniform samples from each of the clusters can be computed immediately in total time $2^{\tilde{\mathcal{O}}(k/\varepsilon^{\mathcal{O}(1)})} \cdot nd$. This achieves $(1+\varepsilon)$-approximation for ℓ_2-Equal k-Median Clustering with the same running time as the algorithm of Kumar et al. [60]. In fact, the same procedure works for the ℓ_0 norm as well, where for estimating the cluster center it suffices to compute the optimal center of a sample of size $\mathcal{O}(1/\varepsilon^2)$ as proven by Alon and Sudakov [4].

In another line of work, FPT-time approximation is achieved via constructing small-sized coresets of the input points. The work of Bandyapadhyay et al. guarantees an ε-coreset for ℓ_2-Equal k-Median Clustering of size $(kd \log n/\varepsilon)^{\mathcal{O}(1)}$, and consequently a $(1+\varepsilon)$-approximation algorithm with running time $2^{\tilde{\mathcal{O}}(k/\varepsilon^{\mathcal{O}(1)})}(nd)^{\mathcal{O}(1)}$ [11] . Thus, in terms of an FPT approximation, ℓ_2-Equal k-Median Clustering is surprisingly "simpler" than its unconstrained variant k-Median Clustering. However, our hardness result shows that the problems are similarly hard in terms of polynomial-time approximation.

An another successful attempt of combining kernelization with approximation algorithms is *lossy kernelization*. This notion was introduced by Lokshtanov et al. [65]. Informally, in lossy kernelization, given an instance of the problem and a parameter, we would like the kernelization algorithm to output a reduced instance of size polynomial in the parameter. However, the notion of equivalence is relaxed in the following way. Given a c-approximate solution (i.e., one with the cost within c-factor of the optimal cost) to the reduced instance, it should be possible in polynomial time to find an αc-approximate solution to the original instance. The factor α is the loss occured while going from the reduced instance to the original instance. Lossy kernels and coresets have a lot of similarities in the sense that both compress the space compared to the original data, and any algorithm applied on a coreset or kernel to efficiently retrieve a solution with a gurantee almost the same as the one provided by the algorithm on the original input. The crucial difference is that coreset constructions result in a small set of weighted points. The weights could be as large as the input size n. Thus, a coreset of size polynomial in k/ε, is not a polynomial-size lossy kernel for parameters k, ε because of the $\log n$ bits required to encode the weights. Moreover, usually coreset constructions do not bound the number of coordinates or the dimension of the points.

While the notion of lossy kernelization proved to be useful in the design of graph algorithms, we are not aware of its applicability in clustering. This brings us to the following question: *What can lossy kernelization offer to clustering?* We make the first step towards the development of lossy kernels for clustering problems. In particular, we study the Parameterized optimization version of ℓ_p-Equal k-Median Clustering, parameterized by the cost B of clustering. We show that the problem admits a 2-factor approximate polynomial kernel. The natural question is whether the factor is optimal,

unfortunately, we do not have an answer to it. However, we complement the result by establishing the lower bounds for the problem that eliminate the existence of exact kernel of polynomial size.

1.3 Overview of the Thesis

The thesis is organized as follows. In Chapter 2, we define common notation used repeatedly throughout the thesis and discuss some standard background in algorithmic complexity. In Chapter 3, we define our models and give a common result relevant for the remaining chapters. In Chapter 4, we show the exact algorithms for the variants of Discrete k-Median Clustering (see Article 1). In Chapter 5, we study the parameterized complexity of Categorical Capacitated k-Median Clustering, Balanced Categorical k-Median Clustering, and Factor-Balanced Categorical k-Median Clustering (see Article 2). In Chapter 6, we show a approximate kernel for Parameterized ℓ_p-Equal k-Median Clustering (see Article 3). Finally, we conclude with discussions and the future research directions in Chapter 7.

Chapter 2

Basic Notions

2.1 Numbers

We denote the set of real numbers by $\mathbb{R}$, the set of integers by $\mathbb{Z}$, and the set of natural numbers by $\mathbb{N}$. We denote the set of nonnegative real numbers by $\mathbb{R}_{\geq 0}$, and by $\mathbb{R}_{>0}$ the set of positive real numbers. Respectively, $\mathbb{Z}_{\geq 0}$ denotes the set of nonnegative integers, that is, $\mathbb{Z}_{\geq 0} = \mathbb{N} \cup \{0\}$. For a vector $\mathbf{x} \in \mathbb{R}^d$, we use $\mathbf{x}[i]$ to denote the i-th element of the vector for $i \in \{1, \ldots, d\}$.

2.2 Metric Spaces

A *metric space* is a pair $(\mathcal{M}, \mathsf{dist})$, where $\mathcal{M}$ is a set of points and $\mathsf{dist} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_{\geq 0}$ is a distance measure function which is a metric, that is, for any $\mathbf{x}, \mathbf{y} \in \mathcal{M}$, it satisfies the following three properties:

1. $\mathsf{dist}(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$,
2. $\mathsf{dist}(\mathbf{x}, \mathbf{y}) = \mathsf{dist}(\mathbf{y}, \mathbf{x})$,
3. triangle inequality, that is, for any point $\mathbf{z} \in \mathcal{M}$

$$\mathsf{dist}(\mathbf{x}, \mathbf{y}) \leq \mathsf{dist}(\mathbf{x}, \mathbf{z}) + \mathsf{dist}(\mathbf{z}, \mathbf{y}).$$

The points in set $\mathcal{M}$ can be points in $\mathbb{R}^d$. For $p \geq 1$, the l_p-norm defines the distance

between two points $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ as

$$\mathsf{dist}(\mathbf{x}, \mathbf{y}) = \mathsf{dist}_p(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_p = (\sum_{i=1}^{d} |\mathbf{x}[i] - \mathbf{y}[i]|^p)^{\frac{1}{p}}.$$

For $p = 1$, $\mathsf{dist}_1(\mathbf{x}, \mathbf{y})$, also known as the Manhattan distance or taxicab distance, is

$$\mathsf{dist}_1(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_1 = \sum_{i=1}^{d} |\mathbf{x}[i] - \mathbf{y}[i]|.$$

The l_2-norm is the regular Euclidean distance. For $p = 0$, $\mathsf{dist}_o(\mathbf{x}, \mathbf{y})$ is the number of indices at which vector $\mathbf{x}$ and $\mathbf{y}$ differ, also called Hamming distance, and for $p = \infty$,

$$\mathsf{dist}_\infty(\mathbf{x}, \mathbf{y}) = \max_{i \in \{1,\ldots,d\}} |\mathbf{x}[i] - \mathbf{y}[i]|.$$

It is also possible to consider problems for other metrics spaces, for example, for the *graph metric*. Here, the points of the metric space is the set V of vertices of an undirected edge-weighted connected graph G and the distance between vertices u and v is the length of the shortest path connecting them.

2.3 Complexity Theory

In this section, we recall some basic definitions regarding the approximation algorithms and discuss parameterized complexity.

A *decision problem* L is a subset of Σ^*, where Σ^* is a set of strings over a finite alphabet Σ. The input of a decision problem L is a string I over Σ, and the instance is yes/no depending on whether $I \in L$ or not. A *minimization* problem Π is a computable function $\Pi \colon \Sigma^* \times \Sigma^* \to \mathbb{R}_{\geq 0}$. The instances of a minimization problem Π is $I \in \Sigma^*$, and a solution to I is simply a string $s \in \Sigma^*$. Then the function $\Pi(\cdot, \cdot)$ defines the *value* $\Pi(I, s)$ of a solution s to an instance I. The optimum value of an instance I is $\mathsf{Opt}_\Pi(I) = \min_{s \in \Sigma^*} \Pi(I, s)$. A solution s is *optimal* if $\mathsf{Opt}_\Pi(I) = \Pi(I, s)$.

2.3.1 Approximation Algorithms

An NP-hardness optimization problem implies that no known algorithm that can solve "all the instances" of problem "optimally" in "polynomial time". To deal with the NP-hardness of a problem, one of the above requirements must be relaxed. One of the

common approaches is to relax the requirement of the optimal solution and settles for a solution "closer" to the optimal. Towards, this we consider designing approximation algorithms for the optimization problems to obtain near-optimal solutions as opposed to a classical exact algorithm which has to correctly determine whether an input is a yes or no instance of a decision problem. In this section we introduce the standard terminology for approximation algorithms. For an in-depth introduction to the area of approximation algorithms, we refer to the classical book by Vazirani [79], and by Williamson and Shymos [80]. Formally, for a minimization problem Π and a value $\alpha > 1$, an α-approximate algorithm is a polynomial time algorithm which for all the instances of the problem produces a solution whose objective value is at most a factor of α times of the value of the optimal, i.e, $\alpha \cdot \mathsf{Opt}$, where Opt is the objective value of the optimal solution. The complexity class that contains problems admitting constant-factor approximation algorithms is called APX. A problem is said to have a *polynomial-time approximation scheme* (PTAS) if for every fixed $\epsilon > 0$, there is a polynomial-time algorithm solving the problem and producing a solution $(1 + \epsilon)$ factor to the optimum. Formally, we define it as follows.

For a minimization problem Π, a *polynomial-time approximation scheme* (PTAS) is a family of algorithms $\{\mathcal{A}_\epsilon\}$, where there is an algorithm for each $\epsilon > 0$, such that $\{\mathcal{A}_\epsilon\}$ is an $(1 + \epsilon)$-approximation algorithm.

The running time of a PTAS algorithm is of the form $|I|^{f(\epsilon)}$, where I is an input instance and f is some function of ϵ. Unless $\mathsf{P} = \mathsf{NP}$, $\mathsf{PTAS} \subsetneq \mathsf{APX}$, that is, there exist problems that are in APX but without a PTAS. An equivalent of NP-hardness for approximation algorithms is APX-hardness. Therefore, showing APX-hardness would imply the non-existence of a PTAS. The APX-hardness of a problem can be shown by giving a PTAS reduction from some known APX-hard problem.

For the purpose of this work it would be suffice to know that 3-Dimensional Matching (3DM) is APX-hard. In (3DM), we are given three disjoint sets of elements X, Y and Z such that $|X| = |Y| = |Z| = n$ and a set of m triples $T \subseteq X \times Y \times Z$. In addition, each element of $W := X \cup Y \cup Z$ appears in at most 3 triples. A set $M \subseteq T$ is called a matching if no element of W is contained in more than one triple of M. The goal is to find a maximum cardinality matching. We use the following proposition due to Petrank [73].

Proposition 1. *[Restatement of Theorem 4.4 from [73]] There exists a constant* $0 < \gamma < 1$*, such that it is* NP*-hard to distinguish the instances of the* 3DM *problem in which a perfect matching exists, from the instances in which there is a matching of size at most* $(1 - \gamma)n$*.*

2.3.2 Parameterized Complexity

A *parameterized problem* Π is a subset of $\Sigma^* \times \mathbb{N}$, where Σ is a finite alphabet. Thus, an instance of Π is a pair (I, k), where $I \subseteq \Sigma^*$ and k is a nonnegative integer called a *parameter*. It is said that a parameterized problem Π is *fixed-parameter tractable* (FPT) if it can be solved in $f(k) \cdot |I|^{\mathcal{O}(1)}$ time for some computable function f that depends on the parameter k only. The parameterized complexity class FPT is composed of fixed-parameter tractable problems. The complexity class XP consists of problems that can be solved with running time $|I|^{f(k)}$, where f is some function of k.

We now discuss the kernelization algorithms and polynomial kernels, and ways for proving lower bounds for kernelization.

Kernelization. A *kernelization* algorithm (or *kernel*) for a parameterized problem Π is an algorithm $\mathcal{A}$ that, given an instance (I, k) of Π, in polynomial time produces an instance (I', k') of Π such that

(i) $(I, k) \in \Pi$ if and only if $(I', k') \in \Pi$, and

(ii) $|I'| + k' \leq g(k)$ for a computable function $g(\cdot)$.

The function $g(\cdot)$ is called the *size* of a kernel; a kernel is *polynomial* if $g(\cdot)$ is a polynomial. Every decidable FPT problem admits a kernel. However, it is unlikely that all FPT problems have polynomial kernels and the parameterized complexity theory provides tools for refuting the existence of polynomial kernels up to some reasonable complexity assumptions. The standard assumption here is that $\mathsf{NP} \not\subseteq \mathsf{coNP} / \mathsf{poly}$. In this work, we use a type of reduction for deriving kernelization lower bound called *polynomial parameter transformation* (PPT). Here, we establish a kernelization lower bound of some problems by showing a PPT reduction from an already known hard problem.

Let $\Pi, \Pi' \subseteq \Sigma^* \times \mathbb{N}$ be two parameterized problems. An algorithm $\mathcal{A}$ is called a *polynomial parameter transformation* if, given an instance (I, k) of problem Π, $\mathcal{A}$ works in polynomial time and outputs an equivalent instance $(I', k') \in \Pi'$, such that $k' \leq p(k)$ for some polynomial $p(\cdot)$.

Note, we have no constraints on the size of I', and only the polynomial bound on the parameter k' is essential. In this work, we use the result of Dell and Marx [30] about kernelization lower bounds for the Perfect r-Set Matching problem. A hypergraph $\mathcal{H}$ is said to be *r-uniform* for a positive integer r, if every hyperedge of $\mathcal{H}$ has size

r. Similarly to graphs, a set of hyperedges M is a *matching* if the hyperedges in M are pairwise disjoint, and M is *perfect* if every vertex of $\mathcal{H}$ is *saturated* in M, that is, included in one of the hyperedges of M. Perfect r-Set Matching asks, given an r-uniform hypergraph $\mathcal{H}$, whether $\mathcal{H}$ has a perfect matching. Dell and Marx [30] proved the following kernelization lower bound.

Proposition 2. *[[30]] Let $r \geq 3$ be an integer and let ε be a positive real. If $\mathsf{NP} \subseteq \mathsf{coNP} / \mathsf{poly}$, then* Perfect r-Set Matching *does not have kernels with $\mathcal{O}(\left(\frac{|V(\mathcal{H})|}{r}\right)^{r-\varepsilon})$ hyperedges.*

We need a weaker claim.

Corollary 1. Perfect r-Set Matching *admits no polynomial kernel when parameterized by the number of vertices of the input hypergraph unless $\mathsf{NP} \subseteq \mathsf{coNP} / \mathsf{poly}$.*

2.3.3 Lower Bounds

In this section, we discuss the hardness assumptions commonly used to show the lower bounds such as W-hierarchy, Exponential Time Hypothesis and Set Cover Conjecture.

W-Hierarchy. Parameterized complexity theory provides a framework to refute the existence of an FPT algorithm for a problem, that is, it gives some evidence that a specific problem is not fixed-parameter tractable. Unlike the NP-complete problems, there is a hierarchy of hard parameterized problems occupying the different levels of this hierarchy, called W-hierarchy. Downey and Fellows introduced the W-hierarchy in an attempt to capture the exact complexity of various hard parameterized problems. We omit the formal details here and refer to the book [29]. The following relation is known among the classes in W-hierarchy: $\mathsf{FPT} = W[0] \subseteq W[1] \subseteq W[2] \cdots \subseteq W[P]$. Similar to $\mathsf{P} \neq \mathsf{NP}$, it is widely believed that $\mathsf{FPT} \neq W[1]$, and used as a working hypothesis of parameterized complexity. Thus, if for any $i \geq 1$, a parameterized problem is $W[i]$-hard, then is unlikely to be fixed parameter tractable. The parameterized hardness of a problem can be shown by giving a *parameterized reduction* from a known $W[i]$-hard problem that transfers fixed-parameter tractability.

Let $A, B \subseteq \Sigma^* \times \mathbb{N}$ be two parameterized problems. A *parameterized reduction* from A to B is an algorithm that, given an instance (I, k) of A outputs an instance (I', k') of B such that

- (I, k) is a yes-instance of A if and only if (I', k') is a yes-instance of B,

- $k' \leq g(k)$ for some computable function g and
- the running time is $f(k) \cdot |I|^{\mathcal{O}(1)}$ for some computable function f.

Exponential Time Hypothesis. Recall that, in the CNF-SAT problem, we are given a propositional Boolean formula $\phi = C_1 \wedge \ldots \wedge C_m$ over n variables $X = \{x_1, x_2, ..., x_n\}$ such that each clause C_i is a disjunction of literals of the form x_i or $\neg x_i$, for some $1 \leq i \leq n$. The task is to determine whether formula has a satisfying assignment, that is, an assignment of true/false values to the variables so that formula ϕ becomes true. By the famous Cook-Levin Theorem [27], CNF-SAT is NP-hard, that is, we do not expect it to be solvable in polynomial time. However it can be solved in time $\mathcal{O}^*(2^n)$ by trying all possible true/false assignments. For this classical problem, we do not know any faster algorithm than this brute force. For a positive integer q, a q-CNF formula is a special case of CNF-SAT, where each clause C_i is a disjunction of at most q literals. In 2001, Impagliazzo, Paturi, and Zane [50] introduced a conjecture which provides a tight understanding of the complexity of q-SAT, for $q \geq 3$ known as Exponential Time hypothesis (ETH) which is defined as follows.

Conjecture 1 (Exponential Time Hypothesis (ETH))**.** *There is a positive real number δ such that 3-SAT with n variables and m clauses can not be solved in time $2^{\delta n}(n+m)^{\mathcal{O}(1)}$.*

Note that ETH is a stronger assumption than P $\neq$ NP. This conjecture also implies that FPT $\neq W[1]$ [29]. Hence, it can also give conditional evidence that certain problems are not fixed-parameter tractable. Also, it can be used to argue that a parameterized problem can not be solved within a running time of $2^{o(k)} \cdot |I|^{\mathcal{O}(1)}$ or $f(k) \cdot |I|^{o(k)}$. For an in-depth study to the topics we refer to Chapter 14 of [29].

The main usage in this work is through the following proposition due to [64].

Proposition 3. *Assuming ETH, there is no $2^{o(n)}$ time algorithm for the* Dominating set *problem where n is the number of vertices of G .*

Note, that by Proposition 3, any polynomial time reduction from Dominating set to an another problem whose size is linear in n shows that the latter does not have an subexponential algorithm. That is, such reductions provide an algorithmic ETH lower bounds for the target problem.

Set Cover Conjecture In the Set Cover problem, the input is a ground set of n elements and a collection of m sets, and the goal is to find the smallest sub-collection

of sets whose union is the entire ground set. An exhaustive search takes $\mathcal{O}(2^m n)$ time, and a dynamic-programming algorithm has runtime $\mathcal{O}(2^n \cdot mn)$ [40]. The Set Cover Conjecture implies a $2^{\Omega(n)}$ lower bound for Set Cover even when size of each set is $\mathcal{O}(1)$. Note that no algorithm that runs in time $\mathcal{O}^*(2^{(1-\epsilon)n})$ is known, where $\epsilon > 0$ denotes a fixed constant. Further, it was conjectured using the set cover conjecture that the above runtime is optimal, even if the input sets are small [28]. To state Set Cover Conjecture, we consider a variant of the Set Cover problem called Δ-Set Cover where all the sets have size at most $\Delta > 0$, and the conjecture is defined as follows.

Conjecture 2 (Set Cover Conjecture (SeCoCo) [28])**.** *For every fixed $\epsilon > 0$ there is $\Delta(\epsilon) > 0$, such that no algorithm (even randomized) solves Δ-Set Cover in time $\mathcal{O}^*(2^{(1-\epsilon)n})$.*

2.3.4 Parameterized Approximation and Lossy Kernels

We also consider the parameterized analog of optimization problems. Since we only deal with minimization problems where the minimized value is nonnegative, we state the definitions only for optimization problems of this type. A *parameterized minimization* problem Π is a computable function

$$\Pi\colon \Sigma^* \times \mathbb{N} \times \Sigma^* \to \mathbb{R}_{\geq 0} \cup \{+\infty\}.$$

The instances of a parameterized minimization problem Π are pairs $(I, k) \in \Sigma^* \times \mathbb{N}$, and a solution to (I, k) is simply a string $s \in \Sigma^*$, such that $|s| \leq |I| + k$. Then the function $\Pi(\cdot, \cdot, \cdot)$ defines the *value* $\Pi(I, k, s)$ of a solution s to an instance (I, k). The optimum value of an instance (I, k) is

$$\mathsf{Opt}_\Pi(I, k) = \min_{s \in \Sigma^* \text{ s.t. } |s| \leq |I|+k} \Pi(I, k, s).$$

A solution s is *optimal* if $\mathsf{Opt}_\Pi(I, k) = \Pi(I, k, s)$.

A parameterized minimization problem Π is said to be FPT if there is an algorithm that for each instance (I, k) of Π computes an optimal solution s in $f(k) \cdot |I|^{\mathcal{O}(1)}$ time, where $f(\cdot)$ is a computable function. Let $\alpha \geq 1$ be a real number.

An FPT *α-approximation algorithm* for Π is an algorithm that in $f(k) \cdot |I|^{\mathcal{O}(1)}$ time computes a solution s for (I, k) such that $\Pi(I, k, s) \leq \alpha \cdot \mathsf{Opt}_\Pi(I, k)$, where $f(\cdot)$ is a computable function.

Note that the above definition only defines constant factor FPT-approximation algo-

rithms. However, the definition can in a natural way be extended to approximation algorithms whose approximation ratio depeneds on the parameter k, on the instance I, or on both.

It is useful for us to make some comments about defining $\Pi(\cdot,\cdot,\cdot)$ for the case when the considered problem is parameterized by the solution value. For simplicity, we do it informally and refer to [41] for details and explanations. If s is not a "feasible" solution to an instance (I,k), then it is convenient to assume that $\Pi(I,k,s)=+\infty$. Otherwise, if s is "feasible" but its value is at least $k+1$, we set $\Pi(I,k,s)=k+1$.

Lossy Kernelization Similar to kernels for parameterized problems, we define an extension of kernelization to optimization problem, call α-*approximate* or *lossy* kernels. Informally, an α-approximate kernel of size $g(\cdot)$ is a polynomial-time algorithm, that given an instance (I,k), outputs an instance (I',k') such that $|I'|+k' \leq g(k)$ and any c-approximate solution s' to (I',k') can be turned in polynomial time into a $(c\cdot\alpha)$-approximate solution s to the original instance (I,k). More precisely, let Π be a parameterized minimization problem and let $\alpha \geq 1$. An α-*approximate* (or *lossy*) kernel for Π is a pair of polynomial algorithms $\mathcal{A}$ and $\mathcal{A}'$ such that

(i) given an instance (I,k), $\mathcal{A}$ (called a *reduction algorithm*) computes an instance (I',k') with $|I'|+k' \leq g(k)$, where $g(\cdot)$ is a computable function,

(ii) the algorithm $\mathcal{A}'$ (called a *solution-lifting algorithm*), given the initial instance (I,k), the instance (I',k') produced by $\mathcal{A}$, and a solution s' to (I',k'), computes an solution s to (I,k) such that

$$\frac{\Pi(I,k,s)}{\mathsf{Opt}_\Pi(I,k)} \leq \alpha \cdot \frac{\Pi(I',k',s')}{\mathsf{Opt}_\Pi(I',k')}.$$

For simplicity, we assume here that $\frac{\Pi(I,k,s)}{\mathsf{Opt}_\Pi(I,k)} = 1$ if $\mathsf{Opt}_\Pi(I,k) = \Pi(I,k,s) = 0$ and $\frac{\Pi(I,k,s)}{\mathsf{Opt}_\Pi(I,k)} = +\infty$ if $\mathsf{Opt}_\Pi(I,k) = 0$ and $\Pi(I,k,s) > 0$; the same assumption is used for $\frac{\Pi(I',k',s')}{\mathsf{Opt}_\Pi(I',k')}$. As with classical kernels, $g(\cdot)$ is called the *size* of an approximate kernel, and an approximate kernel is polynomial if $g(\cdot)$ is a polynomial. Analogous to the result that a decidable parameterized decision problem admits a kernel if and only if it is FPT, it holds that a computable parameterized optimization problem admits polynomial time α-approximation if and only if it admits α-lossy kernel.

Chapter 3

Problem Definitions

In this section, we discuss and define the clustering problems that we consider in this thesis.

3.1 Clustering

In its most general form, the k-Median Clustering problem is defined as follows. Given a set $\mathbf{X}$ of n points in a space $\mathcal{M}$ with a distance measure function $\mathsf{dist} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_{\geq 0}$, a description of a set $\mathbf{F} \subseteq \mathcal{M}$ of possible centers and an integer k, and the task is to find a pair $(\mathcal{X}, C)$, where $\mathcal{X}$ is a partition of $\mathbf{X}$ into k subsets $\{X_1, \ldots, X_k\}$, called clusters, $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\} \subseteq \mathbf{F}$ is a set of k centers, and the goal is to minimize the following cost over all pairs $(C, \mathcal{X})$

$$\mathsf{cost}(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}(\mathbf{c}_i, \mathbf{x}).$$

Note that one can place the center at any point in $\mathbf{F}$. We say that a partition $\{X_1, \ldots, X_k\}$ of $\mathbf{X}$ is an *k-median clustering* of $\mathbf{X}$. We assume that we are given black-box access to dist. Namely, given two points $\mathbf{x}, \mathbf{y} \in \mathbf{X}$, we assume that $\mathsf{dist}(\mathbf{x}, \mathbf{y})$ can be computed in constant time.

We mainly study the variant of the problem where $\mathbf{F} = \mathcal{M}$, that is, centers can be placed arbitrarily anywhere in the metric space $\mathcal{M}$. In the literature, this variant is often called Continuous k-Median Clustering. Moreover, in the continuous version of the problem, we often call centers medians. In our thesis, whenever we mention k-Median Clustering, we refer to this continuous variant of the problem. The variant when $\mathbf{F}$ is a

finite set given as a part of input and distinct from $\mathbf{X}$, the problem is called Discrete k-Median Clustering. This problem is also known as k-Median Facility Location where input point set $\mathbf{X}$ corresponds to the set of clients that need to be served by the facilities selected from the set of centers $\mathbf{F}$, and the cost of establishing a facility is zero, but we have an upper bound on the number of facilities allowed to set up, and the objective is to minimize the total cost of serving the clients.

For a k-median clustering $\{X_1, \ldots, X_k\}$ and given vectors $\mathbf{c}_1, \ldots, \mathbf{c}_k$ in $\mathcal{M}$, we define the *cost of clustering with respect to* $\mathbf{c}_1, \ldots, \mathbf{c}_k$ as

$$\mathsf{cost}(X_1, \ldots, X_k, \mathbf{c}_1, \ldots, \mathbf{c}_k) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}(\mathbf{x}, \mathbf{c}_i).$$

In Categorical k-Median Clustering, $\mathcal{M}$ is the set of strings of length m from Σ over a finite alphabet equipped with the Hamming distance, and the objective is to minimize the following cost over all pairs $(C, \mathcal{X})$

$$\mathsf{cost}(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}).$$

Observe that viewing points as strings is very natural concerning dist_0 (Hamming distance) and is especially interesting when Σ is small.

In Euclidean k-Median Clustering, $\mathcal{M}$ is a d-dimensional Euclidean space $\mathbb{R}^d$, dist is the Euclidean distance, denoted by $\mathsf{dist}_2(\mathbf{x}, \mathbf{y})$, i.e.,

$$\mathsf{dist}_2(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_2 = (\sum_{i=1}^{d} |\mathbf{x}[i] - \mathbf{y}[i]|^2)^{\frac{1}{2}}.$$

One can also consider the generalization of Euclidean k-Median Clustering to ℓ_p distances, we call the problem ℓ_p-k-Median Clustering. Here, the metric space $\mathcal{M}$ is still $\mathbb{R}^d$, but dist is defined by the ℓ_p-norm, i.e.,

$$\mathsf{dist}_p(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_p = (\sum_{i=1}^{d} |\mathbf{x}[i] - \mathbf{y}[i]|^p)^{\frac{1}{p}}.$$

The basic model of k-Median Clustering has various weaknesses, and one such is that it allows no control over the structure of the clusters, apart from the global cost minimization. This motivates us to consider clustering variants with size constraints called Capacitated k-Median clustering. Here, along with the k-median clustering

instance, we are given two positive integers p and q, where $p \leq q$ and the objective is to find a partition $\mathcal{X} = \{X_1, \ldots, X_k\}$ of $\mathbf{X}$ and centers $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\}$ minimizing k-median clustering $\mathsf{cost}(\mathcal{X}, C)$ over all the pairs $(\mathcal{X}, C)$ subject to the constraint that the size of each resulting cluster is at least p and at most q. If $p = q = \frac{n}{k}$, that is, the size of each resulting cluster is required to be equal to $\frac{n}{k}$, we call the problem Equal k-Median Clustering.

We call the Capacitated k-Median clustering problem Categorical Capacitated k-Median Clustering when $\mathcal{M}$ is the set of strings of length m from Σ over a finite alphabet and distance considered is Hamming, i.e., dist_0. More precisely, in Categorical Capacitated k-Median Clustering we are given a multiset $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ of n points from Σ^m over a finite alphabet, a positive integer k, a non-negative integer B, and positive integers p and q such that $p \leq q$, and the goal is to decide whether there is a partition $\mathcal{X} = \{X_1, \ldots, X_k\}$ of $\mathbf{X}$, where $p \leq |X_i| \leq q$, and vectors $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\}$ in Σ^m such that

$$\mathsf{cost}(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}) \leq B.$$

The next most natural model is restricting the input in ℓ_p-k-Median Clustering to the integral values. Moreover, we also desire that the resulting cluster's size is the same. That is, when $\mathcal{M}$ is $\mathbb{Z}^d$, and dist is defined as the l_p-norm, and the objective is to find a partition $\mathcal{X} = \{X_1, \ldots, X_k\}$ of $\mathbf{X} \subseteq \mathbb{Z}^d$ of points and k centers $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\}$ in $\mathbb{R}^d$ such that size of each cluster is same, that is $|X_1| = \ldots, |X_k| = \frac{|\mathbf{X}|}{k}$ minimizing the following objective function over all the pairs $(C, \mathcal{X})$

$$\mathsf{cost}(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_i,).$$

We call the problem ℓ_p-Equal k-Median Clustering.

In the last, we briefly mention the some related types of clustering problems called k-Center and k-Means. k-Means is defined analogously to k-Median Clustering, except that the objective is to minimize the following cost over all pairs $(C, \mathcal{X})$

$$\mathsf{cost}(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} (\mathsf{dist}(\mathbf{c}_i, \mathbf{x}))^2.$$

.

In k-Center, the objective is to minimize the maximum distance of a point to its

nearest center, i.e., $\min_{i=1}^{k} \max_{\mathbf{x}\in\mathbf{X}_i} \mathsf{dist}(\mathbf{x}, \mathbf{c}_i)$. The k-Center problem is also known in the literature as k-supplier.

3.2 Common Result

In this section, we provide an auxiliary result for Capacitated k-Median clustering which will be used later in several chapters.

Observe that given vectors $\mathbf{c}_1, \ldots, \mathbf{c}_k$, we can find a k-median clustering $\{X_1, \ldots, X_k\}$ that minimizes $\sum_{j=1}^{k} \sum_{\mathbf{x}\in X_j} \mathsf{dist}(\mathbf{x}, \mathbf{c}_j)$ by following the greedy procedure. For each $i \in \{1, \ldots, n\}$, we find $j \in \{1, \ldots, k\}$ such that $\mathsf{dist}(\mathbf{x}_i, \mathbf{c}_j)$ is minimum (ties are broken arbitrarily) and place $\mathbf{x}_i$ in the cluster X_j. Since

$$\sum_{i=1}^{n} \min\{\mathsf{dist}(\mathbf{c}_j, \mathbf{x}_i) \mid 1 \leq j \leq k\} \leq \sum_{j=1}^{k} \sum_{\mathbf{x}_i\in X_j} \mathsf{dist}(\mathbf{c}_j, \mathbf{x}_i),$$

for every k-median clustering $\{X_1, \ldots, X_k\}$, the described greedy procedure produces optimal partition of $\mathbf{X}$ (some sets may be empty). However, the constructed k-clustering does not respect the size constraints. Still, given vectors $\mathbf{c}_1, \ldots, \mathbf{c}_k$, we can decide in polynomial time whether an instance of Capacitated k-Median clustering has a solution with the medians $\mathbf{c}_1, \ldots, \mathbf{c}_k$ using a reduction to the classical Minimum Weight Perfect Matching problem on bipartite graphs that is well-known to be solvable in polynomial time by the Hungarian method of Kuhn [59] (see also [66]).

Recall that a *matching* M of a graph G is a set of edges without common vertices. It is said that a matching M *saturates* a vertex v if M has an edge incident to v. A matching M is *perfect* if every vertex of G is saturated. The task of Minimum Weight Perfect Matching is, given a bipartite graph G and a weight function $w\colon E(G) \to \mathbb{R}_{\geq 0}$, to find a perfect matching M (if it exists) such that its weight $w(M) = \sum_{e\in M} w(e)$ is minimum. We show the following result.

Lemma 1. *Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be a collection of n points in space $\mathcal{M}$, k be a positive integer, and let p and q be two positive integers such that $p \leq q$. Let also $\mathbf{c}_1, \ldots, \mathbf{c}_k$ be the points in $\mathcal{M}$. Then a capacitated k-median clustering of minimum $\mathsf{cost}(X_1, \ldots, X_k; \mathbf{c}_1, \ldots, \mathbf{c}_k)$ can be computed in polynomial time.*

Proof. Assume $p \leq \frac{n}{k} \leq q$, that is $kp \leq n \leq kq$. Otherwise, a capacitated k-median clustering does not exist. Given $\mathbf{X}$ and centers $\{\mathbf{c}_1, \ldots, \mathbf{c}_k\}$, and positive integers p and q, we construct the bipartite graph G as follows.

- For each $i \in \{1, \ldots, k\}$, construct a set of p vertices $W_i = \{v_1^i, \ldots, v_p^i\}$ and a set of $q - p$ vertices $W_i' = \{v_{p+1}^i, \ldots, v_q^i\}$; note that $W_i' = \emptyset$ if $p = q$. Let $V_i = W_i \cup W_i'$ for $i \in \{1, \ldots, k\}$ and denote $V = \bigcup_{i=1}^{k} V_i$; the block of vertices V_i corresponds to the median $\mathbf{c}_i$.

- For each $i \in \{1, \ldots, n\}$, construct a vertex u_i corresponding to each point $\mathbf{x}_i$ of $\mathbf{X}$ and make u_i adjacent to the vertices of V. Denote $U = \{u_1, \ldots, u_n\}$.

- Construct a set of $s = kq - n$ vertices $U' = \{u_1', \ldots, u_s'\}$ that we call *fillers* and make the vertices of U' adjacent to the vertices of W_j' for all $j \in \{1, \ldots, k\}$; note that $U' = \emptyset$ if $n = qk$ and observe that $qk \geq n$ by our assumption.

Observe that G is a bipartite graph, where $U \cup U'$ and V form the bipartition. Note also that $|U \cup U'| = |V| = qk$.

- For every $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, k\}$, set $w(u_i v_h^j) = \mathsf{dist}(\mathbf{c}_j, \mathbf{x}_i)$ for $h \in \{1, \ldots, q\}$, that is, the weight of all edges joining u_i corresponding to $\mathbf{x}_i$ with the vertices of V_j corresponding to the median $\mathbf{c}_j$.

- For every $i \in \{1, \ldots, s\}$ and $j \in \{1, \ldots, k\}$, set $w(u_i' v_h^j) = 0$ for $h \in \{p+1, \ldots, q\}$, that is, the edges incident to the fillers have zero weights.

We show one-to-one correspondence between perfect matchings of G and k-clusterings of $\mathbf{X}$.

In the forward direction, assume that M is a perfect matching of G. We construct the clustering $\{X_1, \ldots, X_k\}$ as follows. For every $h \in \{1, \ldots, n\}$, u_h is saturated by M and, therefore, there are $i_h \in \{1, \ldots, k\}$ and $j_h \in \{1, \ldots s\}$ such that edge $u_h v_{j_h}^{i_h} \in M$. Consider $M' = \{u_h v_{j_h}^{i_h} \mid 1 \leq h \leq n\} \subseteq M$. We cluster the points of $\mathbf{X}$ according to M'. Formally, we place $\mathbf{x}_h$ in X_{i_h} for each $h \in \{1, \ldots, n\}$. Observe that for each $i \in \{1, \ldots, k\}$, the vertices of W_i are adjacent only to the vertices of U. Since these vertices are saturated by M, we obtain that $|X_i| \geq p$ for every $i \in \{1, \ldots, k\}$. Since $|V_i| = q$, $|X_i| \leq q$ for all $i \in \{1, \ldots, k\}$. Now we upper bound the cost of the obtained k-clustering:

$$\sum_{i=1}^{k} \sum_{\mathbf{x}_j \in X_i} \mathsf{dist}(\mathbf{c}_i, \mathbf{x}_j) = \sum_{h=1}^{n} \mathsf{dist}(\mathbf{c}_{i_h}, \mathbf{x}_h) = w(M') \leq w(M).$$

For the reverse direction, consider a k-clustering $\{X_1, \ldots, X_k\}$ for $\mathbf{X}$ such that $p \leq |X_i| \leq q$ for all $i \in \{1, \ldots, k\}$. Let $i \in \{1, \ldots, k\}$. Consider the cluster X_i and assume

that $X_i = \{\mathbf{x}_{j_1}, \ldots, \mathbf{x}_{j_{h_i}}\}$ and $p \leq |X_i| \leq q$. Recall that every vertex of V_i is adjacent to every vertex of U. Let $M_i = \{u_{j_1}v_1^i, \ldots, u_{j_{h_i}}v_{h_i}^i\}$. Clearly, M_i is a matching saturating the first $p \leq h_i \leq q$ vertices of V_i. In particular, the vertices of W_i are saturated. We construct M_i for every $i \in \{1, \ldots, k\}$ and set $M' = \bigcup_{i=1}^{k} M_i$. Since $\{X_1, \ldots, X_k\}$ is a partition of $\mathbf{X}$, M' is a matching saturating every vertex of U. Denote by V' the set of vertices of V that are not saturated by M'. Notice that $V' \subseteq \bigcup_{i=1}^{k} W_i'$ because the vertices of each W_i are saturated by M_i. Observe that every vertex of U' is adjacent to every vertex of W_i' for $i \in \{1, \ldots, k\}$, that is, $G[U' \cup V']$ is a complete bipartite graph. Because $|U'| = |V'| = s$, $G[U' \cup V']$ has a perfect matching M''. We set $M = M' \cup M''$. It is easy to see that M is a matching, and since M saturates every vertex of G, M is a perfect matching. To evaluate the weight of M, recall that the edges of G incident to the fillers have zero weights, that is, $w(M'') = 0$. Then

$$\begin{aligned} w(M) =& w(M') = w(\bigcup_{i=1}^{k} M_i) = \sum_{i=1}^{k} \sum_{e \in M_i} w(e) = \sum_{i=1}^{k} \left(w(u_{j_1}v_1^i) + \cdots + w(u_{j_{h_i}}v_{h_i}^i)\right) \\ =& \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in X_i} \mathsf{dist}(\mathbf{c}_i, \mathbf{x}_j). \end{aligned}$$

Thus, finding a k-clustering $\{X_1, \ldots, X_k\}$ that minimizes $\mathsf{cost}(X_1, \ldots, X_k, \mathbf{c}_1, \ldots, \mathbf{c}_k)$ is equivalent to computing a perfect matching of minimum weight in G. Then, because a perfect matching of minimum weight in G can be found in polynomial time [59, 66], a k-clustering of minimum cost can be found in polynomial time.

□

Chapter 4

Exact Exponential Algorithms for Clustering Problems

In this chapter, we study Discrete k-Median Clustering. In particular, we give exact algorithms for two of its variants in general metric space. First, when $\mathbf{F} = \mathbf{X}$, that is, one can pick centers from any point in the input set $\mathbf{X}$, which we call Restricted k-Median Clustering. The second variant is when the set of centers $\mathbf{F}$ is a finite set distinct from $\mathbf{X}$ and given as part of the input. Recall that this variant is known as k-Median Facility Location. We also complement the results by showing that under a certain complexity-theoretic assumption, the running time of the algorithm is asymptotically optimal upto the base of the exponent. The results mentioned in this chapter have appeared in Article 1. Formally, we define Restricted k-Median Clustering as follows.

Restricted k-Median Clustering

Input: A set of n points $\mathbf{X}$ with a distance measure function $\mathsf{dist} : \mathbf{X} \times \mathbf{X} \to \mathbb{R}_{\geq 0}$ and an integer k.

Task: The task is to find a pair $(C, \mathcal{X})$, where $\mathcal{X}$ is a partition of $\mathbf{X}$ into k subsets $\mathcal{X} = \{X_1, \ldots, X_k\}$, called clusters, and $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\} \subseteq \mathbf{X}$ is a set of k centers. The goal is to minimize the following cost over all pairs $(C, \mathcal{X})$

$$\mathsf{cost}(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}(\mathbf{c}_i, \mathbf{x}).$$

We say that a partition $\{X_1, \ldots, X_k\}$ of $\mathbf{X}$ is a *restricted k-median clustering* of $\mathbf{X}$.

Agarwal and Procopiuc [2] gave an *exact* algorithm for k-Center in $\mathbb{R}^d$ running in $n^{\mathcal{O}(k^{1-\frac{1}{d}})}$ time. In particular, in two dimensional space, their algorithm runs in $2^{\mathcal{O}(\sqrt{n}\log n)}$ time for any value of k, i.e., in *sub-exponential* time. This initiated this work and led us towards a natural question, studying the complexity of Restricted k-Median Clustering in general metric space.

Recall that for Restricted k-Median Clustering, it is easy to design an exact algorithm that runs in time $\binom{n}{k} \cdot n^{\mathcal{O}(1)}$ – it simply enumerates all sets of centers of size k, and the corresponding partition of $\mathbf{X}$ into clusters is obtained by assigning each point to its nearest center. Then, we simply return the solution with the minimum cost. The naïve algorithm has running time $\mathcal{O}^*(2^n)$ when k belongs to the range $n/2 \pm o(n)$, $\binom{n}{k} \simeq 2^n$. We design an exact algorithms for the problem with running time $c^n \cdot n^{\mathcal{O}(1)}$ for a constant $c < 2$, that is, as small as possible. In particular, we obtain an $\mathcal{O}^*((1.89)^n)$ time *exact* algorithm for Restricted k-Median Clustering that works for any value of k. We show the following result.

Theorem 1. *There is an exact algorithm for* Restricted k-Median Clustering *running in time* $(1.89)^n \cdot n^{\mathcal{O}(1)}$*, where n is the number of points in* $\mathbf{X}$*.*

This is the first non-trivial exact algorithms for Restricted k-Median Clustering. Our algorithm is quite general in the sense that it does not use any properties of the underlying (metric) space – it does not even require the distances to satisfy the triangle inequality. We complement this result by showing that the running time of our algorithm is asymptotically optimal, up to the base of the exponent. That is, unless the Exponential Time Hypothesis fails, there is no algorithm for these problems running in time $2^{o(n)} \cdot n^{\mathcal{O}(1)}$. Recall that the formal definition of ETH is given in Section 2, and we prove the ETH-hardness result in Section 4.2.

Theorem 2. Restricted k-Median Clustering *cannot be solved in time $2^{o(n)}$ time unless the exponential-time hypothesis fails, where n is the number of points in* $\mathbf{X}$*.*

To explain the idea behind Theorem 1, consider the following fortuitous scenario. Suppose that the optimal solution only contains clusters of size exactly 2. In this case, it is easy to solve the problem optimally by reducing the problem to finding a Minimum Weight Perfect Matching in the complete graph defining the metric. Note that the cluster-center always belongs to its own cluster, which implies that a cluster of size 2 contains one *additional* point. This immediately suggests the connection to minimum-weight matching. Note that the problem of finding Minimum Weight Perfect Matching is known to be polynomial-time solvable by the classical result of

Edmonds [52]. This idea can also be extended if the optimal solution only contains clusters of size 1 and 2, by finding matching in an auxiliary graph. However, the idea does not generalize to clusters of size 3 and more, since we need to solve a problem that has a flavor similar to the 3-dimensional matching problem or the "star partition" problem, which are known to be NP-hard [16, 43, 56]. Nevertheless, if the number of points belonging to the clusters of size at least 3 is *small*, one can "guess" these points, and solve the remaining points using matching. However, the number of points belonging to the clusters of size at least 3 can be quite large – it can be as high as n. But note that the number of *centers* corresponding to clusters of size at least 3 can be at most $n/3$. We show that "guessing" the subset of centers of such clusters is sufficient (as opposed to guessing *all* the points in such clusters), in the sense that an optimal clustering of the "residual" instance can be found again by finding a minimum-weight matching in an appropriately constructed auxiliary graph.

We briefly explain the idea behind the construction of this auxiliary graph. Note that in order to find an optimal clustering in the "residual" instance, we need to figure out the following things: (1) the set of points that are involved in clusters of size 1, i.e., *singleton* clusters, (2) the pairs of points that become clusters of size 2, and (3) for each center c_i of a cluster of size at least 3, the set of at least two additional points that are connected to c_i. We find the set of points of type (1) by matching them to a set of *dummy* points with zero-weight edges. The pairs of points involved in clusters of size 2 naturally correspond to a matching, such that the weight of each edge corresponds to the distance between the corresponding pair of points. Finally, to find points of type (3), we make an appropriate number of *copies* of each guessed center c_i that will be matched to the corresponding points. Although the high-level idea behind the construction of the graph is very natural, it is non-trivial to construct the graph such that a minimum-weight perfect matching in the auxiliary graph exactly corresponds to an optimal clustering (assuming we guess the centers correctly). Thus, this construction pushes the boundary of applicability of matching in order to find an optimal clustering. Since the minimum-weight perfect matching problem can be solved in polynomial time, the running time of our algorithm is dominated by guessing the set of centers of clusters of size at least 3. As mentioned previously, the number of such centers is at most $n/3$, which implies that the number of guesses is at most $\binom{n}{n/3} \leq (1.89)^n$, which dominates the running time of our algorithm. We describe this result in Section 4.1.

We show that the "facility" location version the Restricted k-Median Clustering problem is computational harder. We remind that k-Median Facility Location is defined as follows.

k-Median Facility Location

Input: A set of n points $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, called clients, and a set $\mathbf{F}$ of m possible centers in a space $\mathcal{M} = \mathbf{X} \cup \mathbf{F}$ with a distance measure function $\mathsf{dist} : \mathcal{M} \times \mathcal{M} \to \mathbb{R}_{\geq 0}$ and an integer k.

Task: The task is to find a pair $(C, \mathcal{X})$, where $\mathcal{X}$ is a partition of $\mathbf{X}$ into k subsets $\mathcal{X} = \{X_1, \ldots, X_k\}$, called clusters, and $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\} \subseteq \mathbf{F}$ is a set of k centers. The goal is to minimize the following cost over all pairs $(C, \mathcal{X})$

$$\mathsf{cost}(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}(\mathbf{c}_i, \mathbf{x}).$$

.

For Restricted k-Median Clustering, we beat the "trivial" bound of $\mathcal{O}(2^n)$, by giving a $\mathcal{O}((1.89)^n)$ time algorithm. However, for k-Median Facility Location, we show that it is not possible to obtain a $2^{(1-\epsilon)n} \cdot (mn)^{\mathcal{O}(1)}$ time algorithm for any fixed $\epsilon > 0$, where $m = |\mathbf{F}|$ and $n = |\mathbf{X}|$. To show this result, we use the Set Cover Conjecture (see Section 2), which is a complexity-theoretic hypothesis proposed by Cygan et al. [28]. In Section 4.3, we show the following.

Theorem 3. *Assuming* Set Cover Conjecture, *for any fixed $\epsilon > 0$, there is no $2^{(1-\epsilon)n} \cdot m^{\mathcal{O}(1)}$ time algorithm for k-*Median Facility Location*, where n is the number of clients and m is the number of potential locations.*

We match this lower bound by designing an algorithm with running time $2^n \cdot (mn)^{\mathcal{O}(1)}$ under some mild assumptions. While this algorithm is not obvious, it is a relatively straightforward application of the subset convolution technique. The details are in Section 4.4.

Organization of the chapter In Section 4.1, we give an exact algorithm for Restricted k-Median Clustering. In Section 4.2, assuming the Exponential Time Hypothesis, we establish the impossibility of solving Restricted k-Median Clustering in subexponential time in the number of input points. In Section 4.3, we show that k-Median Facility Location can not be solved in subexponential time in the number of clients assuming the set cover conjecture. Further, in Section 4.4, using subset convolution, we give the exact algorithm for k-Median Facility Location.

4.1 Exact Algorithm for Restricted k-Median Clustering.

In this section, we prove Theorem 1 which is restated.

Theorem 1. *There is an exact algorithm for* Restricted k-Median Clustering *running in time* $(1.89)^n \cdot n^{\mathcal{O}(1)}$*, where* n *is the number of points in* $\mathbf{X}$*.*

Before delving into the proof of Theorem 1, we discuss the approach at a high level. We begin by "guessing" a subset of centers from an (unknown) optimal solution. For each guess, the problem of finding the best (i.e., minimum-cost) clustering that is "compatible" with the guess is reduced to finding a minimum weight perfect matching in an auxiliary graph G (note, here we do not know all the centers) similarly to the proof of Lemma 1.

The graph G is constructed in such a way that this clustering can be extracted by essentially looking at the minimum-weight perfect matching. Note that Minimum Weight Perfect Matching problem is well known to be solvable in polynomial time by the Blossom algorithm of Edmonds [52]. Finally, we simply return a minimum-cost clustering found over all guesses.

Let us fix some optimal restricted k-median clustering solution and let k_1^*, k_2^* and k_3^* be a partition of k, where k_1^* : the number of clusters of size exactly 1, call *Type1*; k_2^* : the number of clusters of size exactly 2, call *Type2*; and k_3^* : the number of clusters of size at least 3, call *Type3*. Let $C_3^* \subseteq \mathbf{X}$ be *Type3* centers, and say $C_3^* = \{c_1, \ldots, c_{k_3^*}\}$. Observe that number of clusters with *Type3* centers is at most $\frac{n}{3}$. Suppose not, then the number of clusters with *Type3* centers is greater than $\frac{n}{3}$. Each *Type3* cluster contains at least three points. This contradicts that the number of input points is n.

Algorithm. First, we guess the partition of k into k_1, k_2, k_3 as well as a subset $C_3 \subseteq X$ of size at most $n/3$. For each such guess (k_1, k_2, k_3, C_3), we construct the auxiliary graph G (as defined subsequently) corresponding to this guess, and compute a minimum weight perfect matching M in G. Let M^* be a minimum weight perfect matching over *all* the guesses. We extract the corresponding clustering $(C^*, \mathcal{X}^*)$ from M^* (also explained subsequently), and return as an optimal solution of the given instance.

Running time. Note that there are at most $\mathcal{O}(k^2)$ tuples (k_1, k_2, k_3) such that $k_1 + k_2 + k_3 \leq k$ (note that k_i's are non-negative integers). Furthermore, there are at most $\sum_{i=0}^{n/3} \binom{n}{i} \leq (1.89)^n$ subsets of $\mathbf{X}$ of size at most $n/3$. Here, the sum of binomial coefficients $\sum_{i=0}^{n/3} \binom{n}{i}$ is upper bounded using the inequality $\binom{n}{\alpha n} \leq \frac{n^n}{(\alpha n)^{\alpha n} \cdot (n-\alpha n)^{n-\alpha n}} =$

$[(\frac{1}{\alpha})^{\alpha} \cdot (\frac{1}{1-\alpha})^{1-\alpha}]^n$ for $\alpha = \frac{1}{3}$ (see [39]). Finally, constructing the auxiliary graph, and finding a minimum-weight perfect matching takes polynomial time. Thus, the running time is dominated by the number of guesses for C_3, which implies that we can bound the running time of our algorithm by $\mathcal{O}^*((1.89)^n)$.

Construction of Auxiliary Graph. From now on assume that our algorithm made the right guesses, i.e., suppose that $(k_1, k_2, k_3) = (k_1^*, k_2^*, k_3^*)$ and $C_3^* = C'$. Then, we initialize the *Type3* centers by placing each center from C' into a separate cluster. At this point, to achieve this, we reduce the problem to the classical Minimum Weight Perfect Matching on an auxiliary graph G, which we define as follows. See Figure 4.1 for an illustration of the construction.

- For each $i \in \{1, \ldots, k_3\}$, construct a set of $s = n - k_3 - 2k_2 - k_1$ vertices $C_i = \{c_1^i, \ldots, c_s^i\}$. Denote $W = \bigcup_{i=1}^{k} C_i$; the block of vertices C_i corresponds to center c_i.
- Let $Y = \mathbf{X} \setminus C'$, that is, a set consisting of unclustered points in $\mathbf{X}$. Observe $|Y| = n - k_3$. Denote $Y = \{y_1, \ldots, y_{(n-k_3)}\}$. For simplicity, we slightly abuse the notation by keeping the vertices in G same as points in Y. That is, for each $i \in \{1, \ldots, (n - k_3)\}$, place a vertex y_i in the set Y. Make each y_i adjacent to all vertices of W.
- For each $i \in \{1, \ldots, k_1\}$, construct an auxiliary vertex u_i. Denote $U_{\mathsf{iso}} = \{u_1, \ldots, u_{k_1}\}$. Make each u_i adjacent to every vertex of Y.
- Construct a set of $s(k_3 - 1)$ vertices, $Z_{\mathsf{fill}} = \{z_1, \ldots, z_{s(k_3-1)}\}$, that we call fillers and make vertices of Z_{fill} adjacent to the vertices of W.

We define edge weights. For an edge $uv \in E(G)$, we will use $w(uv)$ to denote the weight of the edge uv.

- For every $i \in \{1, \ldots, (n - k_3)\}$ and every $j \in \{1, \ldots, k_3\}$ set $w(y_i c_h^j) = \mathsf{dist}(y_i, c_j)$ for $h \in \{1, \ldots, s\}$, i.e., weight of all edges joining y_i in Y with the vertices of C_i corresponding to center c_j.
- For every $i, j \in \{1, \ldots, n - k_3\}$, $i \neq j$, set $w(y_i y_j) = \mathsf{dist}(y_i, y_j)$, i.e., the weight of edges between vertices of Y.
- For every $i \in \{1, \ldots, k_1\}$ and $j \in \{1, \ldots, (n - k_3)\}$, set $w(u_i y_j) = 0$, i.e., the edges incident to the vertices of U_{iso} have zero weights.

- For every $i \in \{1, \ldots, s(k_3-1)\}$ and $j \in \{1, \ldots, k_3\}$, $w(z_i c_h^j) = 0$, for $h \in \{1, \ldots, s\}$, i.e., the edges incident to the fillers have zero weights.

Lemma 2. *The graph G has a perfect matching.*

Proof. We construct a set $M \subseteq E(G)$ that saturates every vertex in G.

Note that $|U_{\mathsf{iso}}| < |Y|$ and every vertex of U_{iso} is adjacent to every vertex of Y. Therefore, we can construct $M_1 \subseteq E(G)$ by arbitrarily mapping each vertex of U_{iso} to a distinct vertex of Y. Clearly, M_1 is matching saturating vertices of U_{iso}. Since $|U_{\mathsf{iso}}| = k_1$, M_1 saturates k_1 vertices of Y. Denote by Y' the set of vertices of Y that are not saturated by M_1. Observe $|Y'| = s + 2k_2$.

Every vertex of Z_{fill} is adjacent to every vertex of W and $|Z_{\mathsf{fill}}| < |W|$. Construct $M_2 \subseteq E(G)$ by arbitrarily mapping each vertex of Z_{fill} to a distinct vertex of W. Thus, M_2 is a matching which saturates every vertex of Z_{fill} and since $|Z_{\mathsf{fill}}| = s(k_3 - 1)$, it also saturates $s(k_3 - 1)$ vertices of W. Denote by W' the set of vertices of W that is not saturated by M_2. Observe $|W'| = s$. Recall, every vertex of W' is adjacent to every vertex of Y' and note that $|W'| < |Y'|$. Therefore, construct $M_3 \subseteq E(G)$ by arbitrarily matching each vertex of W' with a distinct vertex of Y'.

Thus, the matching M_3 saturates s vertices in both the sets W' and Y'. Denote $M' = M_1 \cup M_2 \cup M_3$. Clearly, the vertices of U_{iso}, W and Z_{fill} are saturated by M'.

Denote by $Y'' = Y \setminus Y'$ the set of vertices of Y that are not saturated by M'. Note that $|Y''| = 2k_2$. Consider $M_4 \subseteq E(G)$ which maps these $2k_2$ vertices to each other. We set $M = M' \cup M_4$. It is easy to see that M is a perfect matching. □

We next show one-to-one correspondence between perfect matchings of G and the restricted discrete k-median clusterings of $\mathbf{X}$.

Lemma 3. *Let $\mathsf{OPT}_{\mathsf{mm}}(G)$ = weight of minimum weight perfect matching, and $\mathsf{OPT}_{\mathsf{Rkmed}}(\mathbf{X})$ = optimal clustering cost of restricted k-median clustering of $\mathbf{X}$. Then, $\mathsf{OPT}_{\mathsf{mm}}(G) = \mathsf{OPT}_{\mathsf{Rkmed}}(\mathbf{X})$.*

Proof. In the forward direction, let M denote a minimum weight perfect matching $M \subseteq E(G)$. We construct a restricted k-median clustering of $\mathbf{X}$ of same cost.

Observe that each vertex of Z_{fill} is only adjacent to the vertices of W and $|Z_{\mathsf{fill}}| < |W|$. Let $W_1 \subseteq W$ be a set of vertices matched to vertices of Z_{fill}. Since G has a perfect matching, it saturates Z_{fill}, where $|Z_{\mathsf{fill}}| = s(k_3 - 1)$. Then, $|W_1| = s(k_3 - 1)$. Let $W_2 = W \setminus W_1$ be set of vertices matched to vertices of Y. Clearly, $|W_2| = s$.

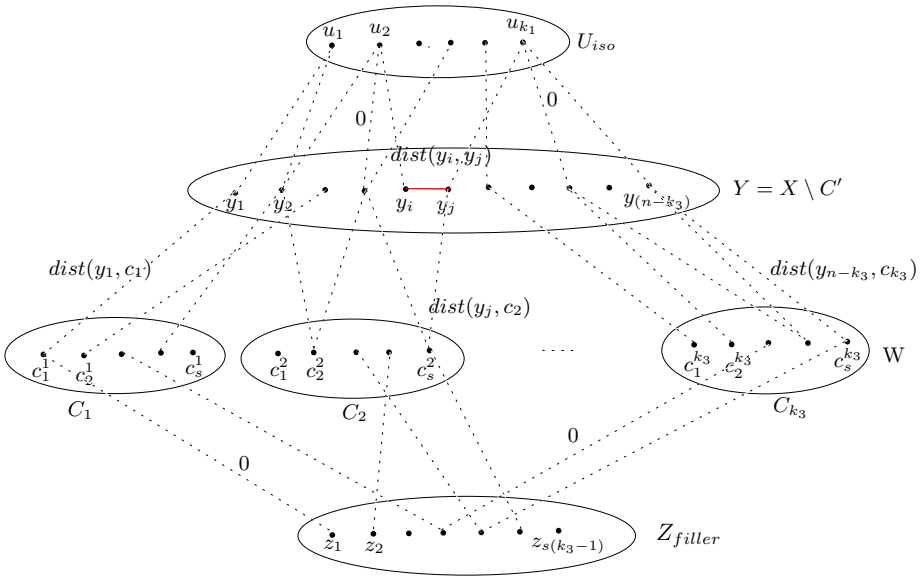

Figure 4.1: Illustration of the graph G produced in the reduction from Restricted k-Median Clustering to Minimum Weight Perfect Matching. To avoid clutter, we only show some representative edges. Recall that we guess the set of k_3 centers of *Type3*, and corresponding to each such center c_i, we add a set C_i consisting of s copies corresponding to that center. Next, we have the set Y corresponding to $n-k_3$ unclustered points. Finally, U_{iso} and Z_{fill} consist of auxiliary vertices in order to ensure a perfect matching. The weights of vertices among Y correspond to the corresponding original distance; whereas the weight of an edge between $y_\ell \in Y$, and a copy c_i^j corresponding to a *Type3* center c_i is defined to be $\mathsf{dist}(y_\ell, c_i)$. The weights of all other edges are equal to zero.

For every $i \in \{1, \ldots, (n - k_3)\}$, vertex $y_i \in Y$ is saturated by M. Therefore, we construct the restricted k-median clustering $\{X_1, \ldots, X_k\}$ of $\mathbf{X}$, where each $X_i \in \{\textit{Type1}, \textit{Type2}, \textit{Type3}\}$, for $i \in \{1, \ldots, k\}$ as follows.

Let $Y' \subseteq Y$ be the set of vertices that are matched to vertices of U_{iso} in M, where $|U_{\mathsf{iso}}| = k_1 < |Y|$. Corresponding to each such vertex in Y', select a center in the solution C, call $C_{\textit{Type1}} = \{c^1_{\textit{Type1}}, \ldots, c^{k_1}_{\textit{Type1}}\}$. Correspondingly, also construct a singleton cluster $X_i = \{c^i_{\textit{Type1}}\}$, for $i \in \{1, \ldots, k_1\}$. Let $X_{\textit{Type1}}$ denote set of all *Type1* clusters.

We now construct *Type3* clusters: Let $Y'_i \subseteq Y$ be the set of vertices matched to set C_i, for $i \in \{1, \ldots, k_3\}$ in M. Consider $X_i = Y'_i \cup \{c_i\}$. Clearly, X_i, for $i \in \{1, \ldots, k_3\}$, corresponds to *Type3*, clusters in $\mathbf{X}$. Let $X_{\textit{Type3}}$ denote set of all *Type3* clusters. Recall, we already guessed set $C' = \{c_1, \ldots, c_{k_3}\}$, that is, *Type3* centers correctly.

Lastly, we construct clusters of *Type2*. Denote by Y'' set of unclustered points in Y. Observe these points form a set of k_2 disjoint edges in M. Arbitrarily, select one of the endpoint of each edge as a center in the solution C, call $C_{\textit{Type2}} = \{c^1_{\textit{Type2}}, \ldots, c^{k_2}_{\textit{Type2}}\}$. That is, for an edge $y_1y_2 \in M$, where $y_1, y_2 \in Y''$, select center as y_1 or y_2. Then construct a cluster X_i, for $i \in \{1, \ldots, k_2\}$ by placing both the endpoints of the edge in the same cluster. Denote by $X_{\textit{Type2}}$ the set of all *Type2* clusters.

Clearly, $X_i \in \{Type1, Type2, Type3\}$, for $i \in \{1, \ldots, k\}$ is a partition of $\mathbf{X}$. Note, since *Type1* clusters are isolated points, therefore, they contribute zero to the total cost of clustering. Now we upper bound the cost of the obtained restricted k-median clustering:

$$\sum_{i=1}^{k} \sum_{x \in X_i} \mathsf{dist}(c_i, x) = \sum_{i=1}^{k_2} \sum_{y \in X_{\textit{Type2}}} \mathsf{dist}(c^i_{\textit{Type2}}, y) + \sum_{i=1}^{k_3} \sum_{y \in X_{\textit{Type3}}} \mathsf{dist}(c_i, y) = \mathsf{OPT}_{\mathsf{mm}}(G).$$

For the reverse direction, consider a restricted k-median clustering $\{X_1, \ldots, X_k\}$ of $\mathbf{X}$ into $\{\textit{Type1}, \textit{Type2}, \textit{Type3}\}$ clusters of $\mathbf{X}$ such that $|\textit{Type1}| = k_1$, $|\textit{Type2}| = k_2$ and $|\textit{Type3}| = k_3$ and $C' = \{c_1, \ldots, c_{k_3}\}$, that is, centers of *Type3* clusters with $\mathsf{OPT}_{\mathsf{Rkmed}}(\mathbf{X})$. We construct a perfect matching $M \subseteq E(G)$ of G as follows.

Observe that each *Type1* cluster is a singleton cluster. Construct $M_1 \subseteq E(G)$ by iterating over each singleton vertex in Y corresponding to each cluster and match it to a distinct vertex in U_{iso}. Since $|\textit{Type1}| = |U_{\mathsf{iso}}| = k_1$, M_1 is a matching saturating set U_{iso}. Also, M_1 saturates k_1 vertices in Y.

Corresponding to each *Type2* cluster, construct $M_2 \subseteq E(G)$ by adding an edge between both the end vertices in Y. Clearly, M_2 is a disjoint set of k_2 edges in G and saturates $2k_2$ vertices in Y.

Denote $Y' \subseteq Y$ be the set of vertices matched by $M_1 \cup M_2$. Clearly, $|Y'| = k_1 + 2k_2$. Let $Y'' = Y \setminus Y'$ be the set of remaining unmatched vertices in Y. Then, $|Y''| = |Y| - |Y'| = n - k_3 - 2k_2 - k_1 = s$.

Note, we already guessed $C' = \{c_1, \ldots, c_{k_3}\}$ and we have a cluster X_i corresponding to each C_i, for $i \in \{1, \ldots, k_3\}$. Construct $M_3 \subseteq E(G)$ by matching each vertex of $X_i \setminus \{c_i\}$ in Y'' to a distinct copy of c_i in W. Since $|Y''| < |W|$, M_3 saturates Y''. Let $W_1 \subseteq W$ be the set of vertices saturated by M_3. Note that $|Y''| = s$, then $|W_1| = s$. Let $W_2 = W \setminus W_1$ be the set of vertices not saturated by M_3, where $|W| = sk_3$. Then, $|W_2| = s(k_3 - 1)$. Every vertex of Z_{fill} is only adjacent to every vertex of W (in particular of W_2). We construct $M_4 \subseteq E(G)$ by matching each vertex of Z_{fill} to a distinct vertex of W_2. Since $|Z_{\mathsf{fill}}| = |W_2| = s(k_3 - 1)$, M_4 saturates Z_{fill} and W_2.

To evaluate the weight of M, recall that the edges of G incident to the set U_{iso} and to the filler vertices Z_{fill} have zero weights, that is, $w(M_1) = w(M_4) = 0$. Then

$$\begin{aligned} w(M) = w(M_2) + w(M_3) &= \sum_{e \in M_2} w(e) + \sum_{e \in M_3} w(e) \\ &= \sum_{c_i : X_i \in X_{Type2}} \sum_{y \in X_i} \mathsf{dist}(c_i, y) + \sum_{c_i : X_i \in X_{Type3}} \sum_{y \in X_i} \mathsf{dist}(c_i, y) \\ &= \mathsf{OPT}_{\mathsf{Rkmed}}(X). \end{aligned}$$

This completes the proof. □

It is straightforward to see that the construction of the graph G from an instance $(\mathbf{X}, \mathsf{dist})$ of Restricted k-Median Clustering can be done in polynomial time. Then, because a perfect matching of minimum weight of the graph G can be found in polynomial time [52] and the total number of guesses is at most $(1.89)^n n^{\mathcal{O}(1)}$, Restricted k-Median Clustering can be solved exactly in $(1.89)^n n^{\mathcal{O}(1)}$ time. This completes the proof of Theorem 1.

4.2 ETH Hardness

In this section, we establish a result around the (im)possibility of solving Restricted k-Median Clustering in subexponential time in the number of points in $\mathbf{X}$. For this, we use the result of Lokshtanov et al. [64] which states that, assuming ETH, the Dominating set problem cannot be solved in time $2^{o(n)}$ time, where n is the number of vertices of the graph.

Given an unweighted, undirected graph G, a dominating set S is a subset of $V(G)$ such that each $v \in V(G)$ is dominated by S, that is, we either have $v \in S$ or there exists an edge $uv \in E(G)$ such that $u \in S$. The decision version of DOMINATING SET is defined as follows.

DOMINATING SET

Input:	Given an unweighted, undirected graph G, positive integer k.
Task:	Determine whether G has a dominating set of size at most k.

We use Proposition 3 from Section 2, to prove the following.

Theorem 2. RESTRICTED k-MEDIAN CLUSTERING *cannot be solved in time* $2^{o(n)}$ *time unless the exponential-time hypothesis fails, where* n *is the number of points in* $\mathbf{X}$.

Proof. We give a reduction from DOMINATING SET to RESTRICTED k-MEDIAN CLUSTERING. Let (G, k) be the given instance of DOMINATING SET. We assume that there is no dominating set in G of size at most $k-1$. This assumption is without loss of generality, since we can use the following reduction iteratively for $k' = 1, 2, \ldots, k$, which only incurs a polynomial overhead.

Now we construct an instance $(\mathbf{X}, \mathsf{dist})$ of RESTRICTED k-MEDIAN CLUSTERING as follows. First, let $\mathbf{X} = V(G)$, i.e., we treat each vertex of the graph as a point in the metric space, and we use the terms vertex and point interchangeably. Recall that the graph G is unweighted, but we suppose that the weight of every edge in $E(G)$ is 1. Then, we let dist be the shortest path metric in G. The following observations are immediate.

Observation 1.

- *For all* $u \in V(G)$, $\mathsf{dist}(u, u) = 0$.
- *For all distinct* $u, v \in V(G)$, $\mathsf{dist}(u, v) = 1$ *if and only if* $uv \in E(G)$, *and* $\mathsf{dist}(u, v) \geq 2$ *if and only if* $uv \notin E(G)$.

We now show that there is a dominating set of size k if and only if there is a restricted k-median clustering of cost exactly $n - k$.

In the forward direction, let $S \subseteq V(G)$ be a dominating set of size k. We obtain the corresponding restricted k-median clustering as follows. We let $S = \{c_1, c_2, \ldots, c_k\}$ to be the set of centers. For a center $c_i \in S$, we define $X'_i = N[c_i]$. Since S is a dominating

set, every vertex in $V(G) \setminus S$ has a neighbor in S. Therefore, $\bigcup_{1 \leq i \leq k} X'_i = V(G)$. Now, we remove all *other* centers except c_i from the set X'_i. Furthermore, if a vertex belongs to multiple X'_i's, we arbitrarily keep it only in a single X'_i. Let $\{X_1, X_2, \ldots, X_k\}$ be the resulting partition of $V(G)$. Observe that in the resulting clustering, centers pay a cost of zero, whereas every other vertex has a center at distance 1. Therefore, the cost of the clustering is exactly $n - k$.

In the other direction, let $(S, \{X_1, X_2, \ldots, X_k\})$ be a given restricted k-median clustering of cost $n - k$. We claim that S is a dominating set of size k. Consider any vertex $u \in V(G) \setminus S$, and suppose $u \in X_i$ corresponding to the center c_i. Since $u \notin S$, $\mathsf{dist}(u, S) \geq d(u, c_i) \geq 1$. This holds for all $n - k$ points of $V(G) \setminus S$. Now, if $u \in X_i$, and $\mathsf{dist}(u, c_i) > 1$ for some vertex $u \in V(G) \setminus S$, then this contradicts the assumption that the given clustering has cost $n - k$. This implies that every $u \in V(G) \setminus S$ has a center in S at distance exactly 1, i.e., u has a neighbor in S. This concludes the proof.

This reduction takes polynomial time. Observe that the number of points in the resulting instance is equal to n, the number of vertices in G. Therefore, if there is an algorithm for Restricted k-Median Clustering with running time subexponential in the number of points n then it would give a $2^{o(n)}$ time algorithm for Dominating set, which would refute ETH, via Proposition 3. □

4.3 SeCoCo Hardness

In this section, we use Set Cover Conjecture stated in Section 2, and give the prove of Theorem 3. Let us restate the theorem.

Theorem 3. *Assuming* Set Cover Conjecture, *for any fixed $\epsilon > 0$, there is no $2^{(1-\epsilon)n} \cdot m^{\mathcal{O}(1)}$ time algorithm for k-*Median Facility Location*, where n is the number of clients and m is the number of potential locations.*

Proof. We give a reduction from Set Cover to k-Median Facility Location.

Given an instance $(\mathcal{U}, \mathcal{S})$ of Set Cover, where $\mathcal{U} = \{u_1, \ldots, u_n\}$ and $\mathcal{S} = \{S_1, \ldots, S_m\}$, such that $S_i \subseteq \mathcal{U}$, we create an instance of k-Median Facility Location by building a bipartite graph $G = ((\mathbf{X} \cup \mathbf{F}), E)$ as follows.

- For each element $u_i \in U$, we create a client, say x_i, for $i \in \{1, \ldots, n\}$. Denote $\mathbf{X} = \{x_1, \ldots, x_n\}$.

- For each set $S_i \in \mathcal{S}$, we create a center [1], say c_i, for $i \in \{1, \ldots, m\}$. Denote $\mathbf{F} = \{c_1, \ldots, c_m\}$.

- For every $i \in \{1, \ldots, n\}$ and every $j \in \{1, \ldots, m\}$, if $u_i \in S_j$, then connect corresponding x_i and c_j with an edge of weight 1, i.e., client x_i pays cost 1 when assigned to facility c_j.

This finishes the construction of G. Now, let dist be the shortest path metric in graph G.

We show that there is set cover of size at most k if and only if there is k-median clustering of cost n.

In the forward direction, assume there is a set cover $\mathcal{S}' \subseteq \mathcal{S}$ of size at most k. Assume $\mathcal{S}' = \{S_1, \ldots, S_k\}$. For a set S_i, we make the corresponding vertex $c_i \in \mathbf{F}$ a center. Then, we create its corresponding cluster X_i as follows. We add all the points x_j such that $c_i x_j \in E$. Finally, we make the clusters X_i pairwise disjoint, by arbitrarily choosing exactly one cluster for every client, if the client is present in multiple clusters. Clearly, $\{X_1, \ldots, X_k\}$ is a partition of $\mathbf{X}$. We now calculate the cost of the obtained k-median clustering.

$$\sum_{i=1}^{k} \sum_{x \in X_i} \mathsf{dist}(c_i, x) = \sum_{i=1}^{k} |X_i| = n.$$

In the reverse direction, suppose there is a k-median clustering $\{X_1, \ldots, X_k\}$ of $\mathbf{X}$ of cost n. Let $C = \{c_1, \ldots, c_k\} \subseteq \mathbf{F}$ be a set of centers. Every client must be at distance at least 1 from its corresponding center. We claim that each client in a cluster is at distance exactly 1 from its corresponding center. Suppose not, then there exists a client with distance strictly greater than 1 from its center. The total number of clients is n. This contradicts that the cost of k-median clustering is n. Thus, every element is chosen in some set corresponding to set C. Therefore, a subfamily $\mathcal{S}' \subseteq \mathcal{S}$ corresponding to set C forms a cover of $\mathcal{U}$. Since $|C| = k$, $\mathcal{S}'$ is a cover of $\mathcal{U}$ of size at most k.

Clearly, this reduction takes polynomial time. Furthermore, observe that the number of clients in the resulting instance is same as the number of elements in $\mathcal{U}$. Therefore, if there is an $2^{(1-\epsilon)n} \cdot m^{\mathcal{O}(1)}$ time algorithm for k-Median Facility Location then it would give a $2^{(1-\epsilon)n} \cdot m^{\mathcal{O}(1)}$ time algorithm for Set Cover, which, in turn, refutes Set Cover Conjecture. □

[1] In the context of k-Median Facility Location, we use center and facility interchangeably.

4.4 A $2^n \cdot (mn)^{\mathcal{O}(1)}$ Time Algorithm for k-Median Facility Location

Let $(\mathbf{X}, \mathbf{F}, \mathsf{dist}, k)$ be a given instance of k-Median Facility Location, where $n = |\mathbf{X}|$ denotes the number of clients, and $m = |\mathbf{F}|$ denotes the number of potential locations. In this section, we give a $2^n \cdot (mn)^{\mathcal{O}(1)}$-time exact algorithm, under a mild assumption that any distance in the input is a non-negative integer that is bounded by a polynomial in the input size[2]. Let $M \coloneqq n \cdot D$, where D denotes the maximum inter-point distance in the input. Note that $M = (mn)^{\mathcal{O}(1)}$.

We define k functions $\mathsf{cost}_1, \mathsf{cost}_2, \ldots, \mathsf{cost}_k : 2^{\mathbf{X}} \to M$, where $\mathsf{cost}_i(\mathbf{Y})$ denotes the minimum cost of clustering the clients of $\mathbf{Y}$ into at most i clusters. In other words, $\mathsf{cost}_i(\mathbf{Y})$ is the optimal i-Median Facility Location cost, restricted to the instance $(\mathbf{Y}, \mathbf{F}, \mathsf{dist})$. First, notice that $\mathsf{cost}_1(\mathbf{Y})$ is simply the minimum cost of clustering all points of $\mathbf{Y}$ into a single cluster. This value can be computed in $\mathcal{O}(mn)$ time by iterating over all centers in $\mathbf{F}$, and selecting the center c that minimizes the cost $\sum_{p \in \mathbf{Y}} \mathsf{dist}(p, c)$. Thus, the values $\mathsf{cost}_1(\mathbf{Y})$ for all subsets $\mathbf{Y} \subseteq \mathbf{X}$ can be computed in $\mathcal{O}(2^n mn)$ time. Next, we have the following observation.

Observation 2. *For any $\mathbf{Y} \subseteq \mathbf{X}$ and for any $1 \leq i \leq k$,*

$$\mathsf{cost}_i(\mathbf{Y}) = \min_{\substack{A \cup B = \mathbf{Y} \\ A \cap B = \emptyset}} \mathsf{cost}_{i-1}(A) + \mathsf{cost}_1(B).$$

Note that since we are interested in clustering of $\mathbf{Y}$ into *at most* i clusters, we do not need to "remember" the set of facilities realizing $\mathsf{cost}_{i-1}(A)$ and $\mathsf{cost}_1(B)$ in Observation 2. Next, we discuss the notion of subset convolution that will be used to compute $\mathsf{cost}_i(\cdot)$ values that is faster than the naïve computation.

Subset Convolutions. Given two functions $f, g : 2^{\mathbf{X}} \to \mathbb{Z}$, the *subset convolution* of f and g is the function $(f * g) : 2^{\mathbf{X}} \to \mathbb{Z}$, defined as follows.

$$\forall \mathbf{Y} \subseteq \mathbf{X} : \qquad (f * g)(\mathbf{Y}) = \sum_{\substack{A \cup B = \mathbf{Y} \\ A \cap B = \emptyset}} f(A) \cdot g(B) \tag{4.1}$$

It is known that, given all the 2^n values of f and g in the input, all the 2^n values of $f * g$ can be computed in $\mathcal{O}(2^n \cdot n^3)$ arithmetic operations, see e.g., Theorem 10.15 in the Parameterized Algorithms book [29]. This is known as *fast subset convolution*.

[2]Since the integers are encoded in binary, this implies that the length of the encoding of any distance is $\mathcal{O}(\log(m) + \log(n))$.

Now, let $(f \oplus g)(Y) = \min_{\substack{A \cup B = Y \\ A \cap B = \emptyset}} f(A) + g(B)$. We observe that $f \oplus g$ is equal to the subset convolution $f * g$ in the integer min-sum semiring $(\mathbb{Z} \cup \{\infty\}, \min, +)$, i.e., in Equation 4.1, we use the mapping $+ \mapsto \min$, and $\cdot \mapsto +$. This, combined with a simple "embedding trick" enables one to compute all values of $f \oplus g : 2^X \to \{-N, \ldots, N\}$ in time $2^n n^{\mathcal{O}(1)} \cdot \mathcal{O}(N \log N \log\log N)$ using fast subset convolution – see Theorem 10.17 of [29]. Finally, Observation 2 implies that cost_i is exactly $\mathsf{cost}_{i-1} \oplus \mathsf{cost}_1$, and we observe that the function values are upper bounded by $n \cdot D = M$. We summarize this discussion in the following proposition.

Proposition 4. *Given all the 2^n values of cost_{i-1} and cost_1 in the input, all the 2^n values of cost_i can be computed in time $2^n n^{\mathcal{O}(1)} \cdot \mathcal{O}(M \log M \log\log M)$.*

Using Proposition 4, we can compute all the 2^n values of $\mathsf{cost}_2(\cdot)$, using the pre-computed values $\mathsf{cost}_1(\cdot)$. Then, we can use the values of $\mathsf{cost}_2(\cdot)$ and $\mathsf{cost}_1(\cdot)$ to compute the values of $\mathsf{cost}_3(\cdot)$. By iterating in this manner $k - 1 \leq n$ times, we compute the values of $\mathsf{cost}_k(\cdot)$ for all 2^n subsets of k, and the overall time is upper bounded by $2^n mn^{\mathcal{O}(1)} \cdot \mathcal{O}(M \log M \log\log M)$, which is $2^n \cdot (mn)^{\mathcal{O}(1)}$, if $M = (mn)^{\mathcal{O}(1)}$. Note that $\mathsf{cost}_k(\mathbf{X})$ corresponds to the optimal cost of clustering. Finally, the computed values of the functions $\mathsf{cost}_i(\cdot)$ can be used to also compute a clustering $\{X_1, X_2, \ldots, X_k\}$ of $\mathbf{X}$, and the corresponding centers $\{c_1, c_2, \ldots, c_k\}$. We omit the straightforward details. This proves the following theorem.

Theorem 4. k-Median Facility Location *can be solved optimally in $2^n \cdot (mn)^{\mathcal{O}(1)}$ time, assuming the distances are integers that are bounded by polynomial in the input size.*

Chapter 5

Parameterized Categorical Capacitated Clustering

In this chapter, we study the parameterized complexity of Categorical Capacitated k-Median Clustering. It is a generalization of Categorical k-Median Clustering. The results mentioned in this chapter appeared in Article 2.

Categorical Capacitated k-Median Clustering

Input: A multiset $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ of n points from Σ^m over a finite alphabet, a positive integer k, a nonnegative integer B, and positive integers p and q such that $p \leq q$.

Task: Decide whether there is a partition $\mathcal{X} = \{X_1, \ldots, X_k\}$ of $\mathbf{X}$, where $p \leq |X_i| \leq q$, and vectors $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\}$ in Σ^m such that

$$cost(C, \mathcal{X}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}) \leq B.$$

Recall that the sets $X_1, \ldots, X_k$ are called clusters and the vectors $\mathbf{c}_1, \ldots, \mathbf{c}_k$ are centers.

Parameterized algorithms for the vanilla variant of Categorical Capacitated k-Median Clustering (without constraints on the sizes of clusters) were given by Fomin, Golovach, and Panolan in [37]. One of the main results of their paper is the theorem providing an algorithm of running time $2^{\mathcal{O}(B \log B)} \cdot (nm)^{\mathcal{O}(1)}$ for vanilla clustering over the binary field. In other words, the problem is fixed-parameter tractable (FPT) parameterized by B. The main question that we address in this chapter is whether clustering constraints impact the problem's parameterized complexity.

Our main result is that Categorical Capacitated k-Median Clustering is fixed-parameter tractable when parameterized by the budget B and the alphabet size. More precisely, we show the following:

Theorem 5. Categorical Capacitated k-Median Clustering *can be solved in* $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ *time.*

Theorem 5 generalizes the result (Theorem 1) in [37]. Interestingly, for approximation algorithms, introducing clustering constraints makes the problem much more computationally challenging. However, from the parameterized complexity perspective, adding constraints on the sizes of clusters does not change the complexity of the problem. We note that Theorem 5 is tight in the sense that it is unlikely that the dependence on the alphabet size could be made polynomial because the results of Fomin, Golovach, and Simonov [38] imply that Categorical Capacitated k-Median Clustering is $\mathsf{W}[1]$-hard when parameterized by B and m.

We also observe that Categorical Capacitated k-Median Clustering is NP-complete even for binary points, $k = 2$ and $p = q = \frac{n}{2}$.

Theorem 6. *For every fixed integer constant* $c \geq 0$, Categorical Capacitated k-Median Clustering *is* NP*-complete for* $k = 2$*, binary points and* $q - p \leq c$.

Theorem 5 can be used to establish fixed-parameter tractability of several other variants of constrained clustering discussed in the literature. In some applications, it is natural to require that the sizes of clusters be approximately equal, see, e.g., [78].

We consider variants of Categorical Capacitated k-Median Clustering, where the input contains additional parameters besides a set $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ of n points over a finite alphabet Σ^m and integers k and B, and the task is to find clusters $X_1, \ldots, X_k$ and medians $\mathbf{c}_1, \ldots, \mathbf{c}_k \in \Sigma^m$ such that $\sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}) \leq B$ and the sizes of the clusters satisfy special balance properties.

- In Balanced Categorical k-Median Clustering, we are additionally given a nonnegative integer δ and it should hold that $||X_i| - |X_j|| \leq \delta$ for all $i, j \in \{1, \ldots, k\}$, that is, the sizes of clusters can differ by at most δ.

- In Factor-Balanced Categorical k-Median Clustering, we are given a real $\alpha \geq 1$ and it is required that $|X_i| \leq \alpha |X_j|$ for all $i, j \in \{1, \ldots, k\}$, that is, the ratio of the sizes of the clusters is upper bounded by α.

By making use of Theorem 5, we prove that Balanced Categorical k-Median Clustering and Factor-Balanced Categorical k-Median Clustering are solvable in time $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$.

Corollary 2. Balanced Categorical k-Median Clustering *and* Factor-Balanced Categorical k-Median Clustering *are solvable in time* $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$.

Finally, we discuss kernelization for these problems. In particular, we show that Balanced Categorical k-Median Clustering admits a polynomial kernel under the combined parameterization by k, B and δ.

Theorem 7. Balanced Categorical k-Median Clustering *admits a kernel, where the output set has* $O(k(B + \delta k))$ *points from a space of dimension* $\mathcal{O}(B(B + k))$ *over an alphabet of size at most* $B + k$.

In [37, Theorem 3], Fomin, Golovach and Panolan proved that Categorical k-Median Clustering for binary points does not admit a polynomial kernel when parameterized by B, unless $\mathsf{NP} \subseteq \mathsf{coNP/poly}$. This immediately implies the following proposition.

Proposition 5. Categorical Capacitated k-Median Clustering *(*Balanced Categorical k-Median Clustering *and* Factor-Balanced Categorical k-Median Clustering*, respectively) has no polynomial kernel when paramterized by* B*, unless* $\mathsf{NP} \subseteq \mathsf{coNP/poly}$*, even if* $\Sigma = \{0, 1\}$.

That is, neither of the considered problems has a polynomial kernel when parameterized by B only, unless $\mathsf{NP} \subseteq \mathsf{coNP/poly}$.

Organization of the chapter In Section 5.1, we show that Categorical Capacitated k-Median Clustering is NP-complete for $k = 2$ and binary matrices even if the clusters are required to be of the same size. In Section 5.2, we show our main result by constructing an FPT algorithm for Categorical Capacitated k-Median Clustering parameterized by $B + |\Sigma|$. In Section 5.3, we discuss Balanced Categorical k-Median Clustering and Factor-Balanced Categorical k-Median Clustering. Further, in Section 5.4, we discuss kernelization for clustering problems with size constraints.

5.1 Hardness of Clustering

In [32], Feige proved that Categorical k-Median Clustering is NP-complete for $k = 2$ and binary points, that is, for the case $\Sigma = \{0, 1\}$. This result immediately implies that Categorical Capacitated k-Median Clustering is also NP-complete for $k = 2$ and binary points.

To see it, note that an instance $(\mathbf{X}, \Sigma, k, B)$ of Categorical k-Median Clustering is equivalent to the instance $(\mathbf{X}, \Sigma, k, B, p, q)$ of Categorical Capacitated k-Median Clustering for $p = 1$ and $q = n$. However, we would like to underline that Categorical Capacitated k-Median Clustering is NP-hard even if $p = q$. For this, we use some details of the hardness proof of Feige [32].

Feige proved that Categorical k-Median Clustering is NP-hard by showing a reduction from the Max-Cut problem [32]. In Max-Cut, we are given a graph G and a nonnegative integer ℓ, and the task is to find a *cut* $(S, \overline{S})$, that is, a partition of the vertex set into a set S and its complement $\overline{S} = V(G) \setminus S$ such that the size of the cut, i.e., the number of edges between S and $\overline{S}$ is at least ℓ. The reduction constructed by Feige has the property given in the following lemma.

Lemma 4 ([32])**.** *There is a polynomial time reduction from* Max-Cut *to* Categorical k-Median Clustering *that computes from an instance* (G, ℓ) *of* Max-Cut *an instance* $(\mathbf{X}, \Sigma, 2, B)$ *of* Categorical k-Median Clustering *where* $\Sigma = \{0, 1\}$*, such that the following holds: if* (G, ℓ) *is a yes-instance of* Max-Cut *with a cut* $(S, \bar{S})$ *of size at least* ℓ*, then* $(\mathbf{X}, \Sigma, 2, B)$ *is a yes-instance of* Categorical k-Median Clustering *that has a solution* $\{X_1, X_2\}$ *with the property that* $|X_1|/|X_2| = |S|/|\overline{S}|$.

Theorem 6. *For every fixed integer constant* $c \geq 0$*,* Categorical Capacitated k-Median Clustering *is* NP*-complete for* $k = 2$*, binary points and* $q - p \leq c$.

Proof. We show the theorem by a reduction from Max-Cut that is well-known to be NP-complete [43]. Given an instance (G, ℓ) of Max-Cut, we construct an auxiliary instance $(G', 2\ell)$ of Max-Cut, where G' is the union of two disjoint copies G_1 and G_2 of G. Then for the constructed instance $(G', 2\ell)$ we can use as a black box the algorithm of Feige [32] from Lemma 4 to produce the instance $(\mathbf{X}, \Sigma, 2, B)$ of Categorical k-Median Clustering with $\Sigma = \{0, 1\}$. We further set $p = q = |V(G)|$ and consider the instance $(\mathbf{X}, \Sigma, 2, B, p, q)$ of Categorical Capacitated k-Median Clustering. Clearly, $q - p \leq c$. We show that (G, ℓ) is a yes-instance of Max-Cut if and only if $(\mathbf{X}, \Sigma, 2, B, p, q)$ is a yes-instance of Categorical Capacitated k-Median Clustering.

In the forward direction, assume that (G, ℓ) is a yes-instance of Max-Cut and let $(S, \bar{S})$ be a cut of size at least ℓ. Let S_1 and S_2 be the copies of S in G_1 and G_2, respectively. We now consider $S' \subseteq V(G')$ such that $S' = S_1 \cup (V(G_2) \setminus S_2)$. Clearly, $\overline{S'} = (V(G_1) \setminus S_1) \cup S_2$ and $(S', \overline{S'})$ is a cut of G' of size at least 2ℓ. Moreover, $|S'| = |S_1| + |V(G_2) \setminus S_2| = |S_2| + |V(G_1) \setminus S_1| = |\overline{S'}|$. Hence, $(G', 2\ell)$ is a yes-instance of Max-Cut with a solution $(S', \overline{S'})$ that has the property that $|S'| = |\overline{S'}|$. By Lemma 4, $(\mathbf{X}, \Sigma, 2, B)$ is a yes-instance of Categorical k-Median Clustering that has a solution $\{X_1, X_2\}$ such that $|X_1| = |X_2|$. This implies that $p \leq |X_1|, |X_2| \leq q$. Therefore, $\{X_1, X_2\}$ is also solution for the instance $(\mathbf{X}, \Sigma, 2, B, p, q)$ of Categorical Capacitated k-Median Clustering. Thus, $(\mathbf{X}, \Sigma, 2, B, p, q)$ is a yes-instance of Categorical Capacitated k-Median Clustering.

In the reverse direction, suppose that $(\mathbf{X}, \Sigma, 2, B, p, q)$ is a yes-instance of Categorical Capacitated k-Median Clustering. Then there is a 2-clustering $\{X_1, X_2\}$ for $\mathbf{X}$ of cost at most B. This means that $(\mathbf{X}, \Sigma, 2, B)$ is a yes-instance of Categorical k-Median Clustering because $(\mathbf{X}, \Sigma, 2, B)$ is obtained from $(G', 2\ell)$ by a polynomial reduction from Lemma 4, $(G', 2\ell)$ is a yes-instance of Max-Cut, that is, G' has a cut of size at least 2ℓ. Since G' is a disjoint union of two identical copies of G, each copy has a cut of size at least l. Therefore, (G, ℓ) is a yes-instance of Max-Cut. This completes the hardness proof. □

5.2 FPT Algorithm for Parameterization by B and the Alphabet Size

In this section, we show that Categorical Capacitated k-Median Clustering is FPT when parameterized by B and $|\Sigma|$. Our main result is Theorem 5 that we restate here.

Theorem 5. Categorical Capacitated k-Median Clustering *can be solved in* $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ *time.*

Note that this result is tight in the sense that it is unlikely that the dependence on the alphabet size could be made polynomial. It was shown in [38], that Categorical k-Median Clustering is W[1]-hard when parameterized by B and the number of rows m of the input matrix if $\Sigma = \mathbb{Z}$, i.e., for an infinite alphabet. However, it is straightforward to see that this result holds for $\Sigma = \{0, \ldots, n-1\}$ because our measure is the Hamming distance. For each row of the input matrix, we can replace the original symbols by the symbols of $\Sigma = \{0, \ldots, n-1\}$ in such a way that the original symbols in the row are

the same if and only if the new symbols are the same. Clearly, this replacement gives an equivalent instance. This immediately leads to the following proposition.

Proposition 6. Categorical Capacitated k-Median Clustering *is* W[1]*-hard when parameterized by* B *and* m.

The remaining part of the section contains the proof of Theorem 5. The proof is constructive. In Subsection 5.2.1, we introduce some notation and show technical claims that are used by the algorithm.

5.2.1 Definitions and Technical Lemmata

In this section, we introduce some additional terminology that will be used in this work. Recall, that a partition $\mathcal{X} = \{X_1, \ldots, X_k\}$ of X is a *k-clustering* of X.

Definition 1. *An initial cluster is an inclusion maximal set* $J \subseteq \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ *such that all the points in* J *are identical.*

We say a cluster X_i of $\mathcal{P}$ is *simple* if $X_i \subseteq J$ and X_i is *composite*, otherwise, that is, if X_i contains some $\mathbf{x}_h$, and $\mathbf{x}_j$ in X such that $\mathbf{x}_h \neq \mathbf{x}_j$.

We start by making the following observation about medians of sufficiently big (in B) clusters.

Observation 3. *Let* $\{X_1, \ldots, X_k\}$ *be a* k*-median clustering of collection of points* $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ *of* Σ^m *of cost at most* B*, and let* $|X_i| \geq B + 1$ *for some* $i \in \{1, \ldots, k\}$*. Then for all vectors* $\mathbf{c}_1, \ldots, \mathbf{c}_k \in \Sigma^m$ *such that* $\sum_{h=1}^{k} \sum_{x \in X_h} \mathsf{dist}_0(\mathbf{c}_h, \mathbf{x}) \leq B$*,* $\mathbf{c}_i = \mathbf{x}$ *for at least* $|X_i| - B$ *equal points* $\mathbf{x}$ *in* $\mathbf{X}$*. Moreover, if* $|X_i| \geq 2B + 1$*, then* $\mathbf{c}_i$ *is unique.*

Proof. To show the first part of the claim, assume that $\mathbf{c}_i \in \Sigma^m$ is distinct from at least $B + 1$ points $\mathbf{x}$ of X_i. Then

$$B \geq \sum_{h=1}^{k} \sum_{\mathbf{x} \in X_h} \mathsf{dist}_0(\mathbf{c}_h, \mathbf{x}) \geq \sum_{\mathbf{x} \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}) \geq B + 1;$$

a contradiction. For the second part of the claim, note that if $|X_i| \geq 2B + 1$, then $\mathbf{c}_i$ should coincide with more than half of the points $\mathbf{x}$ in X_i and, therefore, the choice of $\mathbf{c}_i$ is unique. □

We use the following simple observation about the number of composite clusters and the number of initial clusters having elements in the composite clusters of a solution.

Observation 4. *Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be a collection of n points $\mathbf{X}$ of Σ^m with the partition $\mathcal{J} = \{J_1, \ldots, J_s\}$ of $\mathbf{X}$ into initial clusters. Let also $\mathcal{X} = \{X_1, \ldots, X_k\}$ be a k-median clustering of $\mathbf{X}$ of cost at most B. Then $\mathcal{X}$ contains at most B composite clusters and $\mathcal{J}$ has at most $2B$ initial clusters with nonempty intersections with the composite clusters of $\mathcal{X}$.*

Proof. Let $\mathbf{c}_1, \ldots, \mathbf{c}_k$ be medians such that $\sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}) \leq B$. Note that if X_i is a composite cluster for some $i \in \{1, \ldots, k\}$, then $\mathbf{c}_i$ is distinct from $\mathbf{x}$ for at least one $\mathbf{x} \in X_i$ and $\sum_{x \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}) \geq 1$. Therefore, $\mathcal{X}$ contains at most B composite clusters. For the second claim, notice that if $\mathcal{J}$ has $t \geq B$ initial clusters with nonempty intersections with composite clusters, then because $\mathcal{X}$ has at most B composite clusters, for at least $t - B$ of these initial clusters J_j, $\mathbf{x} \neq \mathbf{c}_i$ for $\mathbf{x} \in J_j$ and all the medians $\mathbf{c}_i$ of composite clusters. Hence, $t \leq 2B$. □

Let $J \subseteq \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be an initial cluster. Due to size constraints, it may happen that a k-median clustering $\{X_1, \ldots, X_k\}$ with several simple clusters $X_i \subseteq J$ provides a solution. This means, that we should partition a subset of J into blocks of bounded size. To verify whether we are able to create such a partition, we use the following observation.

Observation 5. *Let p and q be positive integers, $p \leq q$. A finite set S can be partitioned into h subsets such that each of them has size at least p and at most q if and only if $\left\lceil \frac{|S|}{q} \right\rceil \leq h \leq \left\lfloor \frac{|S|}{p} \right\rfloor$.*

Proof. If S can be partitioned into h subsets of size at least p and at most q, then, trivially, $ph \leq |S|$ and $qh \geq |S|$, i.e., $\left\lceil \frac{|S|}{q} \right\rceil \leq h \leq \left\lfloor \frac{|S|}{p} \right\rfloor$. If $\left\lceil \frac{|S|}{q} \right\rceil \leq h \leq \left\lfloor \frac{|S|}{p} \right\rfloor$, then S has h disjoint subsets $S_1, \ldots, S_h$ of size p. Then the remaining $|S| - ph$ elements can be greedily added to these subsets without exceeding the upper bound q on the size. □

Let $\mathcal{J} = \{J_1, \ldots, J_s\}$ be the partition of $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ into initial clusters. For a k-median clustering $\mathcal{X} = \{X_1, \ldots, X_k\}$, we define the graph $G(\mathcal{X}, \mathcal{J})$ as the intersection graph of the sets of $\mathcal{X}$ and $\mathcal{J}$, that is, $G(\mathcal{X}, \mathcal{J})$ is the bipartite graph with the set of vertices $\mathcal{X} \cup \mathcal{J}$ such that for every $i \in \{1, \ldots, k\}$ and $j \in \{1, \ldots, s\}$, X_i and J_j are adjacent if and only if $X_i \cap J_j \neq \emptyset$. We show that we can assume $G(\mathcal{X}, \mathcal{J})$ to be a forest. This can be proved using an Integer Linear Program or flow formulation of the clustering problem with given medians. For simplicity, we provide a direct proof.

Lemma 5. *Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be a collection of points of n points of Σ^m with the partition $\mathcal{J} = \{J_1, \ldots, J_s\}$ of $\mathbf{X}$ into initial clusters. Also, let $\mathcal{X} = \{X_1, \ldots, X_k\}$ be a k-clustering for $\mathbf{X}$. Then there is a k-clustering $\mathcal{X}' = \{X'_1, \ldots, X'_k\}$ such that (i)*

$|X_i| = |X'_i|$ *for all* $i \in \{1, \ldots, k\}$, *(ii)* $\mathsf{cost}(X'_1, \ldots, X'_k) \leq \mathsf{cost}(X_1, \ldots, X_k)$, *and (iii)* $G(\mathcal{X}', \mathcal{J})$ *is a forest.*

Proof. Assume that $\mathcal{X}' = \{X'_1, \ldots, X'_k\}$ is a k-median clustering for $\mathbf{X}$ satisfying conditions (i) and (ii) such that the number of edges of $G(\mathcal{X}', \mathcal{J})$ is minimum. Denote by $\mathbf{c}_1, \ldots, \mathbf{c}_k$ optimal medians for $X'_1, \ldots, X'_k$. We claim that $G(\mathcal{X}', \mathcal{J})$ is a forest.

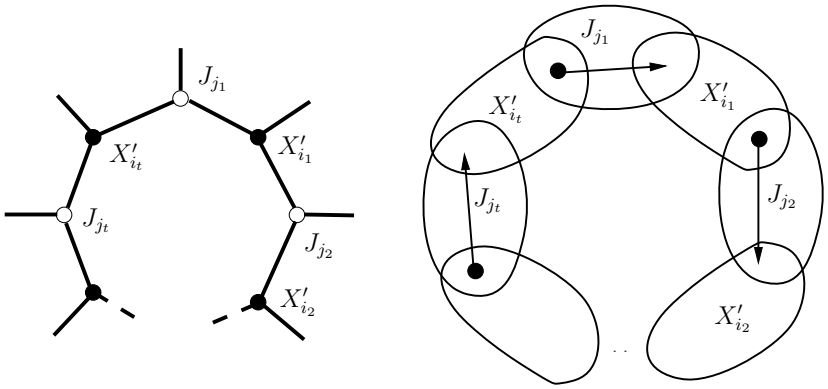

Figure 5.1: A cycle in $G(\mathcal{X}', \mathcal{J})$ and the cluster rearrangement scheme.

The proof is by contradiction. Assume that $G(\mathcal{X}', \mathcal{J})$ has a cycle. This means that there are distinct $i_1, \ldots, i_t \in \{1, \ldots k\}$ and distinct $j_1, \ldots, j_t \in \{1, \ldots, s\}$ such that $X'_{i_h} \cap J_{j_h} \neq \emptyset$ and $X'_{i_h} \cap J_{j_{h+1}} \neq \emptyset$ for all $h \in \{1, \ldots, t\}$; here and further in the proof, we assume that $j_{t+1} = j_1$ and $i_{t+1} = i_1$ (see Figure 5.1(a)).

For $h \in \{1, \ldots, s\}$, denote by $\mathbf{y}_h$ the point coinciding with $\mathbf{x}_{h'}$ for $\mathbf{x}_{h'} \in J_h$. We observe that either

$$\sum_{h=1}^{t} (\mathsf{dist}_0(\mathbf{c}_{i_h}, \mathbf{y}_{j_h}) + \mathsf{dist}_0(\mathbf{c}_{i_h}, \mathbf{y}_{j_{h+1}})) \geq 2 \sum_{h=1}^{t} \mathsf{dist}_0(\mathbf{c}_{i_h}, \mathbf{y}_{j_h}) \tag{5.1}$$

or

$$\sum_{h=1}^{t} (\mathsf{dist}_0(\mathbf{c}_{i_h}, \mathbf{y}_{j_h}) + \mathsf{dist}_0(\mathbf{c}_{i_h}, \mathbf{y}_{j_{h+1}})) \geq 2 \sum_{h=1}^{t} \mathsf{dist}_0(\mathbf{c}_{i_h}, \mathbf{y}_{j_{h+1}}) \tag{5.2}$$

because the sums of the left and right parts of inequalities (5.1) and (5.2) are the same.

We assume without loss of generality that (5.1) holds, as the second case is symmetric. This means that

$$\sum_{h=1}^{t} \mathsf{dist}_0(\mathbf{c}_{i_h}, \mathbf{y}_{j_{h+1}}) \geq \sum_{h=1}^{t} \mathsf{dist}_0(\mathbf{c}_{i_h}, \mathbf{y}_{j_h}). \tag{5.3}$$

We iteratively modify $\mathcal{X}'$ by moving a representative of J_{j_h} in $X_{i_{h-1}}$ to X_{i_h} for $h \in \{2,\ldots,t+1\}$, that is, representatives are moved cyclically without changing the cluster sizes (see Figure 5.1(b)). We show that this procedure does not increase the clustering cost with respect to the medians $\mathbf{c}_1,\ldots,\mathbf{c}_k$.

Formally, we construct the k-clusterings $\mathcal{X}^{(0)},\mathcal{X}^{(1)},\ldots$, where $\mathcal{X}^{(p)} = \{X_1^{(p)},\ldots,X_k^{(p)}\}$ for $p=0,1,\ldots$, starting from $\mathcal{X}^{(0)}=\mathcal{X}'$ while $J_{j_{h+1}} \cap X_{i_h}^{(p)} \neq \emptyset$ for all $h \in \{1,\ldots,t\}$.

Assume that $\mathcal{X}^{(p)} = \{X_1^{(p)},\ldots,X_k^{(p)}\}$ is constructed and $J_{j_{h+1}} \cap X_{i_h}^{(p)} \neq \emptyset$ for all $h \in \{1,\ldots,t\}$. For every $h \in \{1,\ldots,t\}$, let $x_{i'_h} \in J_{j_{h+1}} \cap X_{i_h}^{(p)}$.

$$X_{i_h}^{(p+1)} = (X_{i_h}^{(p)} \setminus \{x_{i'_h}\}) \cup \{x_{i'_{h-1}}\}$$

for all $h \in \{1,\ldots,t\}$ assuming that $i'_0 = i'_t$, and we set $X_q^{(p+1)} = X_q^{(p)}$ for $q \in \{1,\ldots,k\} \setminus \{i_1,\ldots,i_t\}$. Clearly, $|X_i^{(p+1)}| = |X_q^{(p)}|$ for all $i \in \{1,\ldots,r\}$. We have that

$$\begin{aligned}\Big(\sum_{i=1}^{k}\sum_{\mathbf{x}\in X_i^{(p)}} \mathsf{dist}_0(\mathbf{c}_i,\mathbf{x})\Big) - \Big(\sum_{i=1}^{k}\sum_{\mathbf{x}\in X_i^{(p+1)}} \mathsf{dist}_0(\mathbf{c}_i,\mathbf{x})\Big) &= \sum_{h=1}^{t}(\mathsf{dist}_0(\mathbf{c}_{i_h},\mathbf{x}_{i'_h}) - \mathsf{dist}_0(\mathbf{c}_{i_h},\mathbf{x}_{i'_{h-1}}))\\ &= \sum_{h=1}^{t}(\mathsf{dist}_0(\mathbf{c}_{i_h},\mathbf{y}_{j_{h+1}}) - \mathsf{dist}_0(\mathbf{c}_{i_h},\mathbf{y}_{j_h}))\\ &= \Big(\sum_{h=1}^{t}(\mathsf{dist}_0(\mathbf{c}_{i_h},\mathbf{y}_{j_{h+1}})\Big) - \Big(\sum_{h=1}^{t}\mathsf{dist}_0(\mathbf{c}_{i_h},\mathbf{y}_{j_h})\Big) \geq 0,\end{aligned}$$

where the last inequality follows from (5.3). This means that the cost of the k-clustering $\mathcal{X}^{(p+1)}$ with respect to the medians $\mathbf{c}_1,\ldots,\mathbf{c}_k$ is at most the cost of $\mathcal{X}^{(p)}$ with respect to the same medians.

The next k-clustering $\mathcal{X}^{(p+1)}$ is constructed from $\mathcal{X}^{(p)}$ if $J_{j_{h+1}} \cap X_{i_h}^{(p)} \neq \emptyset$ for all $h \in \{1,\ldots,t\}$. Thus, the sequence is finite, and for the last k-clustering $\mathcal{X}^{(q)}$, there is $h \in \{1,\ldots,t\}$ such that $J_{j_{h+1}} \cap X_{i_h}^{(q)} = \emptyset$, that is, $X_{i_h}^{(q)}$ and $J_{j_{h+1}}$ are not adjacent in $G(\mathcal{X}^{(q)},\mathcal{J})$. Note that the rearrangement of elements of clusters does not create new adjancencies in $G(\mathcal{X}^{(q)},\mathcal{J})$ because no cluster gets representatives of an initial cluster that had no representatives in it. We conclude that $G(\mathcal{X}^{(q)},\mathcal{J})$ has less edges than $G(\mathcal{X}',\mathcal{J})$ but this contradicts the choice of $\mathcal{X}'$. Therefore, $G(\mathcal{X}',\mathcal{J})$ is a forest and $\mathcal{X}'$ satisfies conditions (i)–(iii) as required. □

Next, we show that, given a collection of n points $\mathbf{X}$, we can list all potential medians for a k-median clustering of cost at most B in FPT when B and $|\Sigma|$ are parameters. We

show this by making use of the nontrivial result of Marx [68] about the enumeration of subhypergraphs with bounded partial edge cover. This result already proved to be very useful for designing FPT algorithms for clustering problems [37, 38].

Recall that a hypergraph $\mathcal{H}$ is a pair $(V, \mathcal{E})$, where V is a set of *vertices* and $\mathcal{E}$ is a family of subsets of V called *hyperedges*. Similarly to graphs, we denote by $V(\mathcal{H})$ the set of vertices and by $\mathcal{E}(\mathcal{H})$ the set of hyperedges. For a vertex v, we denote by $\mathcal{E}_{\mathcal{H}}(v)$ the set of hyperedges containing v, that is, $\mathcal{E}_{\mathcal{H}}(v) = \{E \in \mathcal{E}(\mathcal{H}) \mid v \in E\}$.

Let $\mathcal{G}$ be a hypergraph and let $U \subseteq V(\mathcal{G})$. We say that a hypergraph $\mathcal{H}$ *appears at* U *as a subhypergraph* if there is a bijection $\pi\colon V(\mathcal{H}) \to U$ with the property that for every $E \in \mathcal{E}(\mathcal{H})$, there is $E' \in \mathcal{E}(\mathcal{G})$ such that $\pi(E) = E' \cap U$.

A *fractional hyperedge cover* of a hypergraph $\mathcal{H}$ is a function $\varphi\colon \mathcal{E}(\mathcal{H}) \to [0,1]$ such that for every vertex $v \in V(\mathcal{H})$, $\sum_{E \in \mathcal{E}_{\mathcal{H}}(v)} \varphi(E) \geq 1$, that is, the sum of the values assigned by f of the hyperedges containing v is at least one. The *fractional cover number* $\rho^*(\mathcal{H})$ of $\mathcal{H}$ is the minimum value $\sum_{E \in \mathcal{E}(\mathcal{H})} \varphi(E)$ taken over all fractional hyperedge covers φ of H.

Proposition 7 ([68])**.** *Let $\mathcal{H}$ be a hypergraph with fractional cover number $\rho^*(\mathcal{H})$, and let $\mathcal{G}$ be a hypergraph whose hyperedges have size at most ℓ. There is an algorithm that enumerates, in $|V(\mathcal{H})|^{\mathcal{O}(|V(\mathcal{H})|)} \cdot \ell^{|V(\mathcal{H})|\rho^*(\mathcal{H})+1} \cdot |\mathcal{E}(\mathcal{G})|^{\rho^*(\mathcal{H})+1} \cdot |V(\mathcal{G})|^2$ time, every $U \subseteq V(\mathcal{G})$ where $\mathcal{H}$ appears at U as subhypergraph in $\mathcal{G}$.*

We apply this result similarly to [38] and, therefore, only briefly sketch the proof of the following lemma.

Lemma 6. *There is an algorithm that, given a collection of n points $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ and a nonnegative integer B, in $2^{\mathcal{O}(B \log B)} \cdot |\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ time outputs a set $\mathcal{M}(\mathbf{X}, B) \subseteq \Sigma^m$ of size $2^{\mathcal{O}(B \log B)} \cdot |\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ such that for every k-clustering $\{X_1, \ldots, X_k\}$ for $\mathbf{X}$ of cost at most B, there are $\mathbf{c}_1, \ldots, \mathbf{c}_k \in \mathcal{M}(\mathbf{X}, B)$ such that $\sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j) \leq B$.*

Proof. Let $\mathcal{S}$ be the set of distinct points s of $\mathbf{X}$. Initially, we set $\mathcal{M}(\mathbf{X}, B) := \mathcal{S}$.

For every $\mathbf{s} \in \mathcal{S}$, we construct the hypergraph $\mathcal{G}_{\mathbf{s}}$ with the vertex set $\{1, \ldots, m\}$ with hyperedges corresponding to the points of $\mathbf{X}$ at Hamming distance at most B from $\mathbf{s}$: for every $\mathbf{x}_i \in \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ such that $\mathsf{dist}_0(\mathbf{s}, \mathbf{x}_i) \leq B$, we introduce the hyperedge

$$E_i = \{j \mid 1 \leq j \leq m \text{ and } \mathbf{x}_i[j] \neq \mathbf{s}[j]\},$$

that is, the hyperedge contains indices, where $\mathbf{s}$ differs from $\mathbf{x}_i$. Note that $|E_i| \leq B$.

Consider an arbitrary k-clustering $\{X_1, \ldots, X_k\}$ for $\mathbf{X}$ of cost at most B. Let $X_i \in \{X_1, \ldots, X_k\}$ and let $\mathbf{s} \in \mathcal{S}$ be such that $\mathbf{s} = \mathbf{x}_j$ for some $\mathbf{x}_j \in X_i$. Let also $\mathbf{c}_i \in \Sigma^m$ be an optimal median for X_i, that is, $\sum_{\mathbf{x}_j \in I_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j)$ is minimum. Notice that if $|X_i| \geq B + 1$, then by Observation 3, every feasible median for X_i is a point of $\mathbf{X}$ and these points are already placed in $\mathcal{M}(\mathbf{X}, B)$. Also, if $\mathbf{c}_i = \mathbf{s}$, then $\mathbf{c}_i \in \mathcal{M}(\mathbf{X}, B)$. Assume that $|X_i| \leq B$ and $\mathbf{c}_i \neq \mathbf{s}$. Clearly, $\mathsf{dist}_0(\mathbf{c}_i, \mathbf{s}) \leq B$. Moreover, for any $\mathbf{x}_j \in X_i$, $\mathsf{dist}_0(\mathbf{s}, \mathbf{x}_j) \leq B$. This holds trivially if $\mathbf{s} = \mathbf{x}_j$. Otherwise, if $\mathbf{s} \neq \mathbf{x}_j$, we have that $\mathsf{dist}_0(\mathbf{s}, \mathbf{x}_j) \leq \mathsf{dist}_0(\mathbf{c}_i, \mathbf{s}) + \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j) \leq \sum_{\mathbf{x}_j \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j) \leq B$. Let

$$D = \{j \mid 1 \leq j \leq m \text{ and } \mathbf{c}_i[j] \neq \mathbf{s}[j]\},$$

that is, D is the set of indices where $\mathbf{s}$ differs from the median $\mathbf{c}_i$.

We consider the hypergraph $\mathcal{H}_i$ with the vertex set D whose edges correspond to the points $\mathbf{x}_j$ for $\mathbf{x}_j \in X_i$. For each $\mathbf{x}_j \in X_i$, we construct the hyperedge

$$F_j = \{h \mid h \in D \text{ and } \mathbf{x}_j[h] \neq \mathbf{s}[h]\},$$

that is, each hyperedge contains indices from D, where $\mathbf{s}$ differs from $\mathbf{x}_j$. We claim that the fractional cover number $\rho^*(\mathcal{H}_i) \leq 2$.

To show this, we define the function $\varphi(F) = \frac{2}{|\mathcal{E}(\mathcal{H}_i)|}$ for every hyperedge F of $\mathcal{H}_i$. We prove that φ is a fractional hyperedge cover of $\mathcal{H}_i$. Thus, we have to show that for every $j \in D$, $\sum_{F \in \mathcal{E}_{\mathcal{H}_i}(j)} \varphi(F) \geq 1$. This is equivalent to proving that for every $j \in D$, at least half of the hyperedges of $\mathcal{H}_i$ contain j. Assume that this is not the case, i.e., there is $j \in D$ such that more than half of hyperedges do not contain j. This mean that for more than half of points $\mathbf{x}_h$ for $\mathbf{x}_h \in X_i$, $\mathbf{s}[j] = \mathbf{x}_h[j] = s$. However, by the definition of D, $\mathbf{s}[j] \neq \mathbf{c}_i[j]$ and, therefore, $\mathbf{c}_i[j] \neq s$. This contradicts the assumption that $\mathbf{c}_i$ is an optimal median for X_i because replacing the current value $\mathbf{c}_i[j]$ by s decreases the cost. Hence, φ is a fractional hyperedge cover. Then

$$\rho^*(\mathcal{H}_i) \leq \sum_{F \in \mathcal{E}(\mathcal{H}_i)} \varphi(F) = \sum_{F \in \mathcal{E}(\mathcal{H}_i)} \frac{2}{|\mathcal{E}(\mathcal{H}_i)|} = 2.$$

Observe that $\mathcal{H}_i$ appears in $\mathcal{G}_\mathbf{s}$ at D because for each $\mathbf{x}_j \in X_i$, $\mathsf{dist}_0(\mathbf{s}, \mathbf{x}_j) \leq B$, that is, for every $\mathbf{x}_j \in X_i$, $\mathcal{G}_\mathbf{s}$ contains the hyperedge E_j corresponding to $\mathbf{x}_j$; the mapping $\pi \colon V(\mathcal{H}_i) \to D$ is the identity function.

We obtain that $\mathcal{H}_i$ is a hypergraph with the fractional cover number at most 2 that appears in $\mathcal{G}_\mathbf{s}$ at D. Notice that, given $\mathbf{s}$ and D, we can list the vectors over Σ^m that differ from $\mathbf{s}$ in the indices from D and the total number of such vectors is at most $|\Sigma|^B$

because $|D| \leq B$. Then $\mathbf{c}_i$ appears in this list. This leads to the following algorithm. We consider all hypergraphs $\mathcal{H}$ on at most B vertices with at most B hyperedges. Then for each $\mathcal{H}$ and every $\mathbf{s} \in \mathcal{S}$, we use the algorithm of Marx from Proposition 7 to enumerate every $D \subseteq V(\mathcal{G}_{\mathbf{s}})$ where $\mathcal{H}$ appears in $\mathcal{G}_D$ as subhypergraph. Then for every D, we list the vectors that differ from $\mathbf{s}$ in the indices from D by brute force. Then these vectors are included in $\mathcal{M}(\mathbf{X}, B)$.

For given $\mathcal{H}$ and $\mathbf{s}$, the sets D can be enumerated in time $2^{\mathcal{O}(B \log B)} \cdot B^{2B+1} \cdot n^3 \cdot m^2$ by Proposition 7. Then generating the vectors that differ from $\mathbf{s}$ in D can be done in $|\Sigma|^B \cdot n^{\mathcal{O}(1)}$ time as we can assume that $|\Sigma| \leq n$. However, we need $2^{\mathcal{O}(B^2)}$ time to generate all hypergraphs with at most B vertices and at most B hyperedges. This gives the total running time $2^{\mathcal{O}(B^2)} \cdot |\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ and the same bound on the size of $\mathcal{M}(\mathbf{X}, B)$.

The running time can be improved by proving that there is a subhypergraph $\mathcal{H}'_i$ of $\mathcal{H}$ with $V(\mathcal{H}'_i) = V(\mathcal{H}_i)$ and $\mathcal{E}(\mathcal{H}'_i) \subseteq \mathcal{E}(\mathcal{H}_i)$ of size $\mathcal{O}(\log B)$ (more precisely, of size at most $160 \ln B$) such that $\rho^*(\mathcal{H}'_i) \leq 4$. The proof is identical to the proof of Claim 18 of [38] (see also Proposition 6.3 of [68]) and we omit it here.

Then we consider all hypergraphs $\mathcal{H}$ with at most B vertices and at most $160 \ln B$ hyperedges. The total number of these hypergraphs is $2^{\mathcal{O}(B \log B)}$. Then, in the same way as above, for each $\mathcal{H}$ and every $\mathbf{s} \in \mathcal{S}$, we use the algorithm of Marx from Proposition 7 to enumerate every $D \subseteq V(\mathcal{G}_{\mathbf{s}})$ where $\mathcal{H}$ appears in $\mathcal{G}_D$ as subhypergraph. For every D, the vectors that differ from $\mathbf{s}$ in the indices from D are enumerated by brute force and each vector is added to $\mathcal{M}(\mathbf{A}, B)$ unless it is already included in the set. The total running time is $2^{\mathcal{O}(B \log B)} \cdot |\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ and the number of vectors in $\mathcal{M}(\mathbf{A}, B)$ is $2^{\mathcal{O}(B \log B)} \cdot |\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$. □

5.2.2 Algorithm

Let $(\mathbf{X}, \Sigma, k, B, p, q)$ be an instance of Categorical Capacitated k-Median Clustering with $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$. First, we compute the partition $\mathcal{J} = \{J_1, \ldots, J_s\}$ of $\mathbf{X}$ into initial clusters.

Choosing potential medians

By the next step , we restrict the set of considered medians. For this, we apply Lemma 6 and construct the set $\mathcal{M} = \mathcal{M}(\mathbf{X}, B)$ of potential medians. Recall that this set has size $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ and can be computed in $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ time. For a

k-clustering $\mathcal{X} = \{X_1, \ldots, X_k\}$, we define the *minimum cost (with respect to $\mathcal{M}$)*, as

$$\min\{\sum_{i=1}^{k} \sum_{\mathbf{x}_j \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j) \mid \mathbf{c}_1, \ldots, \mathbf{c}_k \in \mathcal{M}\}.$$

If $(\mathbf{X}, \Sigma, k, B, p, q)$ is a yes-instance, then it has a solution such that the medians are in $\mathcal{M}$ by Lemma 6. Therefore, solving the problem is equivalent to finding a clustering of minimum cost at most B with respect to $\mathcal{M}$. Throughout this section, whenever we say that $\mathcal{X}$ is a clustering of minimum cost, we mean that the cost is minimum with respect to $\mathcal{M}$.

Structure of solutions

Further, we argue that we can consider solutions of a special structure whose nontrivial part involves bounded number of initial clusters.

By Lemma 5, if $(\mathbf{X}, \Sigma, k, B, p, q)$ is a yes-instance, then there is a solution $\mathcal{X} = \{X_1, \ldots, X_k\}$ to the instance such that the intersection graph $G(\mathcal{X}, \mathcal{J})$ of the initial clusters and the clusters of the solution is a forest. We call such a solution (or k-clustering) *acyclic*. To solve the problem, we check whether the considered instance has an acyclic solution. To simplify notation, we assume that all solutions considered further on are acyclic.

By Observation 4, any k-clustering for $\mathbf{X}$ of cost at most B has at most B composite clusters. We consecutively consider $t = 0, \ldots, \min\{B, k\}$, and for each t, we verify whether there is a solution $\mathcal{X} = \{X_1, \ldots, X_k\}$ with exactly t composite clusters. If we find such a solution, then we return the yes-answer and stop. Otherwise, if we have no solution for all the values of t, we report that $(\mathbf{X}, \Sigma, k, B, p, q)$ is a no-instance. From now on, we assume that nonnegative $t \leq \min\{B, k\}$ is fixed.

It is convenient to consider the special case $t = 0$ separately. If $t = 0$, then a solution $\mathcal{X}$ has no composite cluster, that is, the clusters of the solution form partitions of the initial clusters. Observe that $\mathsf{cost}(\mathcal{X}) = 0 \leq B$ in this case. By Observation 5, the initial clusters can be partitioned into k blocks of size at least p and at most q, if and only if there are positive integers $h_1, \ldots, h_s$ such that $k = h_1 + \cdots + h_s$ and $\left\lceil \frac{|J_i|}{q} \right\rceil \leq h_i \leq \left\lfloor \frac{|J_i|}{p} \right\rfloor$ for every $i \in \{1, \ldots, s\}$. For every $i \in \{1, \ldots, s\}$, we verify whether $\left\lceil \frac{|J_i|}{q} \right\rceil \leq \left\lfloor \frac{|J_i|}{p} \right\rfloor$. If at least one of the inequalities does not hold, the required $h_1, \ldots, h_s$ do not exist. Otherwise, we observe that positive integers $h_1, \ldots, h_s$ such that $k = h_1 + \cdots + h_s$ and $\left\lceil \frac{|J_i|}{q} \right\rceil \leq h_i \leq \left\lfloor \frac{|J_i|}{p} \right\rfloor$ for every $i \in \{1, \ldots, s\}$ exist if and only if $\sum_{i=1}^{s} \left\lceil \frac{|J_i|}{q} \right\rceil \leq k \leq \sum_{i=1}^{s} \left\lfloor \frac{|J_i|}{p} \right\rfloor$. Then we verify

the last inequality.

From now, we assume that $t \geq 1$. Note that we also can assume that $B \geq 1$ because for $B = 0$, no cluster of a solution can be composite.

By Observation 4, there are at most $2B$ initial clusters with nonempty intersections with the composite clusters of a solution $\mathcal{X}$. Since $G(\mathcal{X}, \mathcal{J})$ is a forest, it is easy to observe that at least $t + 1$ initial clusters have nonempty intersections with the composite clusters. We consider $\ell = t + 1, \ldots, 2B$, and for each ℓ, we check whether there is a solution $\mathcal{X} = \{X_1, \ldots, X_k\}$ such that exactly ℓ initial clusters have nonempty intersections with the composite clusters of $\mathcal{X}$. If we find such a solution, then we return the yes-answer and stop. Otherwise, if we have no solution for all the values of ℓ, we report that $(\mathbf{X}, \Sigma, k, B, p, q)$ is a no-instance. From now, we assume that positive $t + 1 \leq \ell \leq 2B$ is given.

Recall that we are looking for an acyclic solution $\mathcal{X} = \{X_1, \ldots, X_k\}$, that is, $G(\mathcal{X}, \mathcal{J})$ is required to be a forest. Let $\mathcal{X}$ be such a k-clustering. Let $\mathcal{X}' \subseteq \mathcal{X}$ be the set of composite clusters and let $\mathcal{J}' \subseteq \mathcal{J}$ be the set of initial clusters having nonempty intersections with the composite clusters. Recall that $|\mathcal{X}'| = t$ and $|\mathcal{J}'| = \ell$ by our assumptions. Note also that the leaves of $G(\mathcal{X}', \mathcal{J}')$ are initial clusters and every connected component of this forest contains at least three vertices.

We consider all forests F on $t + \ell$ vertices such that (i) each connected component of F has at least three vertices, and (ii) F admits a bipartition (U, W) of its vertex set with $|U| = t$ and $|W| = \ell$ such that the leaves of F are in W. Since $t \leq B$ and $\ell \leq 2B$, the number of such forests is $2^{\mathcal{O}(B)}$ [72] and they can be listed in $2^{\mathcal{O}(B)}$ time (see, e.g., [81]). Note that since the leaves are required to be in W, the bipartition (U, W) is unique. From now on, we assume that F together with the bipartition (U, W) is given.

Colorful solutions

Recall that we are looking for a solution such that exactly ℓ initial clusters have nonempty intersections with composite clusters of the solution. We use the *color coding* technique of Alon, Yuster, and Zwick [5] (see [29, Chapter 5] for the detailed introduction) to highlight the initial clusters with nonempty intersections with clusters of a potential solution. We first give a Monte Carlo algorithm with false negatives and then explain how to derandomize it. We color the initial clusters by ℓ colors uniformly at random. We say that a k-clustering $\mathcal{X} = \{X_1, \ldots, X_k\}$ of cost at most B is a *colorful* solution if the initial clusters with nonempty intersections with the clusters of $\mathcal{X}$ have distinct colors. As it is standard for color coding, the algorithm exploits the property that if there

is a solution such that exactly ℓ initial clusters have nonempty intersections with the composite clusters of the solution, then the probability that these ℓ clusters get distinct colors in a random coloring is at least $\frac{\ell!}{\ell^\ell} \geq e^{-\ell} \geq e^{-2B}$. Therefore, with probability at least e^{-2B}, a yes-instance admits a colorful solution.

Finding colorful solutions

Our next task is to explain how to check whether there is a colorful solution for a given random coloring $\psi\colon \mathcal{J} \to \{1,\ldots,\ell\}$ such that $G(\mathcal{X}',\mathcal{J}')$, where $\mathcal{X}'$ is the set of composite clusters in the solution and $\mathcal{J}' \subseteq \mathcal{J}$ is the set of initial clusters having nonempty intersections with the composite clusters, is isomorphic to F. For this, we use dynamic programming over F. Recall that F is given together with the bipartition (U,W) of its vertex set, where the leaves are in W. To construct our dynamic programming algorithm, we formally define k-clusterings forming solutions as follows.

Definition 2 (Feasible k-clustering)**.** *For a given forest F with the bipartition (U,W) of its vertex set, we say that an acyclic k-clustering $\mathcal{X} = \{X_1,\ldots,X_k\}$ of $\mathbf{X}$ is a* feasible *(with respect to F and the parameters t and ℓ) if the following holds:*

(i) $p \leq |X_i| \leq q$ *for* $i \in \{1,\ldots,k\}$,

(ii) the set $\mathcal{X}' \subseteq \mathcal{X}$ *of composite clusters has size* t *and the set* $\mathcal{J}' \subseteq \mathcal{J}$ *of initial clusters having nonempty intersections with the composite clusters has size* ℓ,

(iii) the initial clusters in $\mathcal{J}'$ *are colored by distinct colors by* ψ, *and*

(iv) $G(\mathcal{X}',\mathcal{J}')$ *is isomorphic to F with an isomorphism that bijectively maps* $\mathcal{X}'$ *to* U *and* $\mathcal{J}'$ *to* W.

Then the problem of finding a colorful solution boils down to checking whether there is a feasible k-clustering of cost at most B.

To proceed with the algorithm, we need some auxiliary notation. For a set of colors $P \subseteq \{1,\ldots,\ell\}$, we use $\mathcal{J}(P) \subseteq \mathcal{J}$ to denote the subset of initial clusters with the colors from P and $C(P) \subseteq \{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ is used to denote the set of points in the initial clusters with their colors in P, that is, $C(P) = \bigcup_{J\in\mathcal{J}(P)} J$. We also denote $\mathbf{X}(P)$, the subcollection of $\mathbf{X}$ with the points $\mathbf{x}_i$ such that $\mathbf{x}_i \in C(P)$.

It is common to do bottom-up dynamic programming over rooted trees. However, F may be disconnected. We argue that, given partial solutions for the connected components of F, we can combine them and solve the problem for F. Denote by $F_1,\ldots,F_f$ the

connected components of F. Let $U_i = V(F_i) \cap U$ and $W_i = V(F_i) \cap W$ for $i \in \{1, \ldots, f\}$. Let also $t_i = |U_i|$ and $\ell_i = |W_i|$ for $i \in \{1, \ldots, f\}$.

For $i \in \{1, \ldots, f\}$, $P \subseteq \{1, \ldots \ell\}$ and a positive integer $h \leq k$, denote by $\omega_i(P, h)$ the minimum cost of an h-clustering for $\mathbf{X}(P)$ that is feasible with respect to F_i and the parameters t_i and ℓ_i if $|P| = \ell_i$. We assume that $\omega_i(P, h) = +\infty$ if $|P| \neq \ell_i$ or no h-clustering is feasible. Thus, the functions $\omega_i(P, h)$ represent partial solutions for $F_1, \ldots, F_f$.

We show that if we are given the tables of values of $\omega_i(P, h)$, then we can verify whether there is a feasible k-clustering of cost at most B.

Lemma 7. *Given the values $\omega_i(P, h)$ for all $i \in \{1, \ldots, f\}$, $P \subseteq \{1, \ldots \ell\}$ and positive integers $h \leq k$, it can be decided in time $2^{\mathcal{O}(B)} \cdot n^2$ whether there is a feasible k-clustering for $\mathbf{X}$ of cost at most B with respect to F, t and ℓ.*

Proof. To give the intuition behind the proof, observe that a feasible k-clustering of cost at most B with respect to F, t and ℓ exists if and only if there are positive integers $h_1, \ldots, h_f$ such that $h_1 + \cdots + h_f = k$ and a partition $\{P_1, \ldots, P_f\}$ of $\{1, \ldots, \ell\}$ such that

$$\omega_1(P_1, h_1) + \cdots + \omega_f(P_f, h_f) \leq B$$

because in a feasible clustering the initial clusters in $\mathcal{J}'$ are colored by distinct colors. This leads to the following dynamic programming algorithm.

For $j \in \{1, \ldots, f\}$, $P \subseteq \{1, \ldots, \ell\}$, let $F^{(j)}$ be the disjoint union of $F_1, \ldots, F_j$, $t^{(j)} = t_1 + \cdots + t_j$ and $\ell^{(j)} = \ell_1 + \cdots + \ell_j$. For $j \in \{1, \ldots, f\}$, $P \subseteq \{1, \ldots, \ell\}$ and positive integer h, denote by $w^{(j)}(P, h)$ the minimum cost of an h-clustering for $\mathbf{X}(P)$ that is feasible with respect to $F^{(j)}$, $t^{(j)}$ and $\ell^{(j)}$ if $|P| = \ell^{(j)}$; we also assume that $w^{(j)}(P, h) = +\infty$ if $|P| \neq \ell^{(j)}$ or there is no feasible h-clustering.

Notice that $\omega_1(P, h) = w^{(1)}(P, h)$ and $w^{(f)}(P, h)$ is the minimum cost of an h-clustering for $\mathbf{X}(P)$ that is feasible with respect to F, t and ℓ. Thus, $w^{(f)}(\{1, \ldots, \ell\}, k) \leq B$ if and only if there is a feasible k-clustering for $\mathbf{X}$ of cost at most B with respect to F, t and ℓ.

We compute the values of $w^{(j)}(P, h)$ for $j = 1, 2, \ldots, f$. As we observed, $w^{(1)}(P, h) = \omega_1(P, h)$. To compute $w^{(j)}(P, h)$ for $j \geq 2$, we use the following recurrence:

$$w^{(j)}(P, h) = \min\{\omega_j(Y, h') + w^{(j-1)}(P \setminus Y, h - h') \mid 1 \leq h' < h \text{ and } \emptyset \neq Y \subset P\}; \quad (5.4)$$

we also assume that $w^{(j)}(P, h) = +\infty$ if the set in the right part of (5.4) is empty.

The correctness of (5.4) is proved in the standard way by showing the two opposite

inequalities. Let $P \subseteq \{1,\dots,\ell\}$. To simplify notation, assume that $\{J_1,\dots,J_{s'}\}$ are initial clusters with colors from P. Let also $h \leq k$ be a positive integer.

In the forward direction, suppose that $|P| = \ell^{(j)}$ and $\{X_1,\dots,X_h\}$ is an h-clustering for $\mathbf{X}(P)$ that is feasible with respect to $F^{(j)}$, $t^{(j)}$ and $\ell^{(j)}$ of minimum cost.

Let $\mathcal{X}' \subseteq \{X_1,\dots,X_h\}$ be the set of composite clusters and let $\mathcal{J}' \subseteq \{J_1,\dots,J_{s'}\}$ be the set of initial clusters having nonempty intersections with the composite clusters.

Recall that $|\mathcal{X}'| = t^{(j)}$, $|\mathcal{J}'| = \ell^{(j)}$, and the initial clusters in $\mathcal{J}'$ are colored by distinct colors. Consider an isomorphism α that bijectively maps the vertices of $G(\mathcal{X}',\mathcal{J}')$ to the vertices of F with the property that the vertices of $\mathcal{X}'$ are mapped to $\bigcup_{i=1}^{(j)} U_i$ and $\mathcal{J}'$ are mapped to $\bigcup_{i=1}^{(j)} W_i$. Then ℓ_j clusters of $\mathcal{J}'$ are mapped to W_j.

Denote by $Y \subset P$ the set of their colors. Clearly, $|Y| = \ell_j$ and $|P \setminus Y| = \ell^{(j)} - \ell_j = \ell^{j-1}$. Notice that the clusters of $\mathcal{X}'$ that are mapped to U_j are composed of elements of initial clusters with colors from Y and no other composite cluster contains an element of an initial cluster with a color from Y. To simplify notation, assume that the clusters $X_1,\dots,X_{h'}$ contain elements of the initial clusters with the colors from Y and $X_{h'+1},\dots,X_h$ are the clusters containing elements of the initial clusters with the colors from $P \subseteq Y$. Then we have that $\{X_1,\dots,X_{h'}\}$ is a feasible h'-clustering for $\mathbf{X}(Y)$ with respect to F_j, t_j and ℓ_j. Similarly, we obtain that $\{X_{h'+1},\dots,X_h\}$ is a feasible $(h-h')$-clustering for $\mathbf{X}(P \setminus Y)$ with respect to $F^{(j-1)}$, $t^{(j-1)}$ and $\ell^{(j-1)}$. Thus, $w^{(j)}(P,h) \geq \omega_j(Y,h') + w^{(j-1)}(P \setminus Y, h-h')$ and, therefore,

$$w^{(j)}(P,h) \geq \min\{\omega_j(Y,h') + w^{(j-1)}(P \setminus Y, h-h') \mid 1 \leq h' < h \text{ and } \emptyset \neq Y \subset P\}. \quad (5.5)$$

If either $|P| \neq \ell^{(j)}$ or there is no an h-clustering for $\mathbf{X}(P)$ that is feasible with respect to $F^{(j)}$, $t^{(j)}$ and $\ell^{(j)}$, then $w^{(j)}(P,h) = +\infty$ and (5.5) is trivial.

To show the opposite inequality, let nonempty $Y \subseteq P$ and positive $h' < h$ be such that the right part of (5.4) is minimum. If $\omega_j(Y,h') + w^{(j-1)}(P \setminus Y, h-h') = +\infty$, then the required inequality holds trivially. Assume that this is not the case. Then $|Y| = \ell_j$, $|P \setminus Y| = \ell^{(j-1)}$, there is an h'-clustering $\mathcal{X}^{(1)}$ for $\mathbf{X}(Y)$ of cost $\omega_j(Y,h')$ that is feasible with respect to F_j, t_j and ℓ_j, and there is an $(h-h')$-clustering $\mathcal{X}^{(2)}$ for $\mathbf{X}(P \setminus Y)$ of cost $w^{(j-1)}(P \setminus Y, h-h')$ that is feasible with respect to $F^{(j-1)}$, $t^{(j-1)}$ and $\ell^{(j-1)}$.

Consider $\mathcal{X} = \mathcal{X}^{(1)} \cup \mathcal{X}^{(2)}$ and observe that this is an h-clustering for $\mathbf{X}(P)$ that is feasible with respect to $F^{(j)}$, $t^{(j)}$ and $\ell^{(j)}$. This means that $w^{(j)}(P,h) \leq \omega_j(Y,h') + w^{(j-1)}(P \setminus$

$Y, h - h'$). By the choice of Y and h',

$$w^{(j)}(P,h) \leq \min\{\omega_j(Y,h') + w^{(j-1)}(P \setminus Y, h - h') \mid 1 \leq h' < h \text{ and } \emptyset \neq Y \subset P\}. \quad (5.6)$$

Combining (5.5) and (5.6), we obtain that the recurrence (5.4) holds.

Finally, we compute $w^{(f)}(P,h)$ for all $P \subseteq \{1,\ldots,\ell\}$ and all positive $h \leq k$. In particular, we find $w^{(f)}(\{1,\ldots,\ell\},k)$ and verify whether this value is at most B.

To evaluate the running time, note that to compute the table of values of $w^{(j)}(P,h)$ by (5.4), we consider all nonempty P of size at most ℓ and the nonempty subsets $Y \subset P$.

This means that we consider at most 3^ℓ pairs of sets. Also, we consider all positive $h \leq k$ and $h' \leq h$, that is, at most k^2 pairs of integers. Since $\ell \leq 2B$ and $k \leq n$, the computations can be done in $2^{\mathcal{O}(B)} \cdot n^2$ time. Since $f \leq t \leq B$, the total running time is $2^{\mathcal{O}(B)} \cdot n^2$. □

The final step is to compute the partial solutions for $F_1,\ldots,F_f$. By Lemma 7, we have to compute the tables of values of $\omega_i(P,h)$ for all $i \in \{1,\ldots,f\}$, nonempty $P \subseteq \{1,\ldots,\ell\}$ and positive $h \leq k$. For this, we use the fact that $F_1,\ldots,F_f$ are trees and this allows us to use dynamic programming over these trees.

Lemma 8. *Let T be a tree with a bipartition (U,W) of its vertex set such that $t' = |U| \leq t$, $\ell' = |W| \leq \ell$ and the leaves of T are in W. For a given $P \subseteq \{1,\ldots,\ell\}$ with $|P| = \ell'$ and positive $h \leq k$, the minimum cost of a feasible h-clustering for $\mathbf{X}(P)$ with respect to T, t' and ℓ' can be found in $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ time.*

Proof. We select a vertex $z \in U$ as a root of T. This selection defines a parent-child relation on the set of vertices. For a vertex $x \in V(T)$, we denote by T_x the subtree of T induced by the descendants of x (including the vertex itself). For $x \in V(T)$, let $t_x = V(T_x) \cap U$ and $\ell_x = V(T_x) \cap W$. For every $x \in V(T)$, we compute the tables of auxiliary values depending on whether $x \in U$ or $x \in W$.

For a set of colors $Z \subseteq P$, $J \in \mathcal{J}(Z)$ and $J' \subseteq J$, we use $\mathcal{J}(Z)/J'$ to denote the set of clusters obtained from the initial clusters of $\mathcal{J}(Z)$ by the replacement of J by $J'' = J \setminus J'$ if $J' \subset J$ and $\mathcal{J}(Z)/J' = \mathcal{J}(Z) \setminus \{J\}$ if $J' = J$.

We assume that the clusters of $\mathcal{J}(Z)/J'$ have the inherited colors. We also write $\mathbf{X}(Z)/J'$ to denote the subcollection of $\mathbf{X}(Z)$ obtained by the deletion of the points from J'. Note that $\mathcal{J}(Z)/J'$ is the set of initial clusters for $\mathbf{X}(Z)/J'$.

Suppose that $x \in W$. For every positive integer $h' \leq h$, every $Y \subseteq P$, every $c \in Y$, every $J \in \mathcal{J}(Y)$ and every nonnegative integer $j \leq |J|$, we define $\omega_x^{(1)}(h', Y, c, J, j)$. For technical reasons, it is convenient to define this function for leaves separately.

Definition 3 (Partial solution for a leaf $x \in W$)**.** *Let x be a leaf. We define $\omega_x^{(1)}(h', \{c\}, c, J, j)$ as the minimum cost of an h'-clustering for $\mathbf{X}(Y)/J'$, where $J' \subseteq J$ of size j, such that all the clusters are simple, and $\omega_x^{(1)}(h', Y, c, J, j) = +\infty$ if $Y \neq \{c\}$.*

Definition 4 (Partial solution for an internal $x \in W$)**.** *If x is an internal vertex of T, then $\omega_x^{(1)}(h', Y, c, J, j)$ is the minimum cost of an h'-clustering $\mathcal{X} = \{X_1, \ldots, X_{h'}\}$ for $\mathbf{X}(Y)/J'$, where $J' \subset J$ of size j, such that*

(i) $p \leq |X_i| \leq q$ *for* $i \in \{1, \ldots, h'\}$,

(ii) the set $\mathcal{X}' \subseteq \mathcal{X}$ of composite clusters has size t_x, and the set $\mathcal{J}' \subseteq \mathcal{J}(Y)/J'$ of initial clusters having nonempty intersections with the composite clusters has size ℓ_x,

(iii) $|Y| = \ell_x$ *and the initial clusters in $\mathcal{J}'$ are colored by distinct colors by ψ,*

(iv) $G(\mathcal{X}', \mathcal{J}')$ is isomorphic to T_x with an isomorphism α that bijectively maps $\mathcal{X}'$ to U_x, $\mathcal{J}'$ to W_x, and

(v) $J \setminus J' \in \mathcal{J}'$, $\alpha(J \setminus J') = x$ *and* $\psi(J \setminus J') = c$.

In both cases, we assume that $\omega_x^{(1)}(h, Y, c, J, j) = +\infty$ if there is no such an h'-clustering.

Informally, $\omega_x^{(1)}(h', Y, c, J, j)$ is the minimum cost of an h'-clustering for $\mathbf{X}(Y)/J'$ that is feasible with respect to T_x, t_x and ℓ_x with the additional assumption that we take j elements of J colored by c to include to the composite cluster that corresponds to the parent of x (see Figure 5.2). Observe that the value of $\omega_x^{(1)}(h', Y, c, J, j)$ does not depend on the choice of J'. Notice also that we have the special case when $U_x = \emptyset$, i.e. when x is a leaf because we have no composite clusters in this case. Then we form h' simple clusters from the initial clusters $\mathcal{J}(Y)/J'$.

Now we define the fuction $\omega_x^{(2)}(h', Y, j, \mathbf{s})$ for $x \in U$ for every positive integer $h' \leq h$, every $Y \subseteq X$, every nonnegative integer $j \leq q$, and every $\mathbf{s} \in \mathcal{M}$.

Definition 5 (Partial solution for $x \in U$)**.** $\omega_x^{(2)}(h', Y, j, \mathbf{s})$ *is the minimum cost of an h'-clustering $\mathcal{X} = \{X_1, \ldots, X_{h'}\}$ for $\mathbf{X}(Y)$ such that*

(i) the cost of X_1 is computed with respect to the median $\mathbf{s}$, that is, the cost equals $\sum_{\mathbf{x}_i \in X_1} \mathsf{dist}_0(\mathbf{s}, \mathbf{x}_i)$,

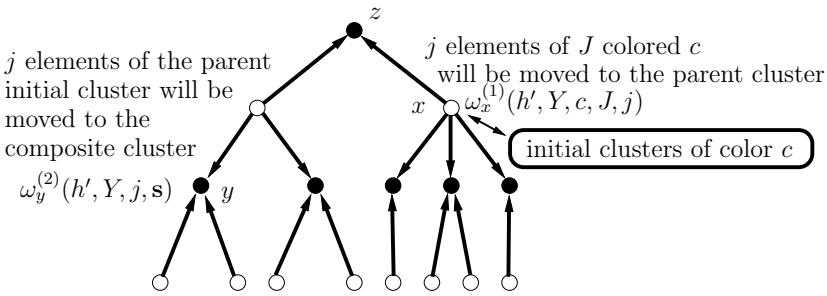

Figure 5.2: The general scheme of dynamic programming over T. The vertices of U corresponding to composite clusters are shown by black bullets and the vertices of W corresponding to initial clusters are white. The arrows show which initial clusters are contributing to composite clusters. Note that J is a (part of) initial cluster of color c and the remaining initial clusters of color c (including the rest of the cluster containing J) are split into simple clusters.

(ii) $p - j \leq |X_1| \leq q - j$ *and* $p \leq |X_i| \leq q$ *for* $i \in \{2, \ldots, h'\}$,

(iii) for the set of composite clusters $\mathcal{X}' \subseteq \mathcal{X}$, $\mathcal{X}'' = \mathcal{X}' \cup \{X_1\}$ *has size* t_x, *and the set* $\mathcal{J}' \subseteq \mathcal{J}(Y)$ *of initial clusters having nonempty intersections with the clusters from* $\mathcal{X}''$ *has size* ℓ_x,

(iv) $|Y| = \ell_x$ *and the initial clusters in* $\mathcal{J}'$ *are colored by distinct colors by* ψ,

(v) $G(\mathcal{X}'', \mathcal{J}')$ *is isomorphic to* T_x *with an isomorphism* α *that bijectively maps* $\mathcal{X}''$ *to* U_x, $\mathcal{J}'$ *to* W_x, *and* $\alpha(X_1) = x$.

In the same way as above for other functions, it is assumed that $\omega_x^{(2)}(h', Y, j, \mathbf{s}) = +\infty$ if there is no such an h'-clustering.

Informally, $\omega_x^{(2)}(h', Y, j, \mathbf{s})$ is the minimum cost of an h'-clustering for $\mathbf{X}(Y)$ that is feasible with respect to T_x, t_x and ℓ_x, where the specific cluster X_1 associated with x is required to have $\mathbf{s}$ as its median and "misses" j elements (see Figure 5.2). Notice that it is not required that $\mathbf{s}$ is optimal for X_1. However, in the future, X_1 is going to be complemented by j elements of an initial cluster corresponding to the parent of x, unless x is a root. Note also that X_1 is not a composite cluster if x has a unique child, but because X_1 is expected to be complemented by other elements, X_1 is counted as a composite cluster in the definition of $\omega_x^{(2)}(h', Y, j, \mathbf{s})$.

Now we explain how to compute the table of values of $\omega_x^{(1)}(h', Y, c, J, j)$ and $\omega_x^{(2)}(h', Y, j, \mathbf{s})$. First, we compute $\omega_x^{(1)}(h', Y, c, J, j)$ for leaves.

Claim 5.2.1. *For every leaf* x *of* T, $\omega_x^{(1)}(h', Y, c, J, j)$ *can be computed in* $\mathcal{O}(n)$ *time.*

Proof. If $Y \neq \{c\}$, $\omega_x^{(1)}(h', Y, c, J, j) = +\infty$ by the definition. Assume that $Y = \{c\}$. Let $J' \subseteq J$ be a set of size j. We compute $\hat{\mathcal{J}} = \mathcal{J}(Y)/J'$ in $\mathcal{O}(n)$ time. Then $\omega_x^{(1)}(h', Y, c, J, j) = 0$ if every set in $\hat{\mathcal{J}}$ can be partitioned into clusters of size at least p and at most q in such a way that the total number of clusters is h', and $\omega_x^{(1)}(h', Y, c, J, j) = +\infty$ otherwise. We apply Observation 5. First, we verify whether every $\hat{J} \in \hat{\mathcal{J}}$ can be partitioned into clusters of size at least p and at most q by checking whether $\left\lceil \frac{|\hat{J}|}{q} \right\rceil \leq \left\lfloor \frac{|\hat{J}|}{p} \right\rfloor$. If this holds, then we observe that we can obtain exactly h' clusters in total if and only if $\sum_{\hat{J} \in \hat{\mathcal{J}}} \left\lceil \frac{|\hat{J}|}{q} \right\rceil \leq h' \leq \sum_{\hat{J} \in \hat{\mathcal{J}}} \left\lfloor \frac{|\hat{J}|}{p} \right\rfloor$. Since checking of these conditions can be done in $\mathcal{O}(n)$ time, the total running time is $\mathcal{O}(n)$. □

Next, we explain how to compute $\omega_x^{(1)}(h', Y, c, J, j)$ for internal vertices if the tables of values of $\omega_y^{(2)}(\cdot, \cdot, \cdot, \cdot)$ are given for all children y of x. This is done by an auxiliary dynamic programming algorithm.

Claim 5.2.2. *Let $x \in W$ be an internal vertex of T and assume that the table of values of $\omega_y^{(2)}(\cdot, \cdot, \cdot, \cdot)$ is computed for every child y of x. Then $\omega_x^{(1)}(h', Y, c, J, j)$ can be computed in $2^{\mathcal{O}(B \log B)} |\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ time.*

Proof. Let $h' \leq h$, $Y \subseteq P$, $c \in Y$, $J \in \mathcal{J}(Y)$ and $j \leq |J|$. If $j = |J|$, then we immediately set $\omega_x^{(1)}(h', Y, c, J, j) = +\infty$ because we have no proper $J' \subset J$ of size j. Also, if $|Y| \neq \ell_x$ or $\psi(J) \neq c$, then $\omega_x^{(1)}(h', Y, c, J, j) = +\infty$ by definition. Assume that $j < |J|$, $J' \subset J$ of size j, $\psi(J) = c$ and $|Y| = \ell_x$. Let $\hat{J} = J \setminus J'$. We denote by $y_1, \ldots, y_f$ the children of x in T.

Consider the initial clusters of color c. By the definition of $\omega_x^{(1)}(h', Y, c, J, j)$, we are interested in an h'-clustering, where the initial clusters of color c distinct from J are split into simple clusters and, possibly, some parts of J also form simple clusters. For a nonnegative integers $\hat{h} \leq h'$ and $\hat{j} \leq |\hat{J}|$, we define $w(\hat{h}, \hat{j})$ to be 0 if the initial clusters of $\hat{\mathcal{J}} = \mathcal{J}(\{c\})/(J \setminus J'')$, where $J'' \subseteq \hat{J}$ of size $\hat{j}$ can be partitioned into $\hat{h}$ simple clusters of size at least p and at most q, and we set $w(\hat{h}, \hat{j}) = +\infty$ otherwise. To compute $w(\hat{h}, \hat{j})$, we use Observation 5 similarly to the proof of Claim 5.2.1. Namely, we verify whether every $\tilde{J} \in \hat{\mathcal{J}}$ can be partitioned into clusters of size at least p and at most q by checking whether $\left\lceil \frac{|\tilde{J}|}{q} \right\rceil \leq \left\lfloor \frac{|\tilde{J}|}{p} \right\rfloor$, and then we check whether $\sum_{\tilde{J} \in \hat{\mathcal{J}}} \left\lceil \frac{|\tilde{J}|}{q} \right\rceil \leq \hat{h} \leq \sum_{\tilde{J} \in \hat{\mathcal{J}}} \left\lfloor \frac{|\tilde{J}|}{p} \right\rfloor$. Since $\hat{h} \leq h' \leq h$, the values of $w(\hat{h})$ can be computed in $\mathcal{O}(n^2)$ time.

Observe that by the definition of $\omega_x^{(1)}(h', Y, c, J, j)$, the points of $\hat{J}$ should be included in f composite clusters associated with the children of x in an h'-clustering for $\mathbf{X}(Y)/J'$. In particular, if $|\hat{J}| < f$, it cannot be done and $\omega_x^{(1)}(h', Y, c, J, j) = +\infty$ by the definition. From now, we assume that $|\hat{J}| \geq f$.

For $i \in \{1,\dots,f\}$, denote by $T^{(i)}$ the subtree of T induced by $\{x\} \cup \bigcup_{i'=1}^{i} V(T_{y_{i'}})$, set $U^{(i)} = U \cap V(T^{(i)})$ and $W^{(i)} = W \cap V(T^{(i)})$. Let also $t^{(i)} = |U^{(i)}|$ and $\ell^{(i)} = |W^{(i)}|$ for $i \in \{1,\dots,f\}$. For each $i \in \{1,\dots,f\}$, each nonnegative $\hat{h} \leq h'$, each positive $\hat{j} \leq |J| - j$, and every $c \in Z \subseteq Y$, define the auxiliary values $w^{(i)}(\hat{h},\hat{j},Z)$.

Definition 6 (Auxiliary partial solution)**.** *$w^{(i)}(\hat{h},\hat{j},Z)$ is the minimum cost of $\hat{h}$-clustering $\mathcal{X} = \{X_1,\dots,X_{\hat{h}}\}$ for $\mathbf{X}(Z)/(J \setminus J'')$, where $J'' \subseteq \hat{J}$ of size $\hat{j}$, such that*

(i) $p \leq |X_{i'}| \leq q$ *for* $i' \in \{1,\dots,\hat{h}\}$,

(ii) the set $\mathcal{X}' \subseteq \mathcal{X}$ of composite clusters has size $t^{(i)}$, and the set $\mathcal{J}' \subseteq \mathcal{J}(Y)/(J \setminus J'')$ of initial clusters having nonempty intersections with the composite clusters has size $\ell^{(i)}$,

(iii) $|Z| = \ell^{(i)}$ *and the initial clusters in $\mathcal{J}'$ are colored by distinct colors by ψ,*

(iv) $G(\mathcal{X}',\mathcal{J}')$ is isomorphic to $T^{(i)}$ with an isomorphism α that bijectively maps $\mathcal{X}'$ to $U^{(i)}$, $\mathcal{J}'$ to $W^{(i)}$, and

(v) $J'' \in \mathcal{J}'$, $\alpha(J'') = x$ *and* $\psi(J'') = c$.

We also follow the same convention as above that $w^{(i)}(\hat{h},\hat{j},Z) = +\infty$ if either there is no $\hat{h}$-clustering satisfying (i)–(v). Observe that, by the definition, $\omega_x^{(1)}(h',Y,c,J,j) = w^{(f)}(h',|J|-j,Y)$. Therefore, we compute the tables of values of $w^{(i)}(\cdot,\cdot,\cdot)$ for $i = 1,\dots,f$.

To initiate the computation of $w^{(i)}(\cdot,\cdot,\cdot)$, it is convenient to formally define this function for $i = 0$. We set

$$w^{(0)}(\hat{h},\hat{j},Z) = \begin{cases} w(\hat{h},\hat{j}) & \text{if } Z = \{c\}, \\ +\infty & \text{otherwise.} \end{cases}$$

For $\mathbf{s} \in M$, denote $d(\mathbf{s}) = \mathsf{dist}_0(\mathbf{s},\mathbf{a}_j)$ for $j \in J$. Then to compute $w^{(i)}(\hat{h},\hat{j},Z)$ for $i \geq 1$, we use the following recurrence:

$$w^{(i)}(\hat{h},\hat{j},Z) = \min\{\omega_{y_i}^{(2)}(\hat{h}',\hat{Z},\hat{j}',\mathbf{s}) + \hat{j}'d(\mathbf{s}) + w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})\}, \qquad (5.7)$$

where the minimum in the right part is taken over all integers $1 \leq \hat{h}' \leq \hat{h}$ and $0 < \hat{j}' \leq \hat{j}$, all sets $\hat{Z}$ such that $c \notin \hat{Z} \subset Z$, and all $\mathbf{s} \in \mathcal{M}$. We assume that $w^{(i)}(\hat{h},\hat{j},Z) = +\infty$ if the set in the right part is empty.

We prove the correctness of (5.7) by showing the opposite inequalities between the left and the right part.

If $w^{(i)}(\hat{h},\hat{j},Z)=+\infty$, then

$$w^{(i)}(\hat{h},\hat{j},Z)\geq\min\{\omega^{(2)}(\hat{h}',\hat{Z},\hat{j}',\mathbf{s})+\hat{j}'d(\mathbf{s})+w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})\}.$$

Suppose that $w^{(i)}(\hat{h},\hat{j},Z)<+\infty$. Consider $\hat{h}$-clustering $\mathcal{X}$ for $\mathbf{X}(Z)/(J\setminus J'')$ of cost $w^{(i)}(\hat{h},\hat{j},Z)$ satisfying (i)–(v). Let $X\in\mathcal{X}$ be the composite cluster such that $\alpha(X)=y_i$. Since $\alpha(J'')=x$, X contains points of J''. Let $\hat{J}''=I\cap J''$ and $\hat{j}'=\hat{J}''$. Denote by $\mathbf{s}\in\mathcal{M}$ the median of X_1. Consider $\hat{\mathcal{J}}''=\alpha^{-1}(V(T_{y_i}))\cap\mathcal{J}'$, that is, the set of initial clusters having nonempty intersections with the composite clusters that are mapped by α to the nodes of T_{y_i}. The coloring ψ colors these clusters by distinct colors and we define $\hat{Z}$ to be the set of colors of the clusters of $\hat{\mathcal{J}}''$; note that $c\notin\hat{Z}$. Denote by $\hat{h}'$ the number of clusters in $\mathcal{X}$ containing points of the initial clusters with colors in $\hat{Z}$ and let $\mathcal{X}_1$ be the set of these clusters; observe that $X\in\mathcal{X}_1$. Let $\mathcal{X}_2=\mathcal{X}\setminus\mathcal{X}_1$.

By the defintion of the values of $w^{(i-1)}(\cdot,\cdot,\cdot)$, we obtain that the cost of clustering for $\mathcal{X}_2$ is at least $w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})$. The cluster X contains $\hat{j}'$ points of J. Since $\mathbf{s}$ is its median, these $\hat{j}'$ points contribute $\hat{j}'d(\mathbf{s})$ to its cost. Then, by the definition of $\omega^{(2)}_{y_i}(\cdot,\cdot,\cdot,\cdot)$, we have that the cost of clustering for $\mathcal{X}_1$ is at least $\omega^{(2)}_{y_i}(\hat{h}',\hat{Z},\hat{j}',\mathbf{s})+\hat{j}'d(\mathbf{s})$. This means that $w^{(i)}(\hat{h},\hat{j},Z)\geq\omega^{(2)}(\hat{h}',\hat{Z},\hat{j}',\mathbf{s})+w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',(Z\setminus\hat{Z})\cup\{c\})$ and

$$w^{(i)}(\hat{h},\hat{j},Z)\geq\min\{\omega^{(2)}_{y_i}(\hat{h}',\hat{Z},\hat{j}',\mathbf{s})+\hat{j}'d(\mathbf{s})+w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})\}. \tag{5.8}$$

For the opposite direction, assume that integers $\hat{h}'$, $\hat{j}'$, a set $\hat{Z}$, and a median $\mathbf{s}$ are chosen in such a way that the value of $\omega^{(2)}_{y_i}(\hat{h}',\hat{Z},\hat{j}',\mathbf{s})+\hat{j}'d(\mathbf{s})+w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})$ is minimum. If the value is $+\infty$, then $w^{(i)}(\hat{h},\hat{j},Z)\leq\omega^{(2)}(\hat{h}',\hat{Z},\hat{j}',\mathbf{s})+\hat{j}'d(\mathbf{s})+w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})$ as required. Assume that $\omega^{(2)}_{y_i}(\hat{h}',\hat{Z},\hat{j}',\mathbf{s})<+\infty$ and $w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})<+\infty$.

By the definition of $\omega^{(2)}_{y_i}(\hat{h}',\hat{Z},\hat{j}',\mathbf{s})$, there is an $\hat{h}'$-clustering $\mathcal{X}_1$ for $\mathbf{X}(\hat{Z})$ of cost $\omega^{(2)}_{y_i}(\hat{h}',\hat{Z},\hat{j}',\mathbf{s})$ satisfying conditions (i)–(v) of the definition. In particular, $\mathcal{X}_1$ contains a special cluster X with the median $\mathbf{s}$ such that $p-\hat{j}'\leq|X|\leq q-\hat{j}'$ and X is mapped to the root y_i of T_{y_i} by the isomorphism α.

Let $J''\subseteq\hat{J}$ of size $\hat{j}$ and let $\hat{J}''\subseteq J''$ of size $\hat{j}'$. By the definition of $w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})$, there is an $(\hat{h}-\hat{h}')$-clustering $\mathcal{X}_2$ for $\mathbf{X}(Z\setminus\hat{Z})/((J\setminus J'')\cup\hat{J}'')$ of cost $w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})$ satisfying conditions (i)–(v) of the definition of $w^{(i-1)}(\cdot,\cdot,\cdot)$.

Observe that the clusters of $\mathcal{X}_1$ and $\mathcal{X}$ are pairwise disjoint and include all **points** of the initial clusters with their colors in Z except $\hat{j}'+j$ points of J. We construct the $\hat{h}$-clustering $\mathcal{X}$ for $\mathbf{X}(Z)/(J\setminus J'')$ as follows. First, we modify the cluster $X\in\mathcal{X}_1$ by

setting $X := X \cup \hat{J}''$. Note that we increase the cost of the cluster by at most $\hat{j}'d(\mathbf{s})$. Then we take the union of $\mathcal{X}_1$ and $\mathcal{X}_2$. The definitions of the values $\omega^{(2)}_{y_i}(\hat{h}', \hat{Z}, \hat{j}', \mathbf{s})$ and $w^{(i-1)}(\hat{h} - \hat{h}', \hat{j} - \hat{j}', Z \setminus \hat{Z})$ imply that $\mathcal{X}$ satisfies conditions (i)–(v) for $w^{(i)}(\hat{h}, \hat{j}, Z)$. Therefore, $w^{(i)}(\hat{h}, \hat{j}, Z) \leq \omega^{(2)}(\hat{h}', \hat{Z}, \hat{j}', \mathbf{s}) + \hat{j}'d(\mathbf{s}) + w^{(i-1)}(\hat{h} - \hat{h}', \hat{j} - \hat{j}', Z \setminus \hat{Z})$.

By the choice of $\hat{h}'$, $\hat{j}'$, $\hat{Z}$, and $\mathbf{s}$,

$$w^{(i)}(\hat{h}, \hat{j}, Z) \leq \min\{\omega^{(2)}_{y_i}(\hat{h}', \hat{Z}, \hat{j}', \mathbf{s}) + \hat{j}'d(\mathbf{s}) + w^{(i-1)}(\hat{h} - \hat{h}', \hat{j} - \hat{j}', Z \setminus \hat{Z})\}. \tag{5.9}$$

Then (5.8) and (5.9) imply (5.7).

We use (5.7) to compute the table of values of $w^{(f)}(\cdot, \cdot, \cdot)$. Then $\omega^{(1)}_x(h', Y, c, J, j) = w^{(f)}(h', |J| - j, Y)$ by the definition.

To evaluate the running time, notice that the initial table $w^{(f)}(\cdot, \cdot, \cdot)$ can be computed in $2^{\mathcal{O}(B)} \cdot n^2$, since $w(\hat{h})$ can be computed in $\mathcal{O}(n^2)$ time and then the table is constructed for at most n values of $\hat{j}$ and at most 2^ℓ sets Z. To compute the table $w^{(i)}(\cdot, \cdot, \cdot)$ from $w^{(i-1)}(\cdot, \cdot, \cdot)$ by (5.7) for $i \in \{1, \ldots, f\}$, we consider all pairs of integers $\hat{h}' \leq \hat{h}$, all pairs of sets Z and $\hat{Z} \subset Z$ and all $\mathbf{s} \in \mathcal{M}$. Since $\hat{h} \leq n$, $Z \subseteq \{1, \ldots, \ell\}$ and $\ell \leq 2B$, and $|\mathcal{M}| = 2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$, $w^{(i)}(\cdot, \cdot, \cdot)$ can be computed in $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$. Then the total running time is $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$. □

Further, we show how to compute $\omega^{(2)}_x(h', Y, j, \mathbf{s})$ if the tables of values of $\omega^{(1)}_y(\cdot, \cdot, \cdot, \cdot, \cdot)$ are already computed. Similarly to the proof of Claim 5.2.2, we also use an auxiliary dynamic programming algorithm.

Claim 5.2.3. *Let $x \in U$ be an internal vertex of T and assume that the table of values of $\omega^{(1)}_y(\cdot, \cdot, \cdot, \cdot, \cdot)$ is computed for every child y of x. Then $\omega^{(2)}_x(h', Y, j, \mathbf{s})$ can be computed in $2^{\mathcal{O}(B)} \cdot n^{\mathcal{O}(1)}$ time.*

Proof. Let $h' \leq h$, $Y \subseteq P$, $j \leq q$, and let $\mathbf{s} \in \mathcal{M}$. If $|Y| \neq \ell_x$, then $\omega^{(2)}_x(h', Y, j, \mathbf{s}) = +\infty$ by defintion. Assume that $|Y| = \ell_x$. In the same way as in the proof of Claim 5.2.2, denote by $y_1, \ldots, y_f$ the children of x in T. For $i \in \{1, \ldots, f\}$, let $T^{(i)}$ be the subtree of T induced by $\{x\} \cup \bigcup_{i'=1}^{i} V(T_{y_{i'}})$, set $U^{(i)} = U \cap V(T^{(i)})$ and $W^{(i)} = W \cap V(T^{(i)})$. Let also $t^{(i)} = |U^{(i)}|$ and $\ell^{(i)} = |W^{(i)}|$ for $i \in \{1, \ldots, f\}$. For an initial cluster J, we denote by $d(J) = \mathsf{dist}_0(\mathbf{s}, \mathbf{x}_i)$ for $\mathbf{x}_i \in J$. Similarly to the proof of Claim 5.2.2, we compute some auxiliary values.

For each $i \in \{1, \ldots, f\}$, every positive integer $\hat{h} \leq h'$, every nonnegative integer $\hat{j} \leq q$, and every nonempty $Z \subseteq P$, we define the auxiliary value as follows.

Definition 7 (Auxiliary partial solution)**.** *$w^{(i)}(\hat{h},\hat{j},Z)$ is the minimum cost of an $\hat{h}$-clustering $\mathcal{X}=\{X_1,\ldots,X_{\hat{h}}\}$ for $\mathbf{X}(Z)$ such that*

(i) the cost of X_1 is computed with respect to the median $\mathbf{s}$, that is, the cost equals $\sum_{\mathbf{x}_i\in X_1}\mathsf{dist}_0(\mathbf{s},\mathbf{x}_i)$,

(ii) $|X_1|=\hat{j}$ and $p\leq|X_{i'}|\leq q$ for $i'\in\{2,\ldots,\hat{h}\}$,

(iii) for the set of composite clusters $\mathcal{X}'\subseteq\mathcal{X}$, $\mathcal{X}''=\mathcal{X}'\cup\{X_1\}$ has size $t^{(i)}$, and the set $\mathcal{J}'\subseteq\mathcal{J}(Z)$ of initial clusters having nonempty intersections with the clusters from $\mathcal{X}''$ has size $\ell^{(i)}$,

(iv) $|Z|=\ell^{(i)}$ and the initial clusters in $\mathcal{J}'$ are colored by distinct colors by ψ,

(v) $G(\mathcal{X}'',\mathcal{J}')$ is isomorphic to $T^{(i)}$ with an isomorphism α that bijectively maps $\mathcal{X}''$ to $U^{(i)}$, $\mathcal{J}'$ to $W^{(i)}$, and $\alpha(X_1)=x$.

We assume that $w^{(i)}(\hat{h},\hat{j},Z)=+\infty$ if there is no such a $\hat{h}$-clustering.

Notice that the parameter $\hat{j}$ defines the size of a selected cluster X_1. Then, by the definition, we have that

$$\omega_x^{(2)}(h',Y,j,\mathbf{s})=\min\{w^{(f)}(h',\hat{j},Y)\mid p-j\leq\hat{j}\leq q-j\} \tag{5.10}$$

assuming that $\omega_x^{(2)}(h',Y,j,\mathbf{s})=+\infty$ if the set in the right part is empty.

We compute the tables of values of $w^{(i)}(\cdot,\cdot,\cdot)$ for $i=1,\ldots,f$.

First, we observe that

$$w^{(1)}(\hat{h},\hat{j},Z)=\min\{\omega_{y_1}^{(1)}(\hat{h}-1,Z,c,J,\hat{j})+\hat{j}d(J)\mid c\in Z,J\in\mathcal{J}(Z)\}; \tag{5.11}$$

as before, $w^{(1)}(\hat{h},\hat{j},Z)=+\infty$ if the set in the right part is empty.

To see that $w^{(1)}(\hat{h},\hat{j},Z)\geq\min\{\omega_{y_1}^{(1)}(\hat{h}-1,Z,c,J,\hat{j})+\hat{j}d(J)\mid c\in Z,J\in\mathcal{J}(Z)\}$, assume that $w^{(1)}(\hat{h},\hat{j},Z)<+\infty$; otherwise, the inequality is trivial. Let $\mathcal{X}=\{X_1,\ldots,X_{\hat{h}}\}$ be an $\hat{h}$-clustering for $\mathbf{X}(Z)$ satisfying conditions (i)–(v).

Since y_1 is the unique child of x in $T^{(1)}$, X_1 consists of $\hat{j}$ points of some initial cluster J. Let c be the color assigned to J by ψ. Then, by the definition of $\omega_{y_1}^{(1)}(\hat{h}-1,Z,c,J,\hat{j})$, $\{X_2,\ldots,X_{\hat{h}}\}$ is an $(\hat{h}-1)$-clustering for $\mathbf{X}(Z)/J'$ for $J'\subseteq J$ of size $\hat{j}$ that satisfies all the condition of the definition of $\omega_{y_1}^{(1)}(\cdot,\cdot,\cdot,\cdots,\cdot)$. Therefore, the cost of $\{X_2,\ldots,X_{\hat{h}}\}$ is an $(\hat{h}-1)$ is at least $\omega_{y_1}^{(1)}(\hat{h}-1,Z,c,J,\hat{j})$. The median of X_1 is $\mathbf{s}$ and X_1 contains $\hat{j}$ points of J.

Therefore, the cost of X_1 is $\hat{j}d(J)$. We conclude that $w^{(1)}(\hat{h},\hat{j},Z) \geq \omega^{(1)}_{y_1}(\hat{h}-1,Z,c,J,\hat{j})+\hat{j}d(J)$. Therefore, $w^{(1)}(\hat{h},\hat{j},Z) \geq \min\{\omega^{(1)}_{y_1}(\hat{h}-1,Z,c,J,\hat{j})+\hat{j}d(J) \mid c \in Z, J \in \mathcal{J}(Z)\}$.

Now we prove that $w^{(1)}(\hat{h},\hat{j},Z) \leq \min\{\omega^{(1)}_{y_1}(\hat{h}-1,Z,c,J,\hat{j})+\hat{j}d(J) \mid c \in Z, J \in \mathcal{J}(Z)\}$. If the right part of (5.11) is $+\infty$, then the inequality is trivial. Assume that this is not the case and let $c \in Z$ and $J \in \mathcal{J}(Z)$ be such that the right part of (5.11) achieves the minimum value for them. Then there is an $(\hat{h}-1)$-clustering $\mathcal{X}$ for $\mathbf{X}(Z)/J'$, where $J' \subseteq J$ has size $\hat{j}$, with the cost $\omega^{(1)}_{y_1}(\hat{h}-1,Z,c,J,\hat{j})$ that satisfies all the condition of the definition of $\omega^{(1)}_{y_1}(\cdot,\cdot,\cdot,\cdots,\cdot)$. Then we construct a new cluster $X = J'$ with the median $\mathbf{s}$. Clearly, the cost is $\hat{j}d(J)$. It is straightforward to verify that $\mathcal{X} \cup \{X\}$ satisfies (i)–(v). Therefore, $w^{(1)}(\hat{h},\hat{j},Z) \leq \omega^{(1)}_{y_1}(\hat{h}-1,Z,c,J,\hat{j})+\hat{j}d(J)$ and $w^{(1)}(\hat{h},\hat{j},Z) \leq \min\{\omega^{(1)}_{y_1}(\hat{h}-1,Z,c,J,\hat{j})+\hat{j}d(J) \mid c \in Z, J \in \mathcal{J}(Z)\}$.

Combining the two inequalities, we conclude that (5.11) holds.

To compute $w^{(1)}(\hat{h},\hat{j},Z)$ for $i \geq 2$, we show that

$$w^{(i)}(\hat{h},\hat{j},Z) = \min\{\omega^{(1)}_{y_i}(\hat{h}',\hat{Z},c,J,\hat{j}')+\hat{j}'d(J)+w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})\}, \quad (5.12)$$

where the minimum is taken over all positive integers $\hat{h}' < \hat{h}$, $\hat{j}' < \hat{j}$, all nonempty sets $\hat{Z} \subset Z$, all $c \in Z$, and $J \in \mathcal{J}(Z)$. As it is standard in our paper, $w^{(i)}(\hat{h},\hat{j},Z) = +\infty$ if the set in the right part of (5.12) is empty.

We prove (5.12) by demonstrating the opposite inequalities between the left and the right part.

If $w^{(i)}(\hat{h},\hat{j},Z) = +\infty$, then $w^{(i)}(\hat{h},\hat{j},Z) \geq \min\{\omega^{(1)}_{y_i}(\hat{h}',\hat{Z},c,J,\hat{j}')+\hat{j}'d(J)+w^{(i-1)}(\hat{h}-\hat{h}',\hat{j}-\hat{j}',Z\setminus\hat{Z})\}$. Assume that this is not the case. Then there is an $\hat{h}$-clustering $\mathcal{X}$ for $\mathbf{X}(Z)$ of cost $w^{(i)}(\hat{h},\hat{j},Z)$ satisfying (i)–(v). In particular, there is $X \in \mathcal{X}$ such that $|X| = \hat{j}$, $\alpha(X) = x$ and $\mathbf{s}$ is its median. Let $J \in \mathcal{J}(Z)$ be the initial cluster such that $\alpha(J) = y_i$. Denote by c its color. By definition, $J \cap X \neq \emptyset$. Let $J' = X \cap J$ and $\hat{j}' = |J'|$. Consider $\hat{\mathcal{J}}' = \alpha^{-1}(V(T_{y_i})) \cap \mathcal{J}'$, that is, the set of initial clusters intersecting composite clusters that are mapped by α to the vertices of T_{y_i}. Note that $J \in \hat{\mathcal{J}}'$. By definition, these clusters are colored by distinct colors by ψ. Denote by $\hat{Z}$ the set of their colors. Clearly, $c \in \hat{Z}$. Let $\mathcal{X}_1 \subseteq \mathcal{X} \setminus \{X\}$ be the set of clusters in $\mathcal{X}$ having nonempty intersections with with the initial clusters from $\hat{\mathcal{J}}'$; note that $X \notin \mathcal{X}_1$ by definition. Set $\hat{h}' = |\mathcal{X}_1|$. Let $\mathcal{X}_2 = \mathcal{X} \setminus \mathcal{X}_1$.

Observe that $\mathcal{X}_1$ is an $\hat{h}'$-clustering for $\mathbf{X}(\hat{Z})/J'$. Moreover, $\mathcal{X}_1$ satisfies all the conditions of the definition of $\omega^{(1)}_{y_i}(\hat{h}',\hat{Z},c,J,\hat{j}')$. This implies that the cost of $\mathcal{X}_1$ is at least $\omega^{(1)}_{y_i}(\hat{h}',\hat{Z},c,J,\hat{j}')$. Consider the clustering $\hat{\mathcal{X}}_2$ obtained from $\mathcal{X}_2$ by the replacement of X

by $\hat{X} = X \setminus J'$. Notice that the clusters of $\hat{\mathcal{X}}_2$ contains only points of initial clusters with colors from $Z \setminus \hat{Z}$. Also, we have hat $|\hat{X}| = \hat{j} - \hat{j}'$ and $|\hat{\mathcal{X}}_2| = \hat{h} - \hat{h}'$ because $i \geq 2$ and $X \neq J'$. Then it is straightforward to verify that $\hat{\mathcal{X}}_2$ is $(\hat{h} - \hat{h}')$-clustering for $\mathbf{X}(Z \setminus \hat{Z})$ satisfying (i)–(v) for $w^{(i-1)}(\cdot,\cdot,\cdot)$.

Therefore, the cost of $\hat{\mathcal{X}}_2$ is at least $w^{(i-1)}(\hat{h} - \hat{h}', \hat{j} - \hat{j}', Z \setminus \hat{Z})$. Finally, recall that $J' \subset X$. Since $\mathbf{s}$ is the median of X, the contribution of J' to the cost is $\hat{j}'d(J)$. We conclude that $w^{(i)}(\hat{h}, \hat{j}, Z) \geq \omega^{(1)}_{y_i}(\hat{h}', \hat{Z}, c, J, \hat{j}') + \hat{j}'d(J) + w^{(i-1)}(\hat{h} - \hat{h}', \hat{j} - \hat{j}', Z \setminus \hat{Z})$. Hence,

$$w^{(i)}(\hat{h}, \hat{j}, Z) \geq \min\{\omega^{(1)}_{y_i}(\hat{h}', \hat{Z}, c, J, \hat{j}') + \hat{j}'d(J) + w^{(i-1)}(\hat{h} - \hat{h}', \hat{j} - \hat{j}', Z \setminus \hat{Z})\}. \quad (5.13)$$

The opposite inequality is trivial if the right part of (5.12) equals $+\infty$. Assume that this is not the case and suppose that positive integers $\hat{h}' < \hat{h}$, $\hat{j}' < \hat{j}$, a set $\hat{Z} \subset Z$, $c \in Z$, and $J \in \mathcal{J}(Z)$ are chosen in such a way that the right part of (5.12) achieves the minimum value for them.

By the definition of $\omega^{(1)}_{y_i}(\hat{h}', \hat{Z}, c, J, \hat{j}')$, there is an $\hat{h}'$-clustering $\mathcal{X}_1$ for $\mathbf{X}(\hat{Z})/J'$ of cost $\omega^{(1)}_{y_i}(\hat{h}', \hat{Z}, c, J, \hat{j}')$ satisfying conditions (i)–(v) of the definition, where $J' \subseteq J$ of size $\hat{j}' = |J'|$. In particular, c is a color of J.

We also have that, by definition of $w^{(i-1)}(\hat{h} - \hat{h}', \hat{j} - \hat{j}', Z \setminus \hat{Z})$, there is an $(\hat{h} - \hat{h}')$-clustering for $\mathbf{X}(Z \setminus \hat{Z})$ satisfying conditions (i)-(v) of the definition. In particular, there is a special cluster $X \in \mathcal{X}_2$ of size $\hat{j} - \hat{j}'$ with the median $\mathbf{s}$.

We construct the clustering $\mathcal{X}$ for $\mathbf{X}(Z)$ as follows. First, we modify the cluster $X \in \mathcal{X}_2$ by replacing it by $X' = X \cup J'$. Then we take the union of $\mathcal{X}_1$ and the modified $\mathcal{X}_2$. It is straightforward to verify that $\mathcal{X}$ is a $\hat{h}$-clustering for $\mathbf{X}(Z)$ satisfying (i)–(v) for $w^{(i)}(\hat{h}, \hat{j}, Z)$. Since X' is obtained by adding $\hat{j}'$ points of J, the cost of $\mathcal{X}$ is $\omega^{(1)}_{y_i}(\hat{h}', \hat{Z}, c, J, \hat{j}') + \hat{j}'d(J) + w^{(i-1)}(\hat{h} - \hat{h}', \hat{j} - \hat{j}', Z \setminus \hat{Z})$. Therefore, $w^{(i)}(\hat{h}, \hat{j}, Z) \leq \omega^{(1)}_{y_i}(\hat{h}', \hat{Z}, c, J, \hat{j}') + \hat{j}'d(J) + w^{(i-1)}(\hat{h} - \hat{h}', \hat{j} - \hat{j}', Z \setminus \hat{Z})$ and, by the choice of $\hat{h}'$, $\hat{j}'$, $\hat{Z}$, c and J,

$$w^{(i)}(\hat{h}, \hat{j}, Z) \leq \min\{\omega^{(1)}_{y_i}(\hat{h}', \hat{Z}, c, J, \hat{j}') + \hat{j}'d(J) + w^{(i-1)}(\hat{h} - \hat{h}', \hat{j} - \hat{j}', Z \setminus \hat{Z})\}. \quad (5.14)$$

By (5.13) and (5.14), we conclude that the recurrence (5.12) holds. Then we compute the tables of values of $w^{(i)}(\cdot,\cdot,\cdot)$ for $i = 1, \ldots, f$ using (5.11) and (5.12). Finally, we apply (5.10) to compute $\omega^{(2)}_x(h', Y, j, \mathbf{s})$.

Clearly, the table of values of $w^{(1)}(\cdot,\cdot,\cdot)$ can be computed in $2^{\mathcal{O}(B)} \cdot n^3$ time because we

consider $\hat{h}, \hat{j} \leq n$ and at most 2^ℓ sets Z, and then go through at most ℓ values of c and at most n sets $\mathcal{J}$. To compute the tables of values of $w^{(i)}(\cdot,\cdot,\cdot)$ for $i \geq 2$, we consider all pairs of integers $\hat{h}' < \hat{h}$, all pairs $\hat{j}' < \hat{j}$, all nonempty sets $\hat{Z} \subset Z$, all $c \in Z$, and $J \in \mathcal{J}(Z)$. Since $\hat{h}', \hat{h}, \hat{j}', \hat{j} \leq n$, the number of pairs of set $\hat{Z} \subset Z$ is at most 3^ℓ, the number of the choices of c is at most ℓ and the number of the choices of J is at most n, we have that the total running time is $2^{\mathcal{O}(B)} \cdot n^{\mathcal{O}(1)}$ because $\ell \leq 2B$. □

Claims 5.2.1–5.2.3 allow us to compute the table of values of $\omega_z^{(2)}(\cdot,\cdot,\cdot,\cdot)$ for the root z of T bottom-up starting from the leaves (recall that $z \in U$). To make the final step of our algorithm, observe that the minimum cost of a feasible h-clustering for $\mathbf{X}(P)$ with respect to T, t' and ℓ' is

$$\min\{\omega_x^{(2)}(h, P, 0, \mathbf{s}) \mid \mathbf{s} \in \mathcal{M}\}$$

by the definition of these values.

The tree T has $\ell' + t' \leq 3B$ vertices. The table of values of either $\omega_x^{(1)}(\cdot,\cdot,\cdot,\cdot,\cdot)$ or $\omega_x^{(2)}(\cdot,\cdot,\cdot,\cdot)$ constructed for every node x has size $2^{\mathcal{O}(B\log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ and can be constructed in $2^{\mathcal{O}(B\log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ time by Claims 5.2.1–5.2.3. Therefore, the total running time is $2^{\mathcal{O}(B\log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$. □

Using Lemmas 7 and 8, we are able to check whether the considered instance has a colorful solution.

Putting all together

Now we are ready to put all ingredients of our algorithm together.

Lemma 9. Categorical Capacitated k-Median Clustering *can be solved in* $2^{\mathcal{O}(B\log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ *time by a Monte Carlo algorithm with false negatives.*

Proof. Let $(\mathbf{X}, \Sigma, k, B, p, q)$ be an instance of Categorical Capacitated k-Median Clustering with $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$. We start with computing the partition $\mathcal{J} = \{J_1, \ldots, J_s\}$ of $\mathbf{X}$ into initial clusters and this step can be done in polynomial time. By the next step, we construct the set $\mathcal{M} = \mathcal{M}(\mathbf{X}, B)$ of potential medians of size $2^{\mathcal{O}(B\log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ in $2^{\mathcal{O}(B\log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ time using Lemma 6.

Then we consider all nonnegative $t \leq \min\{B, k\}$ to guess the number of composite clusters. If $t = 0$, then the problem is solved in polynomial time. If $t \geq 1$, then we proceed and guess the number ℓ of initial clusters having nonempty intersections with composite clusters, where $t + 1 \leq \ell \leq B$. For the chosen values of t and ℓ, we consider

all forests F on $t+\ell$ vertices to guess the structure of $G(\mathcal{X}',\mathcal{J}')$. Recall that we have $2^{\mathcal{O}(B)}$ forests (see [72]) that can be listed in $2^{\mathcal{O}(B)}$ time (see [81]).

Further, we color the elements of $\mathcal{J}$ uniformly at random by ℓ colors and, given t, ℓ, F and a random coloring $\psi\colon \mathcal{J}\to\{1,\ldots,\ell\}$, check whether there is a feasible k-clustering of cost at most B. Recall that if there is a solution with t composite clusters such that exactly ℓ initial clusters have nonempty intersections with the composite clusters of the solution, then the probability that these ℓ clusters are assigned distinct colors in a random coloring ψ is at least e^{-2k}. Then the probability that some initial clusters having nonempty intersections with the composite clusters of the solution obtain the same color is at most $1-e^{-2B}$. This implies that if we try e^{2B} random colorings, then the probability that for every coloring, some initial clusters having nonempty intersections with the composite clusters of the solution are of the same color is at most $(1-e^{-2B})^{e^{2B}}\leq e^{-1}$. This implies that it is sufficient to consider $N=\lceil e^{2B}\rceil$ random colorings ψ. For each coloring, we verify the existence of a colorful solution. If a colorful solution exists for ψ, then we report that $(\mathbf{X},\Sigma,k,B,p,q)$ admits a required solution and stop. Otherwise, if we fail to find a colorful solution for every ψ, we report that there is in solution and the probability of an incorrect answer is at most $e^{-1}<1$.

For given given t, ℓ, F and a random coloring $\psi\colon \mathcal{J}\to\{1,\ldots,\ell\}$, we use Lemmas 7 and 8 for verifying whether a feasible k-clustering exists. For the connected components $F_1,\ldots,F_f$ of F, we apply Lemma 8 and compute the values $\omega_i(P,h)$ for all $i\in\{1,\ldots,f\}$, $P\subseteq\{1,\ldots\ell\}$ and positive integers $h\leq k$. By Lemma 8, this can be done in $2^{\mathcal{O}(B\log B)}|\Sigma|^B\cdot(mn)^{\mathcal{O}(1)}$ time. Given these values, we apply Lemma 7 to check in $2^{\mathcal{O}(B)}\cdot n^2$ time whether there is a feasible k-clustering for $\mathbf{A}$ of cost at most B with respect to F, t and ℓ.

The overall running time of this algorithm is $2^{\mathcal{O}(B\log B)}|\Sigma|^B\cdot(mn)^{\mathcal{O}(1)}$ and this concludes the proof. □

Derandomization

Our algorithm can be derandomized by standard tools [5] (see also [29, Chapter 5]). More precisely, we replace random colorings by functions from a perfect hash family.

Let s and ℓ be positive integers such that $s\geq\ell$. A set $\mathcal{F}$ of functions $\xi\colon\{1,\ldots,s\}\to\{1,\ldots,\ell\}$ is said to be an (s,ℓ)-*perfect hash family* if for every $X\subseteq\{1,\ldots,s\}$ of size ℓ, there is a $\xi\in\mathcal{F}$ such that $\xi|_X$ is a bijection between X and $\{1,\ldots,\ell\}$.

We use the result of Naor, Schulman, and Srinivasan [71] (see also [29, Chapter 5]).

Proposition 8. *For every $s \geq \ell \geq 1$, there is an (s,ℓ)-perfect hash family $\mathcal{F}$ of size $e^{\ell}\ell^{\mathcal{O}(\log \ell)} \cdot \log s$ that can be constructed in $e^{\ell}\ell^{\mathcal{O}(\log \ell)} \cdot s \log s$ time.*

We consider our set of initial clusters $\mathcal{J} = \{J_1, \ldots, J_s\}$ and construct an (s,ℓ)-perfect hash family $\mathcal{F}$. Since $\ell \leq 2B$ and $s \leq n$, $|\mathcal{F}| = e^{2B}(2B)^{\mathcal{O}(\log B)} \cdot \log n$ and $\mathcal{F}$ can be constructed in $e^{2B}(2B)^{\mathcal{O}(\log B)} \cdot n \log n$ time by Proposition 8. For every $\xi \in \mathcal{F}$, we define the coloring $\psi_\xi \colon \mathcal{J} \to \{1, \ldots, \ell\}$ by setting $\psi_\xi(J_i) = \xi(i)$ for $i \in \{1, \ldots, s\}$.

If $(\mathbf{X}, \Sigma, k, B, p, q)$ admits a solution with t composite clusters such that exactly ℓ initial clusters have nonempty intersections with the composite clusters, then there is $\xi \in \mathcal{F}$ such that ψ_ξ colors these initial clusters by distinct colors by the definition of an (s,ℓ)-perfect hash family. Then our randomized algorithm can be modified as follows: instead of trying $N = \lceil e^{2B} \rceil$ random colorings ψ we try all $\xi \in F$, we verify the existence of a colorful solution with respect to ψ_ξ. We obtain that we can solve Categorical Capacitated k-Median Clustering for $(\mathbf{X}, \Sigma, k, B, p, q)$ in $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$ deterministic time and this concludes the proof of Theorem 5.

5.3 Clustering with Size Constraints

In this section, we discuss other variants of Categorical k-Median Clustering with cluster size constraints: Balanced Categorical k-Median Clustering and Factor-Balanced Categorical k-Median Clustering. We also discuss the special case of Categorical Capacitated k-Median Clustering for $p = q = n/k$ that is equivalent to Balanced Categorical k-Median Clustering for $\delta = 0$ and to Factor-Balanced Categorical k-Median Clustering for $\alpha = 1$. We refer to this problem as Categorical Equal Clustering.

Recall that by Theorem 6, Categorical Capacitated k-Median Clustering is NP-complete for $k = 2$ and $p = q = n/2$, that is, Categorical Equal Clustering is NP-complete for $k = 2$. Using the same arguments as in the proof of Theorem 6, we can show the following more general claim.

Theorem 8. *For every fixed $\alpha \geq 1$ ($\delta \geq 0$, respectively),* Factor-Balanced Categorical k-Median Clustering *(*Balanced Categorical k-Median Clustering*, respectively) is* NP*-complete for $k = 2$ and binary matrices.*

From the positive side, we observe that Balanced Categorical k-Median Clustering and Factor-Balanced Categorical k-Median Clustering admit Turing

reductions to Categorical Capacitated k-Median Clustering, that is, Categorical Capacitated k-Median Clustering is the most general among the considered problems. For this, we make the following straightforward observation.

Observation 6. *An instance* $(\mathbf{X}, \Sigma, k, B, \delta)$ *of* Balanced Categorical k-Median Clustering *(an instance* $(\mathbf{X}, \Sigma, k, B, \alpha)$ *of* Factor-Balanced Categorical k-Median Clustering*, respectively) is a yes-instance if and only if there is a non-negative integer* p *such that* $\frac{n}{k} - \delta \leq p \leq \frac{n}{k}$ *(* $\frac{n}{\alpha k} \leq p \leq \frac{n}{k}$*, respectively) and for* $q = p + \delta$ *(*$q = \alpha p$*, respectively),* $(\mathbf{X}, \Sigma, k, B, p, q)$ *is a yes-instance of* Categorical Capacitated k-Median Clustering*.*

Thus, given an algorithm $\mathcal{A}$ for Categorical Capacitated k-Median Clustering, we can solve Balanced Categorical k-Median Clustering for $(\mathbf{X}, \Sigma, k, B, \delta)$ as follows. We consider all p starting from $\max\{1, \lceil \frac{n}{k} \rceil - \delta\}$ up to $\lfloor \frac{n}{k} \rfloor$, and use $\mathcal{A}$ to solve Categorical Capacitated k-Median Clustering for $(\mathbf{X}, \Sigma, k, B, p, \min\{n, p+\delta\})$. If $\mathcal{A}$ returns "yes" for one of the values of p, we conclude that $(\mathbf{X}, \Sigma, k, B, \delta)$ is a yes-instance of Balanced Categorical k-Median Clustering and stop. Otherwise, if $\mathcal{A}$ always returns "no", $(\mathbf{X}, \Sigma, k, B, \delta)$ is a no-instance. Clearly, Factor-Balanced Categorical k-Median Clustering can be solved in similar way. This allows to obtain the following corollary of Theorem 5.

Corollary 2. Balanced Categorical k-Median Clustering *and* Factor-Balanced Categorical k-Median Clustering *are solvable in time* $2^{\mathcal{O}(B \log B)}|\Sigma|^B \cdot (mn)^{\mathcal{O}(1)}$.

5.4 Kernelization for Clustering with Size Constraints

In this section, we discuss kernelization for clustering problems with size constraints. In [37, Theorem 3], Fomin, Golovach and Panolan proved that Categorical k-Median Clustering does not admit a polynomial kernel when parmeterized by B, unless $\mathsf{NP} \subseteq \mathsf{coNP/poly}$. This immediately implies the following proposition.

Proposition 5. Categorical Capacitated k-Median Clustering *(*Balanced Categorical k-Median Clustering *and* Factor-Balanced Categorical k-Median Clustering*, respectively) has no polynomial kernel when paramterized by* B*, unless* $\mathsf{NP} \subseteq \mathsf{coNP/poly}$*, even if* $\Sigma = \{0, 1\}$*.*

Also, by Theorems 6 and 8 the problems are already NP-hard for $k = 2$. Thus, for kernelization, we have to consider more restrictive parameterizations. Up to now, we

have only partial results. In particular, we can show Balanced Categorical k-Median Clustering admits a polynomial kernel when parameterized by B, k and δ.

We start with some auxiliary results. First, we observe that if there is an initial cluster J of size at least $B+1$, then at least one median should be the same as a point of the input set with point in J.

Observation 7. *Let $\{X_1,\ldots,X_k\}$ be a k-clustering of a $\mathbf{X}=\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ of cost at most B and let $J\subseteq\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ be an initial cluster with $|J|\geq B+1$. Then there is an $i\in\{1,\ldots,k\}$ such that an optimal median of X_i coincides with $\mathbf{s}=\mathbf{x}_j$ for $\mathbf{x}_j\in J$.*

Proof. For the sake of contradiction, assume that medians $\mathbf{c}_1,\ldots,\mathbf{c}_k$ for the clusters $\{X_1,\ldots,X_k\}$, respectively, are distinct from $\mathbf{s}$. Then

$$\sum_{i=1}^{k}\sum_{\mathbf{x}_j\in X_i}\mathsf{dist}_0(\mathbf{c}_i,\mathbf{x}_j)\geq\sum_{i=1}^{k}\sum_{\mathbf{x}_j\in J\cap X_i}\mathsf{dist}_0(\mathbf{c}_i,\mathbf{s})\geq|J|>B$$

contradicting that the cost of $\{X_1,\ldots,X_k\}$ is at most B. □

Our next lemma shows that if there is a clustering such that a median $\mathbf{c}_i$ coincides with point $\mathbf{x}_j$, then we can either collect all the elements of the initial cluster J containing $\mathbf{x}_j$ in the same cluster of a solution or form a cluster of a solution out of its elements.

Lemma 10. *Let $\{X_1,\ldots,X_k\}$ be a k-clustering of $\mathbf{X}=\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ with optimal medians $\mathbf{c}_1,\ldots,\mathbf{c}_k$, respectively. Let also $\mathbf{S}\subseteq\{\mathbf{c}_1,\ldots,\mathbf{c}_k\}$ be the set of medians coinciding with points of $\mathbf{X}$. Then there is a k-clustering $\{X'_1,\ldots,X'_k\}$ for $\mathbf{X}$ such that*

(i) $|X'_i|=|X_i|$ *for all* $i\in\{1,\ldots,k\}$,

(ii) $\sum_{i=1}^{k}\sum_{\mathbf{x}_j\in X'_i}\mathsf{dist}_0(\mathbf{c}_i,\mathbf{x}_j)\leq\sum_{i=1}^{k}\sum_{\mathbf{x}_j\in X_i}\mathsf{dist}_0(\mathbf{c}_i,\mathbf{x}_j)$, *and*

(iii) for every $\mathbf{s}\in\mathbf{S}$ and the initial cluster J such that $\mathbf{s}=\mathbf{x}_j$ for $j\in J$, there is $i\in\{1,\ldots,k\}$ such that either $J\subseteq X'_i$ or $X'_i\subset J$.

Proof. Let $\mathbf{c}_1,\ldots,\mathbf{c}_k$ be optimal medians for $X_1,\ldots,X_k$, respectively. Assume without loss of generality that $\mathbf{S}=\{\mathbf{c}_1,\ldots,\mathbf{c}_t\}$, and denote by $J_1,\ldots,J_t$ the initial clusters such that for every $i\in\{1,\ldots,t\}$, $\mathbf{x}_j=\mathbf{c}_i$ for $\mathbf{x}_j\in J_i$. Let $\mathcal{X}'=\{X'_1,\ldots,X'_k\}$ be a k-clustering for $\mathbf{X}$ such that (a) $|X'_i|=|X_i|$ for all $i\in\{1,\ldots,k\}$, (b) $\sum_{i=1}^{k}\sum_{\mathbf{x}_j\in X'_i}\mathsf{dist}_0(\mathbf{c}_i,\mathbf{x}_j)\leq\sum_{i=1}^{k}\sum_{\mathbf{x}_j\in X_i}\mathsf{dist}_0(\mathbf{c}_i,\mathbf{x}_j)$, and (c) $\sum_{i=1}^{t}|X'_i\cap J_i|$ is maximum. We claim that $\mathcal{X}'$ satisfies conditions (i)–(iii) of the lemma. Clearly, (i) and (ii) are fulfilled by conditions (a) and

(b) of the choice of $\mathcal{X}'$. To show (iii), we prove that either $J_i \subseteq X'_i$ or $X'_i \subset J_i$ for every $i \in \{1,\ldots,t\}$.

Assume to the contrary that there is $i \in \{1,\ldots,t\}$ such that neither $J_i \subseteq X'_i$ nor $X'_i \subset J_i$. Then there is a cluster X'_j for $j \in \{1,\ldots,k\}$ such that $j \neq i$, $X'_j \cap J \neq \emptyset$, and there is $\mathbf{x}_h \in X'_i$ such that $\mathbf{x}_h \notin J_i$. Let $\mathbf{x}_\ell \in X'_j \cap J_i$. Consider the k-clustering $\mathcal{X}'' = \{X''_1,\ldots,X''_k\}$ such that $X''_i = (X'_i \cup \{\mathbf{x}_\ell\}) \setminus \{\mathbf{x}_h\}$, $X''_j = (X'_j \cup \{\mathbf{x}_h\}) \setminus \{\mathbf{x}_\ell\}$, and $X''_h = X'_h$ for $h \in \{1,\ldots,k\}$ such that $h \neq i,j$. In words, we exchange the elements $\mathbf{x}_h$ and $\mathbf{x}_\ell$ between X'_i and X'_j. Then

$$\sum_{p=1}^{k}\sum_{\mathbf{x}_q\in X'_p}\mathsf{dist}_0(\mathbf{c}_p,\mathbf{x}_q)-\sum_{p=1}^{k}\sum_{\mathbf{x}_q\in X''_p}\mathsf{dist}_0(\mathbf{c}_p,\mathbf{x}_q)$$
$$=\mathsf{dist}_0(\mathbf{c}_i,\mathbf{x}_h)+\mathsf{dist}_0(\mathbf{c}_j,\mathbf{x}_\ell)-\mathsf{dist}_0(\mathbf{c}_i,\mathbf{x}_\ell)-\mathsf{dist}_0(\mathbf{c}_j,\mathbf{x}_h),$$

and since $\mathbf{c}_i = \mathbf{x}_\ell$, we obtain that

$$\sum_{p=1}^{k}\sum_{\mathbf{x}_q\in X'_p}\mathsf{dist}_0(\mathbf{c}_p,\mathbf{x}_q)-\sum_{p=1}^{k}\sum_{\mathbf{x}_q\in X''_p}\mathsf{dist}_0(\mathbf{c}_p,\mathbf{x}_q)$$
$$=\mathsf{dist}_0(\mathbf{x}_\ell,\mathbf{x}_h)+\mathsf{dist}_0(\mathbf{c}_j,\mathbf{x}_\ell)-\mathsf{dist}_0(\mathbf{c}_j,\mathbf{x}_h)\geq 0$$

by the triangle inequality. This means that

$$\sum_{p=1}^{k}\sum_{\mathbf{x}_q\in X''_p}\mathsf{dist}_0(\mathbf{c}_p,\mathbf{x}_q)\leq\sum_{p=1}^{k}\sum_{\mathbf{x}_q\in X'_p}\mathsf{dist}_0(\mathbf{c}_p,\mathbf{x}_q)\leq\sum_{p=1}^{k}\sum_{\mathbf{x}_q\in X_p}\mathsf{dist}_0(\mathbf{c}_p,\mathbf{x}_q). \tag{5.15}$$

Since $|X''_i| = |X'_i|$ for all $i \in \{1,\ldots,t\}$, $\mathcal{X}''$ satisfies condition (a) of the choice of $\mathcal{X}'$. Condition (b) is satisfied because of (5.15). However, $|X''_i \cap J| = |(X'_i \cap X) \cup \{\mathbf{x}_\ell\}| = |X'_i \cap J| + 1$. Because $\mathcal{X}''$ was obtained by the exchange $\mathbf{x}_h$ and $\mathbf{x}_\ell$ between X'_i and X'_j, $X'_p \cap J_p \subseteq X''_p \cap J_p$ for $p \in \{1,\ldots,t\}$. We obtain that $\sum_{p=1}^{t}|X'_p \cap J_p| < \sum_{p=1}^{t}|X''_p \cap J_p|$ contradicting (c). Therefore, either $J_p \subseteq X'_p$ or $X'_p \subset J_p$ for every $p \in \{1,\ldots,t\}$ as it claimed. □

The following lemma is used to find medians if the sizes of clusters in a solution are sufficiently big.

Lemma 11. *Let $\mathcal{X} = \{X_1,\ldots,X_k\}$ be a k-clustering of $\mathbf{X} = \{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ of cost at most B such that $s \leq |X_i| \leq s+\delta$ for all $i \in \{1,\ldots,k\}$, where δ is a nonnegative integer and an integer s satisfying $s \geq 2B+1+(k-1)\delta$. Then for every initial clusters $J \subseteq \{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$, the following is fulfilled for $\mathbf{c} = \mathbf{x}_j$ for $\mathbf{x}_j \in J$:*

(i) if $|J| \mod s \geq B + 1 + (k-1)\delta$*, then exactly* $\left\lceil \frac{|J|}{s} \right\rceil$ *clusters of* $\mathcal{X}$ *have optimal medians coinciding with* $\mathbf{c}$ *(the other medians are different),*

(ii) if $|J| \mod s \leq B + (k-1)\delta$*, then exactly* $\left\lfloor \frac{|J|}{s} \right\rfloor$ *clusters of* $\mathcal{X}$ *have optimal medians coinciding with* $\mathbf{c}$*.*

Proof. We start with proving (i). Let $|J| \mod s \geq B + 1 + (k-1)\delta$. We show that (i) holds for J by induction on $p = \left\lfloor \frac{|J|}{s} \right\rfloor$.

The base case is $p = 0$. Then $\left\lceil \frac{|J|}{s} \right\rceil = 1$. As $|J| \mod s \geq B + 1 + (k-1)\delta$ and $\left\lfloor \frac{|J|}{s} \right\rfloor = 0$, $B + 1 \leq |J| \leq s$. By Observation 7, there is a cluster in $\mathcal{X}$ whose optimal median is $\mathbf{c}$. Thus, at least one optimal median coincides with $\mathbf{c}$. Without loss of generality, we assume that $\mathbf{c}$ is the median of X_1. We now show that $\mathbf{c}_i \neq \mathbf{c}$ for $i \in \{2, \ldots, k\}$. Assume to the contrary that there exists $\mathbf{c}_h \in \{X_2, \ldots, X_k\}$ such that $\mathbf{c}_h = \mathbf{c}$. By Lemma 10, there is a k-clustering $\mathcal{X} = \{X'_1, \ldots, X'_k\}$ for $\mathbf{X}$ such that $|X'_i| = |X'_i|$ for all $i \in \{1, \ldots, k\}$, $\sum_{i=1}^{k} \sum_{\mathbf{x}_j \in X'_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j) \leq \sum_{i=1}^{r} \sum_{\mathbf{x}_j \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j)$, and $J \subseteq X'_1$. Then

$$\sum_{i=1}^{k} \sum_{\mathbf{x}_j \in X'_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j) \geq \sum_{\mathbf{x}_j \in X'_h} \mathsf{dist}_0(\mathbf{c}_h, \mathbf{x}_j) = \sum_{\mathbf{x}_j \in X'_h} \mathsf{dist}_0(\mathbf{c}, \mathbf{x}_j) \geq |X'_h| \geq s \geq B + 1,$$

contradicting that $\mathsf{cost}(\mathcal{X}) \leq B$. We conclude that exactly one median coincides with $\mathbf{c}$, that is, (i) holds for $p = 0$.

Now let $p \geq 1$ and assume that the claim holds when p is smaller. Note that $k \geq 2$ in this case. We observe that, because $|J| \mod s \geq B + 1 + (k-1)\delta$, $|J| \geq sp + B + 1 + (k-1)\delta$. By Observation 7, there is a cluster in $\mathcal{X}$ whose optimal median is $\mathbf{c}$. Without loss of generality, we assume that $\mathbf{c}$ is the median of X_1. Then by Lemma 10, there is a k-clustering $\{X'_1, \ldots, X'_k\}$ for $\mathbf{X}$ such that $|X'_i| = |X_i|$ for all $i \in \{1, \ldots, k\}$, $\sum_{i=1}^{k} \sum_{\mathbf{x}_j \in X'_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j) \leq \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j)$, and $X'_1 \subset J$.

Consider $\mathbf{X}' = \mathbf{X} \setminus X_1$, that is, $\mathbf{X}'$ is obtained from $\mathbf{X}$ by the deletion of the points in X_1. Notice that $\mathcal{X}' = \{X'_2, \ldots, X'_k\}$ is an $(k-1)$-clustering for $\mathbf{X}'$ of cost at most B. Moreover, because $|X'_i| = |X_i| \geq s \geq 2B + 1$, $\mathbf{c}_2, \ldots, \mathbf{c}_k$ are unique optimal medians for $X'_2, \ldots, X'_k$, respectively, by Observation 3. Let $J' = J \setminus X'_1$. Since $|X'_1| \leq s + \delta$,

$$|J'| = |J| - |X'_1| \geq sp + B + 1 + (k-1)\delta - s - \delta = s(p-1) + B + 1 + (k-2)\delta \geq B + 1 + (k-2)\delta.$$

By our inductive hypothesis, exactly $\left\lceil \frac{|J'|}{s} \right\rceil$ clusters of $\mathcal{X}'$ have optimal medians coinciding with $\mathbf{c}$. As $|X_1| \geq s$, $\left\lfloor \frac{|J'|}{s} \right\rfloor \leq p-1$. Because $|J'| \geq s(p-1) + B + 1 + (k-2)\delta$, $\left\lfloor \frac{|J'|}{s} \right\rfloor \geq p-1$. Hence, $\left\lfloor \frac{|J'|}{s} \right\rfloor = p - 1$ and $\left\lceil \frac{|J'|}{s} \right\rceil = p$. Since $\mathbf{c}_2, \ldots, \mathbf{c}_k$ are optimal medians, exactly p

of them are equal to $\mathbf{c}$. Together with the median $\mathbf{c}_1 = \mathbf{c}$, exactly $p+1$ medians in $\{\mathbf{c}_1, \ldots, \mathbf{c}_k\}$ are equal to $\mathbf{c}$. Then exactly $\left\lceil \frac{|J|}{s} \right\rceil = p+1$ clusters of $\mathcal{X}$ have optimal medians coinciding with $\mathbf{c}$. This completes the proof of (i).

To show (ii), we first claim that for every initial cluster J, there are at least $p = \left\lfloor \frac{|J|}{s} \right\rfloor$ clusters in $\mathcal{X}$, whose optimal medians are equal to $\mathbf{c}$, where $\mathbf{c} = \mathbf{x}_j$ for $\mathbf{x}_j \in J$. The proof is by induction on p.

The claim is trivial if $p = 0$. Let $p \geq 1$ and assume that the claim holds when p is smaller. Since $p \geq 1$, $|J| \geq s \geq B+1$. By Observation 7, there is a cluster in $\mathcal{X}$ whose optimal median is $\mathbf{c}$. Without loss of generality, we assume that $\mathbf{c}$ is the median of X_1. Then by Lemma 10, there is an k-clustering $\{X'_1, \ldots, X'_k\}$ for $\mathbf{X}$ such that $|X'_i| = |X_i|$ for all $i \in \{1, \ldots, k\}$, $\sum_{i=1}^{k} \sum_{\mathbf{x}_j \in I'_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j) \leq \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j)$, and either $J \subseteq X'_1$ or $X'_1 \subset J$.

Suppose that $J \subseteq X'_1$. Then $|J| \leq |X'_i| \leq s + \delta < 2s$. This means that $p = 1$ and our claim holds, as $\mathbf{c} = \mathbf{c}_1$.

Assume from now that this is not the case, that is, $X'_i \subset J$. Then we argue similarly to the proof of (i). Consider $\mathbf{X}' = \mathbf{X} \setminus X_1$, that is, $\mathbf{X}'$ is obtained from $\mathbf{X}$ by the deletion of the points in X_1. Notice that $\mathcal{X}' = \{X'_2, \ldots, X'_k\}$ is an $(k-1)$-clustering for $\mathbf{X}'$ of cost at most B. Moreover, because $|X'_i| = |X_i| \geq s \geq 2B+1$, $\mathbf{c}_2, \ldots, \mathbf{c}_k$ are unique optimal medians for $X'_2, \ldots, X'_k$, respectively, by Observation 3. Let $J' = J \setminus X'_1$.

If $\left\lfloor \frac{|J'|}{s} \right\rfloor \geq p-1$, then by the inductive assumption, there are at least $p-1$ clusters in $\mathcal{X}'$, whose optimal medians coincide with $\mathbf{c}$. Thus, at least $p-1$ medians from $\{\mathbf{c}_2, \ldots, \mathbf{c}_k\}$ are equal to $\mathbf{c}$. Taking into account $\mathbf{c}_1 = \mathbf{c}$, we have that at least p medians from $\{\mathbf{c}_1, \ldots, \mathbf{c}_k\}$ are equal to $\mathbf{c}$, as required.

Let $\left\lfloor \frac{|J'|}{s} \right\rfloor \leq p-2$. Note that $p \geq 2$ in this case. Since $|X'_1| \leq s + \delta$ and $|J| \geq ps$, we obtain that $|J'| = |J| - |X'_1| \geq (p-2)s + (s - \delta)$. Thus, $\left\lfloor \frac{|J'|}{s} \right\rfloor = p-2$ and

$$|J'| \mod s \geq s - \delta \geq 2B + 1 + (k-1)\delta \geq B + 1 + (k-2)\delta.$$

By the already proven (i), we have that there are at least $\left\lceil \frac{|J'|}{s} \right\rceil = p-1$ clusters in $\mathcal{X}'$, whose optimal medians coincide with $\mathbf{c}$. Since $\mathbf{c}_1 = \mathbf{c}$, we again obtain that at least p medians from $\{\mathbf{c}_1, \ldots, \mathbf{c}_k\}$ are equal to $\mathbf{c}$. This concludes the proof of our auxiliary claim.

To finish the proof of (ii), assume that $|J| \mod s \leq B + (k-1)\delta$. We already have that at least $p = \left\lfloor \frac{|J|}{s} \right\rfloor$ clusters of $\mathcal{X}$ have optimal medians coinciding with $\mathbf{c}$. It remains to

show that there are at most p such clusters. Assume to the contrary that at least $p+1$ medians are equal to $\mathbf{s}$ and assume without loss of generality that $\mathbf{c} = \mathbf{c}_1 = \ldots = \mathbf{c}_{p+1}$. Then

$$\sum_{i=1}^{k}\sum_{\mathbf{x}_j\in X_i}\mathsf{dist}_0(\mathbf{c}_i,\mathbf{x}_j) \geq \sum_{i=1}^{p+1}\sum_{\mathbf{x}_j\in X_i}\mathsf{dist}_0(\mathbf{c}_i,\mathbf{x}_j) = \sum_{\mathbf{x}_j\in X_1\cup\ldots\cup X_{p+1}}\mathsf{dist}_0(\mathbf{c},\mathbf{x}_j)$$
$$\geq \sum_{\mathbf{x}_j\in(X_1\cup\ldots\cup X_{p+1})\setminus J}\mathsf{dist}_0(\mathbf{c},\mathbf{x}_j) \geq |(X_1\cup\ldots\cup X_{p-1})\setminus X|.$$

We know that $|X_i| \geq s$ for $i \in \{1,\ldots,k\}$. Then $|X_1\cup\ldots\cup X_{p+1}| \geq s(p+1)$. Since $|J| \bmod s \leq B+(k-1)\delta$, $|J| \leq ps+B+(k-1)\delta$. This implies $|(X_1\cup\ldots\cup X_{p+1})\setminus J| \geq s-B-(k-1)\delta \geqslant B+1$. Hence $\sum_{\mathbf{x}_j\in(X_1\cup\ldots\cup X_{P+1})\setminus J}\mathsf{dist}_0(\mathbf{c},\mathbf{x}_j) \geq B+1 > B$ contradicting that $\mathsf{cost}(\mathcal{X}) \leq B$. This proves that exactly $\left\lfloor\frac{|J|}{s}\right\rfloor$ clusters of $\mathcal{X}$ have optimal medians coinciding with $\mathbf{c}$. □

Lemma 11 allows us to compute optimal medians and solve Balanced Categorical k-Median Clustering if the average size of clusters is sufficiently big.

Lemma 12. Balanced Categorical k-Median Clustering *can be solved in polynomial time for instances* $(\mathbf{X},\Sigma,k,B,\delta)$ *with* $\frac{n}{k} \geq 2B+1+\delta k$.

Proof. Let $(\mathbf{X},\Sigma,k,B,\delta)$ be an instance of Balanced Categorical k-Median Clustering with $\frac{n}{k} \geq 2B+1+\delta k$. Clearly, we can assume that $\delta \leq n-1$. If $(\mathbf{X},\Sigma,k,B,\delta)$ is a yes-instance, then there is an integer s such that $\frac{n}{k}-\delta \leq s \leq \frac{n}{k}$ and $s \leqslant |X_i| \leqslant s+\delta$ for a solution $\{X_1,\ldots,X_k\}$ to the instance.

Then we consider all integers s such that $\frac{n}{k}-\delta \leq s \leq \frac{n}{k}$. For each value of s, we check whether there is a solution $\{X_1,\ldots,X_k\}$ for the considered instance with $s \leq |X_i| \leq s+\delta$, for all $i \in \{1,\ldots,k\}$. If yes, we return the yes-answer, otherwise, if we fail to find a solution for every s, then the algorithm returns the no-answer.

Let s be fixed. For each initial cluster J, we compute $\left\lfloor\frac{|J|}{s}\right\rfloor$ and $|J| \bmod s$. Using these two values, we find the medians coinciding with $\mathbf{c}$ such that $\mathbf{c} = \mathbf{x}_j$ for $\mathbf{x}_j \in J$ using Lemma 11. Denote by $\mathcal{C}$ the obtained collection of medians. If $|\mathcal{C}| \neq k$, then we discard the current choice of s. Otherwise, $\mathcal{C}$ contains exactly k potential medians and we combine Observation 6 and Lemma 1 to decide whether $(\mathbf{X},\Sigma,k,B,\delta)$ admits a solution with these medians.

Since we consider at most $\delta+1 \leq n$ values of s and the algorithm from Lemma 1 is polynomial, the total running time of our algorithm is polynomial. □

In [37], Fomin et al. proved that Categorical k-Median Clustering admits a polynomial kernel when parameterized by B and k for binary alphabet. As one of the steps of their kernelization algorithm (see Theorem 2 of [37]), they show that the dimension of the output set can be reduced to $\mathcal{O}(B(B+k))$. Formally, the proof is done for the binary case, that is, for $\Sigma = \{0,1\}$.

However, the reduction rule used in [37] works for arbitrary alphabet Σ because to apply the rule, we only should be able to compute the Hamming distances between pairs of points of $\mathbf{X}$ and check whether two given coordinates of certain subcollection of points are the same.

We state this result in the following lemma.

Lemma 13 ([37]). *There is a polynomial algorithm that, given an instance $(\mathbf{X}, \Sigma, k, B)$ of* Categorical k-Median Clustering *with $\mathbf{X}$, a set of n points from Σ^m, produces an equivalent instance $(\mathbf{X}', \Sigma, k, B)$ with X' of n points from $\Sigma^{m'}$ such that the following holds:*

- $m' = \mathcal{O}(B(B+k))$.
- *$\{X_1, \ldots, X_k\}$ is a solution for $(\mathbf{X}, \Sigma, k, B)$ if and only if it is also a solution for $(\mathbf{X}', \Sigma, k, B)$.*

Now we are ready to show a polynomial kernel for Balanced Categorical k-Median Clustering.

Theorem 7. Balanced Categorical k-Median Clustering *admits a kernel, where the output set has $O(k(B+\delta k))$ points from a space of dimension $\mathcal{O}(B(B+k))$ over an alphabet of size at most $B+k$.*

Proof. Let $(\mathbf{X}, \Sigma, k, B, \delta)$ be an instance of Balanced Categorical k-Median Clustering with $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$.

Suppose $\frac{n}{k} \geq 2B+1+\delta k$. Then, by Lemma 12, the problem can be solved in polynomial time. We do it and return a trivial yes or no-instance, respectively. For example, we can return either the set with $\{0,0\}$ or $\{0,1\}$, respectively, and set $k=1$, $B=0$ and $\delta = 0$. Assume from now that $\frac{n}{k} \leq 2B+\delta k$, that is, $n \leq 2Bk + \delta k^2$. If $\mathbf{X}$ has at least $B+k+1$ pairwise distinct points, then for every k-clustering $\{X_1, \ldots, X_k\}$ and every $\mathbf{c}_1, \ldots, \mathbf{c}_k \in \Sigma^m$, $\sum_{i=1}^{k} \sum_{\mathbf{x}_j \in X_i} \mathsf{dist}_0(\mathbf{c}_i, \mathbf{x}_j) \geq B+1$ because at least $B+1$ points of $\mathbf{X}$ are distinct from each median. Thus, $(\mathbf{X}, \Sigma, k, B, \delta)$ is a no-instance in this case, and we return a trivial no-instance of Balanced Categorical k-Median Clustering.

Assume from now that the number of pairwise distinct points is at most $B + k$. If $|\Sigma| > B+k$, then we can replace every symbol of Σ by a symbol of $\Sigma' = \{0, \ldots, B+k-1\}$ maintaining the following property: for each point of $\mathbf{X}$, the same symbols of Σ are replaced by the same symbols of Σ'. It is straightforward to verify that this replacement produces an equivalent instance because we are using the Hamming distances. From now, we assume that $|\Sigma| \leq B + k$.

Given $(\mathbf{X}, \Sigma, k, B, \delta)$, we consider the instance $(\mathbf{X}, \Sigma, k, B)$ of Categorical k-Median Clustering. We use the algorithm from Lemma 13 and denote by $(\mathbf{X}', \Sigma, k, B)$ the output instance. Then we construct the instance $(\mathbf{X}', \Sigma, k, B, \delta)$ of Balanced Categorical k-Median Clustering and output it.

We show that $(\mathbf{X}, \Sigma, k, B, \delta)$ is a yes-instance of Balanced Categorical k-Median Clustering if and only if $(\mathbf{X}', \Sigma, k, B, \delta)$ is a yes-instance.

For the forward direction, suppose $(\mathbf{X}, \Sigma, k, B, \delta)$ is a yes-instance of Balanced Categorical k-Median Clustering. Let $\mathcal{X} = \{X_1, \ldots, X_k\}$ be a solution to the instance. Clearly, $\mathcal{X}$ is a solution for the instance $(\mathbf{X}, \Sigma, k, B)$ of Categorical k-Median Clustering. By Lemma 13, $\mathcal{X}$ is a solution for $(\mathbf{X}', \Sigma, k, B)$. Then $\mathcal{X}$ is a solution for $(\mathbf{X}', \Sigma, k, B, \delta)$. For the opposite direction, the arguments are similar. Let $\mathcal{X} = \{X_1, \ldots, X_k\}$ be a solution for $(\mathbf{X}', \Sigma, k, B, \delta)$. Then this is a solution for the instance $(\mathbf{X}', \Sigma, k, B)$ of Categorical k-Median Clustering and, by Lemma 13, a solution for $(\mathbf{X}', \Sigma, k, B)$. Finally, $\mathcal{X}$ is a solution of $(\mathbf{X}, \Sigma, k, B, \delta)$.

Recall that $n = \mathcal{O}(k(B+\delta k))$ and note that $\mathbf{X}'$ has dimension $\mathcal{O}(B(B+k))$ by Lemma 13. Since $|\Sigma| \leq B + k$, we conclude that the output set has $\mathcal{O}(k(B + \delta k))$ points from the space of dimension $O(B(B + k))$ over an alphabet of size at most $B + k$.

It is easy to see that our kernelization algorithm is polynomial and this concludes the proof.

□

Chapter 6

FPT Approximation Schemes/ Lossy Kernelization for Clustering

In this chapter, we study the ℓ_p-Equal k-Median Clustering from the perspective of parameterized preprocessing with the cost of clustering as a parameter. The results mentioned in this chapter have appeared in Article 2. Recall that in this problem, we consider the metric space $\mathcal{M} = \mathbb{Z}^d$ and dist is the ℓ_p-norm. The optimization version of the problem is defined as follows.

ℓ_p-Equal k-Median Clustering

Input: A multiset $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ of n points in $\mathbb{Z}^d$, a positive integer k and n is divisible by k

Task: Find a partition $\mathcal{X} = \{X_1, \ldots, X_k\}$ of $\mathbf{X} \subseteq \mathbb{Z}^d$ of points and k centers $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\}$ in $\mathbb{R}^d$ such that size of each cluster is the same, that is, $|X_1| = \ldots = |X_k| = \frac{n}{k}$ minimizing the following objective function over all the pairs $(\mathcal{X}, \mathcal{C})$

$$cost(\mathcal{X}, \mathcal{C}) = \sum_{i=1}^{k} \sum_{\mathbf{x} \in X_i} \mathsf{dist}_p(\mathbf{x}, \mathbf{c}_i).$$

Note that some points in $\mathbf{X}$ may be identical. Here, we consider the situation where every point is a d-dimensional vector with integer coordinates, while the clusters centers are not necessarily from $\mathbf{X}$. Moreover, the coordinates of the cluster center may be real or integer values.

In this work, we need to define the parameterized version of ℓ_p-Equal k-Median Clus-

tering, we call the problem Parameterized ℓ_p-Equal k-Median Clustering with the cost of clustering B (the budget) being the parameter. Following the framework of lossy kernelization [65], when the cost of an optimal clustering exceeds the budget, we assume it is equal to $B+1$. More precisely, in Parameterized ℓ_p-Equal k-Median Clustering, we are given an additional integer B (budget parameter). The task is to find a k-clustering $\{X_1,\ldots,X_k\}$ with $|X_1|=\cdots=|X_k|$ and minimizing the value

$$\mathsf{cost}_p^B(X_1,\ldots,X_k)=\begin{cases}\sum_{i=1}^k \mathsf{cost}_p(X_i) & \text{if } \sum_{i=1}^k \mathsf{cost}_p(X_i)\leq B,\\ B+1 & \text{otherwise.}\end{cases}$$

We believe that restricting the input to the integral values is the most natural model for studying the complexity of Parameterized ℓ_p-Equal k-Median Clustering with respect to the parameter B. Moreover, considering B as a parameter only make sense as input values are suitably discretized which is a common situation when the data is categorical, that is, it can admit a fixed number of possible values. For example, it could be gender, blood type, or political orientation. A prominent example of categorical data is binary data, where the points are binary vectors.

Our first main result is the following theorem providing a polynomial 2-approximate kernel.

Theorem 9. *For every nonnegative integer constant p,* Parameterized ℓ_p-Equal k-Median Clustering *admits a 2-approximate kernel when parameterized by B, where the output collection of points has $\mathcal{O}(B^2)$ points of $\mathbb{Z}^{d'}$ with $d'=\mathcal{O}(B^{p+2})$, where each coordinate of a point takes an absolute value of $\mathcal{O}(B^3)$.*

In other words, the theorem provides a polynomial-time algorithm that compresses the original instance $\mathbf{X}$ to a new instance whose size is bounded by a polynomial of B and such that any c-approximate solution in the new instance can be turned in a polynomial time to a $2c$-approximate solution to the original instance.

A natural question is whether the approximation ratio of the lossy kernel in Theorem 9 is optimal. While we do not have a complete answer to this question, we provide lower bounds supporting our study of the problem from the perspective of approximate kernelization. Our next result rules out the existence of an "exact" kernel for the problem. To state the result, we need to define the decision version of ℓ_p-Equal k-Median Clustering. In this version, we call the problem Decision ℓ_p-Equal k-Median Clustering, the question is whether for a given budget B, there is a k-clustering $\{X_1,\ldots,X_k\}$ with clusters of the same size such that $\sum_{1\leq i\leq k}\mathsf{cost}_p(X_i)\leq B$.

Theorem 10. *For the ℓ_0 and ℓ_1-norms,* Decision ℓ_p-Equal k-Median Clustering *has no polynomial kernel when parameterized by B unless* $\mathsf{NP} \subseteq \mathsf{coNP} / \mathsf{poly}$*, even if the input points are binary, that is, are from $\{0,1\}^d$.*

On the other hand, we prove that Decision ℓ_p-Equal k-Median Clustering admits a polynomial kernel when parameterized by k and B.

Theorem 11. *For every nonnegative integer constant p,* Decision ℓ_p-Equal k-Median Clustering *admits a polynomial kernel when parameterized by k and B, where the output collection of points has $\mathcal{O}(kB)$ points of $\mathbb{Z}^{d'}$ with $d' = \mathcal{O}(kB^{p+1})$ and each coordinate of a point takes an absolute value of $\mathcal{O}(kB^2)$.*

6.1 Lossy Kernel for Parameterized ℓ_p-Equal k-Median Clustering

We briefly sketch the main ideas behind the construction of our lossy kernel for Parameterized ℓ_p-Equal k-Median Clustering. The lossy kernel's main ingredients are a) a polynomial time algorithm based on an algorithm for computing a minimum weight perfect matching in bipartite graphs, b) preprocessing rules reducing the size and dimension of the problem, and c) a greedy algorithm. Each of the steps is relatively simple and easily implementable. However, proving that these steps result in a lossy kernel with the required properties is not easy.

Recall that for a given budget B, we are looking for a k-clustering of a collection of points $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ into k clusters of the same size minimizing the cost. We also assume that the cost is $B+1$ if the instance points do not admit a clustering of cost at most B. Informally, we are only interested in optimal clustering when its cost does not exceed the budget. First, if the cluster's size $s = \frac{n}{k}$ is sufficiently large (with respect to the budget), we can construct an optimal clustering in polynomial time. More precisely, we prove that if $s \geq 4B+1$, then the clusters' medians could be selected from $\mathbf{X}$. Moreover, we show how to identify the (potential) medians in polynomial time. In this case, constructing an optimal k-clustering could be reduced to the classical problem of computing a perfect matching of minimum weight in a bipartite graph.

The case of cluster's size $s \leq 4B$ is different. We apply a set of reduction rules. These rules run in a polynomial time. After exhaustive applications of reduction rules, we either correctly conclude that the considered instance has no clustering of cost at most B or construct an equivalent reduced instance. In the equivalent instance, the dimension

is reduced to $\mathcal{O}(kB^{p+1})$ while the absolute values of the coordinates of the points are in $\mathcal{O}(kB^2)$.

Finally, we apply the only approximate reduction on the reduced instance. The approximation procedure is greedy: whenever there are s equal points, we form a cluster out of them. For the points remaining after the exhaustive application of the greedy procedure, we conclude that either there is no clustering of cost at most B or the number of points is $\mathcal{O}(B^2)$. This construction leads us to the lossy kernel. However, the greedy selection of the clusters composed of equal points may not be optimal. In particular, the reductions used to obtain our algorithmic lower bounds given in Sections 6.2 and 6.3 exploit the property that it may be beneficial to split a block of s equal points between distinct clusters.

Nevertheless, the greedy clustering of equal points leads to a 2-approximation. The proof of this fact requires some work. We evaluate the clustering cost obtained from a given optimal clustering by swapping some points to form clusters composed of equal points. Further, we upper bound the obtained value by the cost of the optimum clustering. For the last step, we introduce an auxiliary clustering problem formulated as a min-cost flow problem. This reduction allows us to evaluate the cost and obtain the required upper bound.

The organisation of the chapter is as follows. In Subsection 6.1.1, we provide some auxiliary results, and in Subsection 6.1.2, we prove the main results. Throughout this section, we assume that $p \geq 0$ defining the ℓ_p-norm is a fixed constant.

6.1.1 Technical Lemmata

We start by proving the following results about the medians of clusters when their size is sufficiently big with respect to the budget.

Lemma 14. *Let $\{X_1, \ldots, X_k\}$ be an equal k-clustering of a collection of points $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ of $\mathbb{Z}^d$ of cost at most $B \in \mathbb{Z}_{\geq 0}$, and let $s = \frac{n}{k}$. Then each cluster X_i for $i \in \{1, \ldots, k\}$ contains at least $s - 2B$ equal points.*

Proof. The claim is trivial if $s \leq 2B + 1$. Let $s \geq 2B + 2$. Assume to the contrary that a cluster X_i has at most $s - 2B - 1$ equal points for some $i \in \{1, \ldots, k\}$. Let $\mathbf{c}_1, \ldots, \mathbf{c}_k$ be optimum medians for the clusters $X_1, \ldots, X_k$, respectively. Then we have that $\mathsf{cost}_p(X_1, \ldots, X_k) = \mathsf{cost}_p(X_1, \ldots, X_k, \mathbf{c}_1, \ldots, \mathbf{c}_k)$.

Let $\mathbf{x}_{i_0} \in X_i$ be a point at the minimum distance from $\mathbf{c}_i$. Since there are at most

$s-2B-1$ points in X_i which are equal to x_{i_0}, there are $t = 2B+1$ points $\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_t} \in X_i$ distinct from x_{i_0}. Observe that

$$\sum_{\mathbf{x}_h \in X_i} \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_h) \geq \sum_{j=0}^{t} \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_{i_j}) \geq \sum_{j=1}^{t} \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_{i_j}). \tag{6.1}$$

Because the points have integer coordinates and by the triangle inequality,

$$1 \leq \mathsf{dist}_p(\mathbf{x}_{i_0}, \mathbf{x}_{i_j}) \leq \mathsf{dist}_p(\mathbf{x}_{i_0}, \mathbf{c}_i) + \mathsf{dist}_p(\mathbf{x}_{i_j}, \mathbf{c}_i) \tag{6.2}$$

for every $j \in \{1, \ldots, t\}$. Since $\mathbf{x}_{i_0}$ is a point of X_i at minimum distance from $\mathbf{c}_i$,

$$\mathsf{dist}_p(\mathbf{x}_{i_0}, \mathbf{c}_i) + \mathsf{dist}_p(\mathbf{x}_{i_j}, \mathbf{c}_i) \leq 2 \cdot \mathsf{dist}_p(\mathbf{x}_{i_j}, \mathbf{c}_i). \tag{6.3}$$

From (6.2) and (6.3), we get $\mathsf{dist}_p(\mathbf{x}_{i_j}, \mathbf{c}_i) \geq \frac{1}{2}$ for $j \in \{1, \ldots, t\}$. Thus, from (6.1), we get

$$\sum_{\mathbf{x}_h \in X_i} \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_h) \geq \sum_{j=1}^{t} \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_{i_j}) \geq \frac{1}{2}t = \frac{1}{2}(2B+1) > B,$$

which is a contradiction with $\mathsf{cost}_p(X_1, \ldots, X_k) \leq B$. This completes the proof. □

Lemma 15. *Let $\{X_1, \ldots, X_k\}$ be an equal k-clustering of a collection of points $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ of $\mathbb{Z}^d$ of cost at most $B \in \mathbb{Z}_{\geq 0}$, and let $s = \frac{n}{k} \geq 4B+1$. Let also $\mathbf{c}_1, \ldots, \mathbf{c}_k \in \mathbb{R}^d$ be optimum medians for $X_1, \ldots, X_k$, respectively. Then for every $i \in \{1, \ldots, k\}$, $\mathbf{c}_i = \mathbf{x}_j$ for $\mathbf{x}_j \in \mathbf{X}_i$ such that $\mathbf{X}_i$ contains at least $s-2B$ points that are equal to $\mathbf{x}_j$ and the choice of $\mathbf{c}_i$ is unique.*

Proof. Consider a cluster X_i with the median $\mathbf{c}_i$ for arbitrary $i \in \{1, \ldots, k\}$. Since $s \geq 4B+1$, then by Lemma 14, there is $\mathbf{x}_j \in X_i$ such that X_i contains at least $s-2B$ points that are equal to $\mathbf{x}_j$. We show that $\mathbf{c}_i = \mathbf{x}_j$. Notice that the choice of the set of at least $s-2B$ equal points is unique because X_i can contain at most $s-(s-2B) = 2B$ distinct from $\mathbf{x}_j$ points, and since $s \geq 4B+1$, $s-2B \geq 2B+1 > 2B$.

The proof is by contradiction. Assume that $\mathbf{c}_i \neq \mathbf{x}_j$. Let $S \subseteq \{1, \ldots, n\}$ be the set of indices of the points $\mathbf{x}_h \in X_i$ that coincide with $\mathbf{x}_j$, and denote by T the set of indices of the remaining points in X_i. We know that $|T| \leq 2B < |S|$ because $s \geq 4B+1$ and

$|S| \geq 2B+1$. Then

$$\begin{aligned}\mathsf{cost}_p(X_i) =& \mathsf{cost}_p(X_i, \mathbf{c}_i) = \sum_{h \in X_i} \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_h) \\ =& \sum_{h \in S} \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_h) + \sum_{h \in T} \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_h) \\ =& (|S| - |T|) \cdot \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_j) + \sum_{h \in T} (\mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_j) + \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_h)).\end{aligned} \tag{6.4}$$

On using the triangle inequality, we get

$$\begin{aligned}(|S| - |T|) \cdot \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_j) + & \sum_{h \in T} (\mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_j) + \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_h)) \\ \geq & (|S| - |T|) \cdot \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_j) + \sum_{h \in T} \mathsf{dist}_p(\mathbf{x}_j, \mathbf{x}_h).\end{aligned} \tag{6.5}$$

We know that $(|S| - |T|) \cdot \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_j) > 0$ because $|S| > |T|$ and $\mathbf{c}_i \neq \mathbf{x}_j$. Then by (6.5), we have

$$(|S| - |T|) \cdot \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_j) + \sum_{h \in T} \mathsf{dist}_p(\mathbf{x}_j, \mathbf{x}_h) > \sum_{h \in T} \mathsf{dist}_p(\mathbf{x}_j, \mathbf{x}_h). \tag{6.6}$$

Combining (6.4)–(6.6), we conclude that $\mathsf{cost}_p(X_i) > \sum_{h \in T} \mathsf{dist}_p(\mathbf{x}_j, \mathbf{x}_h)$. Let $\mathbf{c}'_i = \mathbf{x}_j$. Then

$$\begin{aligned}\mathsf{cost}_p(X_i, \mathbf{c}'_i) = & \sum_{h \in X_i} \mathsf{dist}_p(\mathbf{c}'_i, \mathbf{x}_h) = \sum_{h \in S} \mathsf{dist}_p(\mathbf{c}'_i, \mathbf{x}_h) + \sum_{h \in T} \mathsf{dist}_p(\mathbf{c}'_i, \mathbf{x}_h) \\ = & \sum_{h \in T} \mathsf{dist}_p(\mathbf{c}'_i, \mathbf{x}_h) < \mathsf{cost}_p(X_i)\end{aligned}$$

which contradicts that $\mathbf{c}_i$ is an optimum median for X_i. This concludes the proof. □

We use the following lemma to identify medians.

Lemma 16. *Let $\{X_1, \ldots, X_k\}$ be an equal k-clustering of a collection of points $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ of $\mathbb{Z}^d$ of cost at most $B \in \mathbb{Z}_{\geq 0}$, and let $s = \frac{n}{k} \geq 4B+1$. Suppose that $Y \subseteq \mathbf{X}$ is a collection of at least $B+1$ equal points of $\mathbf{X}$. Then there is an $i \in \{1, \ldots, k\}$ such that an optimum median of X_i coincides with $\mathbf{x}_j$ for $\mathbf{x}_j \in Y$.*

Proof. Let $\mathbf{c}_1, \ldots . \mathbf{c}_k$ be optimum medians of $X_1, \ldots, X_k$, respectively. Since $s \geq 4B+1$, then by Lemma 15, for every $i \in \{1, \ldots, k\}$, $\mathbf{c}_i$ coincides with some element $\mathbf{x}_h$ of the cluster X_i. For the sake of contradiction, assume that $\mathbf{c}_1, \ldots, \mathbf{c}_k$ are distinct from $\mathbf{x}_j \in Y$. This means that $\mathsf{dist}_p(\mathbf{x}_j - \mathbf{c}_i) \geq 1$ because the coordinates of the points of $\mathbf{X}$ are integer.

Then

$$\begin{aligned}\mathsf{cost}_p(X_1,\ldots,X_k) =& \sum_{i=1}^{k}\mathsf{cost}_p(X_i,\mathbf{c}_i) \geq \sum_{i=1}^{k}\sum_{\mathbf{x}_h\in Y\cap X_i}\mathsf{dist}_p(\mathbf{c}_i,\mathbf{x}_h) \geq \sum_{i=1}^{k}|X_i\cap Y|\\ =&|Y| \geq B+1 > B,\end{aligned}$$

contradicting that $\mathsf{cost}_p(X_1,\ldots,X_k) \leq B$. This proves the lemma. □

We use our next lemma to upper bound the clustering cost if we collect $s = \frac{n}{k}$ equal points in the same cluster.

Lemma 17. *Let* $\{X_1,\ldots,X_k\}$ *be an* equal k-clustering *of a collection of points* $\mathbf{X} = \{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ *of* $\mathbb{Z}^d$*, and let* $\mathbf{c}_1,\ldots,\mathbf{c}_k \in \mathbb{R}^d$*. Suppose that* S *is a collection of* $s = \frac{n}{k}$ *equal points of* $\mathbf{X}$ *and* $\mathbf{x}_j \in S$*. Then there is an equal* k*-clustering* $\{X'_1,\ldots,X'_k\}$ *of* $\mathbf{X}$ *with* $X'_1 = S$ *such that*

$$\mathsf{cost}_p(X'_1,\ldots,X'_k,\mathbf{c}'_1,\ldots,\mathbf{c}'_k) \leq \mathsf{cost}_p(X_1,\ldots,X_k,\mathbf{c}_1,\ldots,\mathbf{c}_k) + s\cdot\mathsf{dist}_p(\mathbf{c}_1,\mathbf{x}_j),$$

where $\mathbf{c}'_1 = \mathbf{x}_j$ *and* $\mathbf{c}'_h = \mathbf{c}_h$ *for* $h \in \{2,\ldots,k\}$.

Proof. The claim is trivial if $S = X_1$ because we can set $X'_i = X_i$ for $i \in \{1,\ldots,k\}$. Assume that this is not the case and there are elements of S that are not in X_1; denote by $\mathbf{x}_{i_1},\ldots,\mathbf{x}_{i_t}$ these elements. We assume that $\mathbf{x}_{i_h} \in X_{i'_h}$, for $h \in \{1,\ldots,t\}$ for $i'_h \geq 2$. Because $|S| = s$, there are $\mathbf{x}_{j_1},\ldots,\mathbf{x}_{j_t} \in X_1$ such that $\mathbf{x}_{j_1},\ldots,\mathbf{x}_{j_t} \notin S$. We construct $X'_1,\ldots,X'_k$ from $X_1,\ldots,X_k$ by exchanging the points $\mathbf{x}_{j_h}$ and $\mathbf{x}_{i_h}$ between X_1 and $X_{i'_h}$ for every $h \in \{1,\ldots,t\}$. Notice that $|X'_1| = \cdots = |X'_k|$ because the exchanges do not modify the sizes of the clusters. Thus, $\{X'_1,\ldots,X'_k\}$ is an equal k-clustering. We claim that $\{X'_1,\ldots,X'_k\}$ satisfies the required property.

We have that

$$\begin{aligned}&\mathsf{cost}(X'_1,\ldots,X'_k,\mathbf{c}'_1,\ldots,\mathbf{c}'_k) - \mathsf{cost}(X_1,\ldots,X_k,\mathbf{c}_1,\ldots,\mathbf{c}_k)\\ &= \sum_{i=1}^{k}\sum_{\mathbf{x}_h\in X'_i}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}'_i) - \sum_{i=1}^{k}\sum_{\mathbf{x}_h\in X_i}\mathsf{dist}_p(\mathbf{x},\mathbf{c}_i)\\ &= \sum_{\mathbf{x}_h\in X'_1}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}'_1) - \sum_{\mathbf{x}_h\in X_1}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}_1)\\ &\quad + \sum_{i=2}^{k}\Big(\sum_{\mathbf{x}_h\in X'_i}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}'_i) - \sum_{\mathbf{x}_h\in X_i}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}_i)\Big).\end{aligned} \tag{6.7}$$

Note that $\sum_{\mathbf{x}_h\in X'_1}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}'_1) = 0$ and $\sum_{\mathbf{x}_h\in X_1}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}_1) \geq \sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{j_h},\mathbf{c}_1)$. Also

by the construction of $X'_1,\ldots,X'_k$ and because $\mathbf{c}_i=\mathbf{c}'_i$ for $i\in\{2,\ldots,k\}$, we have that

$$\begin{aligned}\sum_{i=2}^{k}\Big(\sum_{\mathbf{x}_h\in X'_i}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}'_i)-\sum_{\mathbf{x}_h\in X_i}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}_i)\Big)&=\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{j_h},\mathbf{c}'_{i_h})-\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{i_h})\\&=\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{j_h},\mathbf{c}_{i_h})-\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{i_h}).\end{aligned}$$

Then extending (6.7) and applying the triangle inequality twice, we obtain that

$$\begin{aligned}&\mathsf{cost}(X'_1,\ldots,X'_k,\mathbf{c}'_1,\ldots,\mathbf{c}'_k)-\mathsf{cost}(X_1,\ldots,X_k,\mathbf{c}_1,\ldots,\mathbf{c}_k)\\&\leq-\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{j_h},\mathbf{c}_1)+\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{j_h},\mathbf{c}_{i_h})-\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{i_h})\\&=\sum_{h=1}^{t}\big(-\mathsf{dist}_p(\mathbf{x}_{j_h},\mathbf{c}_1)+\mathsf{dist}_p(\mathbf{x}_{j_h},\mathbf{c}_{i_h})-\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{i_h})\big)\\&\leq\sum_{h=1}^{t}\big(\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{i_h})-\mathsf{dist}_p(\mathbf{c}_1,\mathbf{c}_{i_h})\big)\\&\leq\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_1)\leq t\cdot\mathsf{dist}_p(\mathbf{x}_j,\mathbf{c}_1)\leq s\cdot\mathsf{dist}_p(\mathbf{x}_j,\mathbf{c}_1)\end{aligned}$$

as required by the lemma. □

Our next lemma shows that we can solve Parameterized ℓ_p-Equal k-Median Clustering in a polynomial time if the cluster size is sufficiently big with respect to the budget.

Lemma 18. *There is a polynomial-time algorithm that, given a collection* $\mathbf{X}=\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ *of* n *points of* $\mathbb{Z}^d$*, a positive integer* k *such that* n *is divisible by* k*, and a nonnegative integer* B *such that* $\frac{n}{k}\geq 4B+1$*, either computes* $\mathsf{Opt}(X,k)\leq B$ *and produces an equal* k*-clustering of minimum cost or correctly concludes that* $\mathsf{Opt}(\mathbf{X},k)>B$*.*

Proof. Let $\mathbf{X}=\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ be a collection of n points of $\mathbb{Z}^d$ and let k be a positive integer such that n is divisible by k, and suppose that $s=\frac{n}{k}\geq 4B+1$ for a nonnegative integer B.

First, we exhaustively apply the following reduction rule.

Reduction Rule 1. *If* $\mathbf{X}$ *contains a collection of* s *equal points* S*, then set* $\mathbf{X}:=\mathbf{X}\setminus S$ *and* $k:=k-1$*.*

To argue that the rule is safe, let $\mathbf{X}'=\mathbf{X}\setminus S$, where S is a collection of s equal

points of $\mathbf{X}$, and let $k' = k$. Clearly, $\mathbf{X}'$ contains $n' = n - s$ points and $\frac{n'}{k'} = s$. If $\{X'_1, \ldots, X'_{k'}\}$ is an equal k'-clustering of $\mathbf{X}'$, then $\{S, X'_1, \ldots, X'_{k'}\}$ is an equal k-clustering of $\mathbf{X}$. Note that $\mathsf{cost}_p(S) = 0$ because the elements of S are the same. Then $\mathsf{cost}_p(S, X'_1, \ldots, X'_{k'}) = \mathsf{cost}_p(X'_1, \ldots, X'_{k'})$. Therefore, $\mathsf{Opt}(\mathbf{X}, k) \leq \mathsf{Opt}(\mathbf{X}', k')$. We show that if $\mathsf{Opt}(\mathbf{X}, k) \leq B$, then $\mathsf{Opt}(\mathbf{X}, k) \geq \mathsf{Opt}(\mathbf{X}', k')$.

Suppose that $\{X_1, \ldots, X_k\}$ is an equal k-clustering of $\mathbf{X}$ with $\mathsf{cost}_p(X_1, \ldots, X_k) = \mathsf{Opt}(X, k) \leq B$. Denote by $\mathbf{c}_1, \ldots, \mathbf{c}_k$ optimum medians of $X_1, \ldots, X_k$, respectively. Because $|S| = s \geq 4B + 1 \geq B + 1$, there is a cluster whose optimum median is $\mathbf{x}_j$ for $\mathbf{x}_j \in S$. We assume without loss of generality that X_1 is such a cluster and $\mathbf{c}_1 = \mathbf{x}_j$. By Lemma 17, there is a k-clustering $\{S, X'_2, \ldots, X'_k\}$ of $\mathbf{X}$ such that $\mathsf{cost}_p(S, X'_2, \ldots, X'_k, \mathbf{c}'_1, \ldots, \mathbf{c}'_k) \leq \mathsf{cost}_p(X_1, \ldots, X_k, \mathbf{c}_1, \ldots, \mathbf{c}_k) + s \cdot \mathsf{dist}_p(\mathbf{c}_1, \mathbf{x}_j)$, where $\mathbf{c}'_1 = \mathbf{x}_j$ and $\mathbf{c}'_h = \mathbf{c}_h$ for $h \in \{2, \ldots, k\}$. Because $\mathbf{c}_1 = \mathbf{x}_j$, we conclude that $\mathsf{cost}_p(X'_2, \ldots, X'_k) = \mathsf{cost}_p(S, X'_2, \ldots, X'_k, \mathbf{c}'_1, \ldots, \mathbf{c}'_k) \leq \mathsf{cost}_p(X_1, \ldots, X_k, \mathbf{c}_1, \ldots, \mathbf{c}_k) = \mathsf{Opt}(X, k)$. Since $\{X'_2, \ldots, X'_k\}$ is a k'-clustering of $\mathbf{X}'$, we have that $\mathsf{Opt}(\mathbf{X}', k') \leq \mathsf{cost}_p(X'_2, \ldots, X'_k) \leq \mathsf{Opt}(X, k)$ as required.

We obtain that either $\mathsf{Opt}(\mathbf{X}, k) = \mathsf{Opt}(\mathbf{X}', k') \leq B$ or $\mathsf{Opt}(\mathbf{X}, k) > B$ and $\mathsf{Opt}(\mathbf{X}', k') > B$. Notice also that, given an optimum equal k'-clustering of $\mathbf{X}'$, we can construct the optimum k-clustering of X, by making S a cluster. Thus, it is sufficient to prove the lemma for the collection of points obtained by the exhaustive application of Reduction Rule 1. Note that if this collection is empty, then $\mathsf{Opt}(\mathbf{X}, k) = 0$ and the lemma holds. This allows us to assume from now that $\mathbf{X}$ is nonempty and has no s equal points.

Suppose that $\{X_1, \ldots, X_k\}$ be an equal k-clustering with $\mathsf{cost}_p(X_1, \ldots, X_k) = \mathsf{Opt}(\mathbf{X}, k) \leq B$. By Lemma 15, we have that for every $i \in \{1, \ldots, k\}$, the optimum median $\mathbf{c}_i$ for X_i is unique and $\mathbf{c}_i = \mathbf{x}_j$ for $\mathbf{x}_j \in \mathbf{X}_i$ such that $\mathbf{X}_i$ contains at least $s - 2B$ points that are equal to $\mathbf{x}_j$. Notice that $\mathbf{c}_1, \ldots, \mathbf{c}_k$ are pairwise distinct because a collection of equal points cannot be split between distinct clusters in such a way that each of these clusters would contain at least $s - 2B$ points. This holds because any collection of equal points of $\mathbf{X}$ contains at most $s - 1$ elements and $2(s - 2B) > s$ as $s \geq 4B + 1$. By Lemma 16, we have that if $\mathbf{X}$ contains a collection of equal points S of size $B + 1 \leq s - 2B$, then one of the optimum medians should be equal to a point from S.

These observations allow us to construct (potential) medians $\mathbf{c}_1, \ldots, \mathbf{c}_t$ as follows: we iteratively compute inclusion maximal collections S of equal points of $\mathbf{X}$ and if $|S| \geq B + 1$, we set the next median $\mathbf{c}_i$ be equal to a point of S. If the number of constructed potential medians $t \neq k$, we conclude that $\mathbf{X}$ has no equal k-clustering of cost at most B. Otherwise, if $t = k$, we have that $\mathbf{c}_1, \ldots, \mathbf{c}_k$ should be optimum medians for an equal k-clustering of minimum cost if $\mathsf{Opt}(\mathbf{X}, k) \leq B$.

Then we compute in a polynomial time an equal k-clustering $\{X_1, \ldots, X_k\}$ of $\mathbf{X}$ that minimizes $\mathsf{cost}_p(X_1, \ldots, X_k, \mathbf{c}_1, \ldots, \mathbf{c}_k)$ using Lemma 1. If $\mathsf{cost}_p(X_1, \ldots, X_k, \mathbf{c}_1, \ldots, \mathbf{c}_k) > B$, then we conclude that $\mathsf{Opt}(\mathbf{X}, k) > B$. Otherwise, we have that $\mathsf{Opt}(\mathbf{X}, k) = \mathsf{cost}_p(X_1, \ldots, X_k, \mathbf{c}_1, \ldots, \mathbf{c}_k)$ and $\{X_1, \ldots, X_k\}$ is an equal k-clustering of minimum cost. □

Our next aim is to show that we can reduce the dimension and the absolute values of the coordinates of the points if $\mathsf{Opt}(X, k) \leq B$. To achieve this, we mimic some ideas of the kernelization algorithm of Fomin et al. in [37] for the related clustering problem. However, they considered only points from $\{0, 1\}^d$ and the Hamming norm.

Lemma 19. *There is a polynomial-time algorithm that, given a collection $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ of n points of $\mathbb{Z}^d$, a positive integer k such that n is divisible by k, and a nonnegative integer B, either correctly concludes that $\mathsf{Opt}(\mathbf{X}, k) > B$ or computes a collection of n points $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ of $\mathbb{Z}^{d'}$ such that the following holds:*

(i) For every partition $\{I_1, \ldots, I_k\}$ of $\{1, \ldots, n\}$ such that $|I_1| = \cdots = |I_k| = \frac{n}{k}$, either $\mathsf{cost}_p(X_1, \ldots, X_k) > B$ and $\mathsf{cost}_p(Y_1, \ldots, Y_k) > B$ or $\mathsf{cost}_p(X_1, \ldots, X_k) = \mathsf{cost}_p(Y_1, \ldots, Y_k)$, where $X_i = \{\mathbf{x}_h \mid h \in I_i\}$ and $Y_i = \{\mathbf{y}_h \mid h \in I_i\}$ for every $i \in \{1, \ldots, k\}$.

(ii) $d' = \mathcal{O}(kB^{p+1})$.

(iii) $|\mathbf{y}_i[h]| = \mathcal{O}(kB^2)$ for $h \in \{1, \ldots, d'\}$ and $i \in \{1, \ldots, n\}$.

Proof. Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be a collection of n points of $\mathbb{Z}^d$ and let k be a positive integer such that n is divisible by k. Let also B be a nonnegative integer.

We iteratively construct the partition $S = \{S_1, \ldots, S_t\}$ of $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ using the following greedy algorithm. Let $j \geq 1$ be an integer and suppose that the sets $S_0, \ldots, S_{j-1}$ are already constructed assuming that $S_0 = \emptyset$. Let $\mathbf{Z} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \setminus \bigcup_{i=0}^{j-1} S_i$. If $\mathbf{Z} = \emptyset$, then the construction of S is completed. If $\mathbf{Z} \neq \emptyset$, we construct S_j as follows:

- set $S_j := \{\mathbf{x}_h\}$ for arbitrary $\mathbf{x}_h \in \mathbf{Z}$ and set $\mathbf{Z} := \mathbf{Z} \setminus \{\mathbf{x}_h\}$,
- while there is $\mathbf{x}_r \in \mathbf{Z}$ such that $\mathsf{dist}_p(\mathbf{x}_r, \mathbf{x}_{r'}) \leq B$ for some $\mathbf{x}_{r'} \in S_j$, set $S_j := S_j \cup \{\mathbf{x}_r\}$ and set $\mathbf{Z} = \mathbf{Z} \setminus \{\mathbf{x}_r\}$.

The crucial property of the partition S is that every cluster of an equal k-clustering of cost at most B is entirely in some part of the partition.

Claim 6.1.1. *Let $\{X_1, \ldots, X_k\}$ be an equal k-clustering of $\mathbf{X}$ of cost at most B. Then for every $i \in \{1, \ldots, k\}$ there is a $j \in \{1, \ldots, t\}$ such that $X_i \subseteq S_j$.*

Proof of Claim 6.1.1. Denote by $\mathbf{c}_1, \ldots, \mathbf{c}_k \in \mathbb{R}^d$ the optimum medians for the clusters $X_1, \ldots, X_k$, respectively. Assume to the contrary that there is a cluster X_i such that $\mathbf{x}_u, \mathbf{x}_v \in X_i$ with $\mathbf{x}_u$ and $\mathbf{x}_v$ in distinct collections of the partition $\{S_1, \ldots, S_t\}$. Then $\mathsf{dist}_p(\mathbf{x}_u, \mathbf{x}_v) > B$ by the construction of $S_1, \ldots, S_t$ and

$$\begin{aligned}\mathsf{cost}_p(X_1, \ldots, X_k) \geq & \mathsf{cost}_p(X_i) = \mathsf{cost}_p(X_i, \mathbf{c}_i) \geq \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_u) + \mathsf{dist}_p(\mathbf{c}_i, \mathbf{x}_v) \\ \geq & \mathsf{dist}_p(\mathbf{x}_u, \mathbf{x}_v) > B\end{aligned}$$

contradicting that $\mathsf{cost}_p(X_1, \ldots, X_k) \leq B$. □

From the above Claim 6.1.1, we have that if $t > k$, then $\mathbf{X}$ has no equal k-clustering of cost at most B, that is, $\mathsf{Opt}(X, B) > B$. In this case, we return this answer and stop. From now on, we assume that this is not the case and $t \leq k$.

By Lemma 14, at least $\frac{n}{k} - 2B$ points in every cluster of an equal k-clustering of cost at most B are the same. Thus, if $\{X_1, \ldots, X_k\}$ is an equal k-clustering of cost at most B, then for each $i \in \{1, \ldots, k\}$, X_i contains at most $2B+1$ distinct points. By Claim 6.1.1, we obtain that for every $i \in \{1, \ldots, t\}$, S_i should contain at most $k(2B+1)$ distinct points if $\mathbf{X}$ admits an equal k-clustering of cost at most B. Then for each $i \in \{1, \ldots, t\}$, we compute the number of distinct points in S_i and if this number is bigger than $k(2B+1)$, we conclude that $\mathsf{Opt}(\mathbf{X}, k) > B$. In this case, we return this answer and stop. From now, we assume that this is not the case and each S_i for $i \in \{1, \ldots, t\}$ contains at most $k(2B+1)$ distinct points.

For a collection of points $Z \subseteq \mathbf{X}$, we say that a coordinate $h \in \{1, \ldots, d\}$ is *uniform* for Z if $\mathbf{x}_j[h]$ is the same for all $\mathbf{x}_j \in Z$ and h is *nonuniform* otherwise.

Let ℓ_i be the number of nonuniform coordinates for S_i for $i \in \{1, \ldots, t\}$, and let $\ell = \max_{1 \leq i \leq t} \ell_i$. For each $i \in \{1, \ldots, t\}$, we select a set of indices $R_i \subseteq \{1, \ldots, d\}$ of size ℓ such that R_i contains all nonuniform coordinates for S_i. Note that R_i may be empty if $\ell = 0$. We also define a set of coordinates $T_i = \{1, \ldots, d\} \setminus R_i$, for $i \in \{1, \ldots, t\}$.

For every $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, t\}$ such that $x_i \in S_j$, we define an $(\ell+1)$-dimensional point $\mathbf{x}'_i$, where $\mathbf{x}'_i[1, \ldots, \ell] = \mathbf{x}_i[R_j]$ and $\mathbf{x}'_i[\ell+1] = (j-1)(B+1)$. This way we obtain a collection of points $\mathbf{X}' = \{\mathbf{x}'_1, \ldots, \mathbf{x}'_n\}$. For every $j \in \{1, \ldots, t\}$, we define $S'_j = \{\mathbf{x}'_h \mid \mathbf{x}_h \in S_j\}$, that is, we construct the partition $S' = \{S'_1, \ldots, S'_t\}$ of $\{\mathbf{x}'_1, \ldots, \mathbf{x}'_n\}$ corresponding to S.

For each $i \in \{1, \ldots, t\}$, we do the following:

- For each $h \in \{1, \ldots, \ell\}$, we find $M_h^{(i)} = \min\{\mathbf{x}'_j[h] \mid \mathbf{x}'_j \in S'_i\}$.
- For every $\mathbf{x}'_j \in S'_i$, we define a new point $\mathbf{y}_j$ by setting $\mathbf{y}_j[h] = \mathbf{x}'_j[h] - M_h^{(i)}$ for $h \in \{1, \ldots, \ell\}$ and $\mathbf{y}_j[\ell+1] = \mathbf{x}'_j[\ell+1] = (j-1)(B+1)$.

This way, we construct the collection $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ of points from $\mathbb{Z}^{\ell+1}$. Our algorithm returns this collection of points.

It is easy to see that the described algorithm runs in a polynomial time. We show that if the algorithm outputs $\mathbf{Y}$, then this collection of the points satisfies conditions (i)–(iii) of the lemma.

To show (i), let $\{I_1, \ldots, I_k\}$ be a partition of $\{1, \ldots, n\}$ such that $|I_1| = \cdots = |I_k| = \frac{n}{k}$, and let $X_i = \{\mathbf{x}_h \mid h \in I_i\}$ and $Y_i = \{\mathbf{y}_h \mid h \in I_i\}$ for every $i \in \{1, \ldots, k\}$. We show that either $\mathsf{cost}_p(X_1, \ldots, X_k) > B$ and $\mathsf{cost}_p(Y_1, \ldots, Y_k) > B$ or $\mathsf{cost}_p(X_1, \ldots, X_k) = \mathsf{cost}_p(Y_1, \ldots, Y_k)$.

Suppose that $\mathsf{cost}_p(X_1, \ldots, X_k) \leq B$. Consider $i \in \{1, \ldots, k\}$ and denote by $\mathbf{c}_i$ the optimum median for X_i. By Claim 6.1.1, there is a $j \in \{1, \ldots, t\}$ such that $X_i \subseteq S_j$. We define $\mathbf{c}'_i \in \mathbb{R}^{\ell+1}$ by setting $\mathbf{c}'_i[1, \ldots, \ell] = \mathbf{c}_i[R_j]$ and $\mathbf{c}'_i[\ell+1] = (j-1)(B+1)$. Further, we consider $\mathbf{c}''_i \in \mathbb{R}^{\ell+1}$ such that $\mathbf{c}''_i[h] = \mathbf{c}'_i[h] - M_h^{(j)}$ for $h \in \{1, \ldots, \ell\}$ and $\mathbf{c}''_i[\ell+1] = (j-1)(B+1)$. Then by the definitions of $\mathbf{X}'_i$ and $\mathbf{Y}_i$, we have that

$$\mathsf{cost}_p(X_i) = \mathsf{cost}_p(X_i, \mathbf{c}_i) = \mathsf{cost}_p(X'_i, \mathbf{c}'_i) = \mathsf{cost}_p(Y_i, \mathbf{c}''_i) \geq \mathsf{cost}_p(Y_i).$$

This implies that $\mathsf{cost}_p(X_1, \ldots, X_k) \geq \mathsf{cost}_p(Y_1, \ldots, Y_k)$.

For the opposite direction, assume that $\mathsf{cost}_p(Y_1, \ldots, Y_k) \leq B$. Similarly to S', for every $j \in \{1, \ldots, t\}$, we define $S''_j = \{\mathbf{y}_h \mid \mathbf{x}_h \in S_j\}$, that is, we construct the partition $S'' = \{S''_1, \ldots, S''_t\}$ of $\mathbf{Y}$ corresponding to S. We claim that for each $i \in \{1, \ldots, k\}$, there is $j \in \{1, \ldots, t\}$ such that $Y_i \subseteq S_j$.

The proof is by contradiction and is similar to the proof of Claim 6.1.1. Assume that there is $i \in \{1, \ldots, k\}$ such that there are $\mathbf{y}_u, \mathbf{y}_v \in Y_i$ belonging to distinct sets of S''. Then $\mathsf{dist}_p(\mathbf{y}_u, \mathbf{y}_v) \geq |\mathbf{y}_u[\ell+1] - \mathbf{y}_v[\ell+1]| > B$ by the construction of $S''_1, \ldots, S''_t$. Then

$$\begin{aligned}\mathsf{cost}_p(Y_1, \ldots, Y_k) \geq& \mathsf{cost}_p(Y_i) = \mathsf{cost}_p(Y_i, \mathbf{c}_i) \geq \mathsf{dist}_p(\mathbf{c}_i, \mathbf{y}_u) + \mathsf{dist}_p(\mathbf{c}_i, \mathbf{y}_v)\\ \geq& \mathsf{dist}_p(\mathbf{y}_u, \mathbf{y}_v) > B,\end{aligned}$$

where $\mathbf{c}_i$ is an optimum median of Y_i. However, this contradicts that $\mathsf{cost}_p(Y_1,\ldots,Y_k) \leq B$.

Consider $i \in \{1,\ldots,k\}$ and let $\mathbf{c}''_i \in \mathbb{R}^{\ell+1}$ an optimum median for Y_i. Let also $j \in \{1,\ldots,t\}$ be such that $Y_i \subseteq S_j$. Notice that $\mathbf{c}''_i[\ell+1] = (j-1)(B+1)$ by the definition of S_j. We define $\mathbf{c}'_i \in \mathbb{R}^{\ell+1}$ by setting $\mathbf{c}'_i[h] = \mathbf{c}''_i[h] + M_h^{(j)}$ for $h \in \{1,\ldots,\ell\}$ and $\mathbf{c}'_i[\ell+1] = \mathbf{c}''_i[\ell+1] = (j-1)(B+1)$. Then we define $\mathbf{c}_i \in \mathbb{R}^d$, by setting $\mathbf{c}_i[R_j] = \mathbf{c}'_i[1,\ldots,\ell]$ and $\mathbf{c}_i[T_j] = \mathbf{x}_h[T_j]$ for arbitrary $\mathbf{x}_h \in S_j$. Because the coordinates in T_j are uniform for S_j, the values in each coordinate $h \in T_j$ of the coordinates of the points of X_i are the same. This implies that

$$\mathsf{cost}_p(X_i) \leq \mathsf{cost}_p(X_i,\mathbf{c}_i) = \mathsf{cost}_p(X'_i,\mathbf{c}'_i) = \mathsf{cost}_p(Y_i,\mathbf{c}''_i) = \mathsf{cost}_p(Y_i).$$

Hence, $\mathsf{cost}_p(X_1,\ldots,X_k) \leq \mathsf{cost}_p(Y_1,\ldots,Y_k)$. This completes the proof of (i).

To show (ii), we prove that $\ell \leq kB^p(2B+1)$. For this, we show that $\ell_i \leq kB^p(2B+1)$ for every $i \in \{1,\ldots,t\}$. Consider $i \in \{1,\ldots,t\}$. Recall that S_i contains at most $k(2B+1)$ distinct points. Denote by $\mathbf{x}_{j_1},\ldots,\mathbf{x}_{j_r}$ the distinct points in X_i and assume that they are numbered in the order in which they are included in S_i by the greedy procedure constructing this set.

Let $Z_q = \{\mathbf{x}_{j_1},\ldots,\mathbf{x}_{j_q}\}$ for $q \in \{1,\ldots,r\}$. We claim that Z_q has at most $(q-1)B^p$ nonuniform coordinates for each $q \in \{1,\ldots,r\}$. The proof is by induction. The claim is trivial if $q=1$. Let $q>1$ and assume that the claim is fulfilled for Z_{q-1}. By the construction of S_i, $\mathbf{x}_{j_q}$ is at distance at most B from $\mathbf{x}_{j_h}$ for some $h \in \{1,\ldots,q-1\}$. Then because $\mathsf{dist}_p(\mathbf{x}_{j_q},\mathbf{x}_{j_h}) \leq B$, we obtain that the points $\mathbf{x}_{i_q}$ and $\mathbf{x}_{i_h}$ differ in at most B^p coordinates by the definition of the ℓ_p-norm. Then because Z_{q-1} has at most $(q-2)B^p$ nonuniform coordinates, Z_q has at most $(q-1)B^p$ nonuniform coordinates as required.

Because the number of nonuniform coordinates for S_i is the same as the number of nonuniform coordinates for Z_r and $r \leq k(2B+1)$, we obtain that $\ell_i \leq kB^p(2B+1)$. Then $\ell = \max_{1\leq i\leq t} \ell_i \leq kB^p(2B+1)$. Because the points of $\mathbf{Y}$ are in $\mathbb{Z}^{\ell+1}$, we have the required upper bound for the dimension. This concludes the proof of (ii).

Finally, to show (iii), we again exploit the property that every S_i contains at most $k(2B+1)$ distinct points. Let $i \in \{1,\ldots,t\}$ and $h \in \{1,\ldots,d\}$ and denote by $\mathbf{x}_{j_1},\ldots,\mathbf{x}_{j_r}$ the distinct points in X_i. Let $h \in \{1,\ldots,d\}$. We can assume without loss of generality that $\mathbf{x}_{j_1}[h] \leq \cdots \leq \mathbf{x}_{j_r}[h]$. We claim that $\mathbf{x}_{j_r}[h] - \mathbf{x}_{j_1}[h] \leq B(k(2B+1)-1)$. This is trivial if $r=1$. Assume that $r>1$. Observe that $\mathbf{x}_{j_q}[h] - \mathbf{x}_{q-1}[h] \leq B$ for $q \in \{2,\ldots,r\}$. Otherwise, if there is $q \in \{2,\ldots,r\}$ such that $\mathbf{x}_{j_q}[h] - \mathbf{x}_{q-1}[h] > B$, then the distance

from any point in $\{\mathbf{x}_{j_1},\ldots,\mathbf{x}_{j_{q-1}}\}$ to any point in $\{\mathbf{x}_{j_q},\ldots,\mathbf{x}_{j_r}\}$ is more than B but this contradicts that these points are the distinct points of S_i. Then because $\mathbf{x}_{j_q}[h]-\mathbf{x}_{q-1}[h]\leq B$ for $q\in\{2,\ldots,r\}$ and $r\leq k(2B+1)$, we obtain that $\mathbf{x}_{j_r}[h]-\mathbf{x}_{j_1}[h]\leq B(k(2B+1)-1)$.

Then, by the definition of $\mathbf{x}'_1,\ldots,\mathbf{x}'_n$, we obtain that for every $\mathbf{x}'_q,\mathbf{x}'_r\in S'_i$ for some $i\in\{1,\ldots,t\}$ and every $h\in\{1,\ldots,\ell\}$, $|\mathbf{x}'_q[h]-\mathbf{x}'_r[h]|\leq B(k(2B+1)-1)$. By the definition of $M_h^{(i)}$ for $i\in\{1,\ldots,t\}$, we obtain that $|\mathbf{y}_j[h]|\leq B(k(2B+1)-1)$ for every $j\in\{1,\ldots,n\}$ and every $h\in\{1,\ldots,\ell\}$. Because $|\mathbf{y}_j[\ell+1]|\leq(k-1)(B+1)$, we have that $|\mathbf{y}_i[h]|\leq B(k(2B+1)-1)$ for $h\in\{1,\ldots,d'\}$ and $i\in\{1,\ldots,n\}$. This completes the proof of (iii) and the proof of the lemma. □

Finally in this subsection, we show the following lemma that is used to upper bound the additional cost incurred by the greedy clustering of blocks of equal points.

Lemma 20. *Let $\mathbf{X}=\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ be a collection of n points of $\mathbb{Z}^d$ and let k be a positive integer such that n is divisible by k. Suppose that $S_1,\ldots,S_t$ are disjoint collections of equal points of $\mathbf{X}$ such that $|S_1|=\cdots=|S_t|=\frac{n}{k}$ and $\mathbf{Y}=\mathbf{X}\setminus\big(S_1\cup\cdots\cup S_t\big)$. Then $\mathsf{Opt}(\mathbf{Y},k-t)\leq 2\cdot\mathsf{Opt}(\mathbf{X},k)$.*

Proof. Let $\{X_1,\ldots,X_k\}$ be an optimum equal k-clustering of $\mathbf{X}$ with optimum medians $\mathbf{c}_1,\ldots,\mathbf{c}_k$ of $X_1,\ldots,X_k$, respectively, that is, $\mathsf{Opt}(\mathbf{X},k)=\mathsf{cost}_p(X_1,\ldots,X_k)=\mathsf{cost}_p(X_1,\ldots,X_k,\mathbf{c}_1,\ldots,\mathbf{c}_k)$. Let $\mathbf{x}_{i_h}\in\mathbf{S}_h$ for $h\in\{1,\ldots,t\}$. Consider a t-tuple of $(j_1,\ldots,j_t)$ of distinct indices from $\{1,\ldots,k\}$ such that

$$\mathsf{dist}_p(\mathbf{x}_{i_1},\mathbf{c}_{j_1})+\cdots+\mathsf{dist}_p(\mathbf{x}_{i_t},\mathbf{c}_{j_t})=\min_{(q_1,\ldots,q_t)}\big(\mathsf{dist}_p(\mathbf{x}_{i_1},\mathbf{c}_{q_1})+\cdots+\mathsf{dist}_p(\mathbf{x}_{i_t},\mathbf{c}_{q_t})\big),\quad(6.8)$$

where the minimum in the right part is taken over all t-tuples $(q_1,\ldots,q_t)$ of distinct indices from $\{1,\ldots,k\}$. Denote $\ell=k-t$. Iteratively applying Lemma 17 for $S_1,\ldots,S_t$ and the medians $\mathbf{c}_{j_1},\ldots,\mathbf{c}_{j_t}$, we obtain that there is an equal ℓ-clustering $\{Y_1,\ldots,Y_\ell\}$ of $\mathbf{Y}$ such that

$$\mathsf{cost}_p(S_1,\ldots,S_t,Y_1,\ldots,Y_\ell)\leq\mathsf{cost}_p(X_1,\ldots,X_k)+s\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{j_h}).\quad(6.9)$$

Because the points in each S_i are the same, $\mathsf{cost}_p(S_i)=0$ and, therefore, $\mathsf{cost}_p(S_1,\ldots,S_t,Y_1,\ldots,Y_\ell)=\mathsf{cost}_p(Y_1,\ldots,Y_k)$. Then by (6.9),

$$\mathsf{Opt}(\mathbf{Y},\ell)\leq\mathsf{cost}_p(Y_1,\ldots,Y_k)\leq\mathsf{Opt}(\mathbf{X},k)+s\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{j_h}).\quad(6.10)$$

This implies that to prove the lemma, it is sufficient to show that

$$s\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{j_h})\leq \mathsf{Opt}(\mathbf{X},k). \tag{6.11}$$

To prove (6.11), we consider the following auxiliary clustering problem. Let $\mathbf{Z}=S_1\cup\cdots\cup S_t$ and $s=\frac{n}{k}$. The task of the problem is to find a partition $\{Z_1,\ldots,Z_k\}$ of $\mathbf{Z}$, where some sets may be empty and $|Z_i|\leq s$ for every $i\in\{1,\ldots,k\}$, such that

$$\sum_{i=1}^{t}\sum_{\mathbf{x}_h\in Z_i}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}_i) \tag{6.12}$$

is minimum. In words, we cluster the elements of $\mathbf{Z}$ in the optimum way into clusters of size at most s using the optimum medians $\mathbf{c}_1,\ldots,\mathbf{c}_k$ for the clustering $\{X_1,\ldots,X_k\}$. Denote by $\mathsf{Opt}^*(\mathbf{Z},k)$ the minimum value of (6.12). Because in this problem the task is to cluster a subcollection of points of $\mathbf{X}$ and we relax the cluster size constraints, we have that $\mathsf{Opt}^*(\mathbf{Z},k)\leq\mathsf{Opt}(\mathbf{X},k)$. We show the following claim.

Claim 6.1.2.

$$\mathsf{Opt}^*(\mathbf{Z},k)\geq s\cdot\min_{(q_1,\ldots,q_t)}\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{q_h}),$$

where the minimum is taken over all t-tuples $(q_1,\ldots,q_t)$ of distinct indices from $\{1,\ldots,k\}$.

Proof of Claim 6.1.2. We show that the considered auxiliary clustering problem can be reduced to the Min Cost Flow problem (see, e.g., the textbook of Kleinberg and Tardos [57] for the introduction)[1]. We construct the directed graph G and define the *cost* and *capacity* functions $c(\cdot)$ and $\omega(\cdot)$ on the set of arcs $A(G)$ as follows.

- Construct two vertices a and b that are the *source* and *target* vertices, respectively.
- For every $i\in\{1,\ldots,t\}$, construct a vertex u_i (corresponding to S_i) and an arc (a,u_i) with $\omega(a,u_i)=0$.
- For every $j\in\{1,\ldots,k\}$, construct a vertex v_j (corresponding to Z_j) and and arc (v_j,b) with $\omega(v_j,b)=0$.
- For every $h\in\{1,\ldots,t\}$ and every $j\in\{1,\ldots,k\}$, construct an arc (u_h,v_j) and set $\omega(u_i,v_j)=\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_j)$ (recall that $\mathbf{x}_{i_h}\in S_h$).
- For every arc e of G, set $c(e)=s$, where $s=\frac{n}{k}$.

[1]Equivalently one may use the ILP statement.

Then the volume of a *flow* $f\colon A(G) \to \mathbb{R}_{\geq 0}$ is $v(f) = \sum_{i=1}^{t} f(a,u_i)$ and its *cost* is $\omega(f) = \sum_{a\in A(G)} \omega(a)\cdot f(a)$. Let $f^*(\cdot)$ be a flow of volume st with minimum cost. We claim that $\omega(f^*) = \mathsf{Opt}^*(\mathbf{Z},k)$.

Assume that $\{Z_1,\ldots,Z_k\}$ is a partition of $\mathbf{Z}$ such that $|Z_i| \leq s$ for every $i\in\{1,\ldots,k\}$ and $\mathsf{Opt}^*(\mathbf{Z},k) = \sum_{i=1}^{t}\sum_{\mathbf{x}_h\in Z_i} \mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}_i)$. We define the flow $f(\cdot)$ as follows:

- for every $i\in\{1,\ldots,t\}$, set $f(a,u_i) = s$,
- for every $i\in\{1,\ldots,t\}$ and $j\in\{1,\ldots,k\}$, set $f(u_i,v_j) = |S_i\cap Z_j|$, and
- for every $j\in\{1,\ldots,t\}$, set $f(v_j,b) = |Z_j|$.

It is easy to verify that f is a feasible flow of volume st and $\omega(f) = \sum_{i=1}^{t}\sum_{\mathbf{x}_h\in Z_i} \mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}_i)$. Thus, $\omega(f^*)\leq\omega(f) = \mathsf{Opt}^*(\mathbf{Z},k)$.

For the opposite inequality, consider $f^*(\cdot)$. By a well-known property of flows (see [57]), we can assume that $f^*(\cdot)$ is an integer flow, that is, $f^*(e)$ is a nonnegative integer for every $e\in A(G)$. Since $v(f^*) = st$, we have that $f^*(a,u_i) = s$ for every $i\in\{1,\ldots,t\}$. Then we construct the clustering $\{Z_1,\ldots,Z_k\}$ as follows: for every $i\in\{1,\ldots,t\}$ and $j\in\{1,\ldots,k\}$, we put exactly $f^*(u_i,v_j)$ points of S_i into Z_j. Because $f^*(a,u_i) = s$ for every $i\in\{1,\ldots,t\}$ and $c(v_j,b) = s$ for every $j\in\{1,\ldots,k\}$, we obtain that $\{Z_1,\ldots,Z_k\}$ is a partition of $\mathbf{Z}$ such that $|Z_i|\leq s$ for every $i\in\{1,\ldots,k\}$ and $\sum_{i=1}^{t}\sum_{\mathbf{x}_h\in Z_i}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}_i) = \omega(f^*)$. This implies that $\mathsf{Opt}^*(\mathbf{Z},k)\leq\sum_{i=1}^{t}\sum_{\mathbf{x}_h\in Z_i}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}_i) = \omega(f^*)$.

This proves that $\omega(f^*) = \mathsf{Opt}^*(\mathbf{Z},k)$. Moreover, we can observe that, given an integer flow $f(\cdot)$ with $v(f) = st$, we can construct a feasible clustering $\{Z_1,\ldots,Z_k\}$ of cost $\omega(f)$ such that for every $i\in\{1,\ldots,t\}$ and every $j\in\{1,\ldots,k\}$, $|S_i\cap Z_j| = f(u_i,v_j)$. Recall that the capacities of the arcs of G are the same and are equal to s. Then again exploiting the properties of flows (see [57]), we observe that there is a flow $f^*(\cdot)$ with $v(f^*) = st$ of minimum cost such that *saturated* arcs (that is, arcs e with $f^*(e) = c(e) = s$) compose internally vertex disjoint (a,b)-paths, and the flow on other arcs is zero. This implies, that for the clustering $\{Z_1,\ldots,Z_k\}$ constructed for $f^*(\cdot)$, for every $j\in\{1,\ldots,k\}$, either $Z_j = \emptyset$ or there is $i\in\{1,\ldots,t\}$ such that $Z_j = S_i$. Assume that $j_1,\ldots,j_t$ are distinct indices from $\{1,\ldots,k\}$ such that $Z_{j_h} = S_h$ for $h\in\{1,\ldots,t\}$. Then $\omega(f^*) = \sum_{i=1}^{t}\sum_{\mathbf{x}_h\in Z_i}\mathsf{dist}_p(\mathbf{x}_h,\mathbf{c}_i) = s\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{j_h})$ and

$$\mathsf{Opt}^*(\mathbf{Z},k) = \omega(f^*) = s\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{j_h}) \geq s\cdot\min_{(q_1,\ldots,q_t)}\sum_{h=1}^{t}\mathsf{dist}_p(\mathbf{x}_{i_h},\mathbf{c}_{q_h}),$$

where the minimum is taken over all t-tuples $(q_1,\ldots,q_t)$ of distinct indices from $\{1,\ldots,k\}$. This proves the claim. □

Recall that $\mathsf{Opt}^*(\mathbf{Z},k) \leq \mathsf{Opt}(\mathbf{X},k)$. By the choice of $j_1,\ldots,j_t$ in (6.8) and Claim 6.1.2, we obtain that inequality (6.11) holds. Then by (6.11), we have that $\mathsf{Opt}(\mathbf{Y},k-t) \leq 2\cdot\mathsf{Opt}(\mathbf{X},k)$ as required by the lemma. □

6.1.2 Construction of the Lossy Kernel

Now we are ready to show the result about the approximate kernel that we restate.

Theorem 9. *For every nonnegative integer constant p,* Parameterized ℓ_p-Equal k-Median Clustering *admits a 2-approximate kernel when parameterized by B, where the output collection of points has $\mathcal{O}(B^2)$ points of $\mathbb{Z}^{d'}$ with $d'=\mathcal{O}(B^{p+2})$, where each coordinate of a point takes an absolute value of $\mathcal{O}(B^3)$.*

Proof. Let $(\mathbf{X},k,B)$ be an instance of Parameterized ℓ_p-Equal k-Median Clustering with $\mathbf{X}=\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$, where the points are from $\mathbb{Z}^d$ and n is divisible by k. Recall that a lossy kernel consists of two algorithms. The first algorithm is a polynomial time reduction producing an instance $(\mathbf{X}',k',B')$ of bounded size. The second algorithm is a solution-lifting and for every equal k'-clustering $\{X'_1,\ldots,X_{k'}\}$ of $\mathbf{X}'$, this algorithm produces in a polynomial time an equal k-clustering $\{X_1,\ldots,X_k\}$ of $\mathbf{X}$ such that

$$\frac{\mathsf{cost}_p^B(X_1,\ldots,X_k)}{\mathsf{Opt}(\mathbf{X},k,B)} \leq 2\cdot\frac{\mathsf{cost}_p^{B'}(X'_1,\ldots,X'_{k'})}{\mathsf{Opt}(\mathbf{X}',k',B')}.^{2} \tag{6.13}$$

We separately consider the cases when $\frac{n}{k}\geq 4B+1$ and $\frac{n}{k}\leq 4B$.

Suppose that $\frac{n}{k}\geq 4B+1$. Then we apply the algorithm from Lemma 18. If the algorithm returns the answer that $\mathbf{X}$ does not admit an equal k-clustering of cost at most B, then the reduction algorithm returns a trivial no-instance $(\mathbf{X}',k',B')$ of constant size, that is, an instance such that $\mathbf{X}'$ has no clustering of cost at most B'. For example, we set $\mathbf{X}'=\{(0),(1)\}$, $k'=1$, and $B'=0$. Here and in the further cases when the reduction algorithm returns a trivial no-instance, the solution-lifting algorithm returns an arbitrary equal k-clustering of $\mathbf{X}$. Since $\mathsf{cost}_p^B(X_1,\ldots,X_k)=\mathsf{Opt}(\mathbf{X},k,B)=B+1$, (6.13) holds. Assume that the algorithm from Lemma 18 produced an equal k-clustering $\{X_1,\ldots,X_k\}$ of minimum cost. Then the reduction returns an arbitrary instance of Parameterized ℓ_p-Equal k-Median Clustering of constant size. For example, we

[2]Note that by our simplifying assumption, $\frac{\mathsf{cost}_p^B(X_1,\ldots,X_k)}{\mathsf{Opt}(\mathbf{X},k,B)}=1$ if $\mathsf{Opt}(\mathbf{X},k,B)=\mathsf{cost}_p^B(X_1,\ldots,X_k)=0$ and $\frac{\mathsf{cost}_p^B(X_1,\ldots,X_k)}{\mathsf{Opt}(\mathbf{X},k,B)}=+\infty$ if $\mathsf{Opt}(\mathbf{X},k,B)=0$ and $\mathsf{cost}_p^B(X_1,\ldots,X_k)>0$, and the same assumption is used for $\frac{\mathsf{cost}_p^{B'}(X'_1,\ldots,X'_{k'})}{\mathsf{Opt}(\mathbf{X}',k',B')}$.

can use $\mathbf{X}' = \{(0)\}$, $k' = 1$, and $B' = 0$. The solution-lifting algorithms always returns $\{X_1, \ldots, X_k\}$. Clearly, $\mathsf{cost}_p^B(X_1, \ldots, X_k) = \mathsf{Opt}(\mathbf{X}, k, B)$ and (6.13) is fulfilled.

From now on, we assume that $\frac{n}{k} \leq 4B$, that is, $n \leq 4Bk$. We apply the algorithm from Lemma 19. If this algorithm reports that there is no equal k-clustering of cost at most B, then the reduction algorithm returns a trivial no-instance and the solution-lifting algorithm outputs an arbitrary equal k-clustering of $\mathbf{X}$. Clearly, (6.13) is satisfied. Assume that this is not the case. Then we obtain a collection of $n \leq 4Bk$ points $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ of $\mathbb{Z}^{d'}$ satisfying conditions (i)–(iii) of Lemma 19. That is,

(i) for every partition $\{I_1, \ldots, I_k\}$ of $\{1, \ldots, n\}$ such that $|I_1| = \cdots = |I_k| = \frac{n}{k}$, either $\mathsf{cost}_p(X_1, \ldots, X_k) > B$ and $\mathsf{cost}_p(Y_1, \ldots, Y_k) > B$ or $\mathsf{cost}_p(X_1, \ldots, X_k) = \mathsf{cost}_p(Y_1, \ldots, Y_k)$, where $X_i = \{\mathbf{x}_h \mid h \in I_i\}$ and $Y_i = \{\mathbf{y}_h \mid h \in I_i\}$ for every $i \in \{1, \ldots, k\}$,

(ii) $d' = \mathcal{O}(kB^{p+1})$, and

(iii) $|\mathbf{y}_i[h]| = \mathcal{O}(kB^2)$ for $h \in \{1, \ldots, d'\}$ and $i \in \{1, \ldots, n\}$.

By (i), for given an equal k-clustering clustering $\{Y_1, \ldots, Y_k\}$ of $\mathbf{Y}$, we can compute the corresponding clustering $\{X_1, \ldots, X_k\}$ by setting $X_i = \{\mathbf{x}_h \mid \mathbf{y}_h \in Y_i\}$ for $i \in \{1, \ldots, k\}$. Then $\mathsf{Opt}(\mathbf{X}, k, B) = \mathsf{Opt}(\mathbf{Y}, k, B)$ and

$$\frac{\mathsf{cost}_p^B(X_1, \ldots, X_k)}{\mathsf{Opt}(\mathbf{X}, k, B)} = \frac{\mathsf{cost}_p^B(Y_1, \ldots, Y_k)}{\mathsf{Opt}(\mathbf{Y}, k, B)}. \tag{6.14}$$

Hence the instances $(\mathbf{X}, k, B)$ and $(\mathbf{Y}, k, B)$ are equivalent. We continue with the compressed instance $(\mathbf{Y}, k, B)$.

Now we apply the greedy procedure that constructs clusters $S_1, \ldots, S_t$ composed by equal points. Formally, we initially set $\mathbf{X}' := Y$, $k' := k$, and $i := 0$. Then we do the following:

- while $\mathbf{X}'$ contains a collections S of s identical points, set $i := i + 1$, $S_i := S$, $\mathbf{X}' := \mathbf{X}' \setminus S$, and $k' := k' - 1$.

Denote by $\mathbf{X}'$ the set of points obtained by the application of the procedure and let $S_1, \ldots, S_t$ be the collections of equal points constructed by the procedure. Note that $k' = k - t$. We also define $B' = 2B$. Notice that it may happen that $\mathbf{X}' = \mathbf{Y}$ or $\mathbf{X}' = \emptyset$. The crucial property exploited by the kernelization is that by Lemma 20, $\mathsf{Opt}(\mathbf{X}', k') \leq 2 \cdot \mathsf{Opt}(\mathbf{Y}, k)$.

We argue that if $k' > B$, then we have no k-clustering of cost at most B. Suppose that $k' > B'$. Consider an arbitrary equal k'-clustering $\{X'_1, \ldots, X'_{k'}\}$ of $\mathbf{X}'$. Because the construction of $S_1, \ldots, S_t$ stops when there is no collection of s equal points, each cluster X'_i contains at least two distinct points. Since all points have integer coordinates, we have that $\mathsf{cost}_p(X'_i) \geq 1$ for every $i \in \{1, \ldots, k'\}$. Therefore, $\mathsf{cost}_p(X'_1, \ldots, X'_{k'}) = \sum_{i=1}^{k'} \mathsf{cost}_p(X'_i) \geq k' > B' = 2B$. This means that $2 \cdot \mathsf{Opt}(Y, k) \geq \mathsf{Opt}(\mathbf{X}', k') > 2B$ and $\mathsf{Opt}(\mathbf{Y}, k) > B$. Using this, our reduction algorithm returns a trivial no-instance. Then the solution-lifting algorithm outputs an arbitrary equal k-clustering of $\mathbf{X}$ and this satisfies (6.13).

From now on we assume that $k' \leq B' = 2B$ and construct the reduction and solution lifting algorithms for this case.

If $k' = 0$, then $\mathbf{X}' = \emptyset$ and the reduction algorithm simply returns an arbitrary instance of constant size. Otherwise, our reduction algorithms returns $(\mathbf{X}', k', B')$. Observe that since $k' \leq B' = 2B$, $|\mathbf{X}'| \leq n \leq 4B^2$. Recall that $d' = \mathcal{O}(B^{p+2})$ and $|\mathbf{x}'_i[h]| = \mathcal{O}(B^3)$ for $h \in \{1, \ldots, d'\}$ for every point $\mathbf{x}'_i \in \mathbf{X}'$. We conclude that the instance $(\mathbf{X}', k', B')$ of Parameterized ℓ_p-Equal k-Median Clustering satisfies the size conditions of the theorem.

Now we describe the solution-lifting algorithm and argue that inequality (6.13) holds.

If $k' = 0$, then the solution-lifting algorithm ignores the output of the reduction algorithm which was arbitrary. It takes the equal k-clustering $\{S_1, \ldots, S_k\}$ of $\mathbf{Y}$ and outputs the equal k-clustering $\{X_1, \ldots, X_k\}$ of $\mathbf{X}$ by setting $X_i = \{\mathbf{x}_h \mid \mathbf{y}_h \in S_i\}$ for $i \in \{1, \ldots, k\}$. Clearly, $\mathsf{cost}_p(S_1, \ldots, S_k) = \mathsf{cost}_p(X_1, \ldots, X_p) = 0$. Therefore, (6.13) holds.

If $k' > 0$, we consider an equal k'-clustering $\{X'_1, \ldots, X'_{k'}\}$ of $\mathbf{X}'$. The solution-lifting algorithm constructs an equal k-clustering $\{S_1, \ldots, S_t, X'_1, \ldots, X'_{k'}\}$, that is, we just add the clusters constructed by our greedy procedure. Since the points in each set S_i are the same, $\mathsf{cost}_p(S_i) = 0$ for every $i \in \{1, \ldots, t\}$. Therefore,

$$\mathsf{cost}_p(S_1, \ldots, S_t, X'_1, \ldots, X'_{k'}) = \mathsf{cost}_p(X'_1, \ldots, X'_{k'}).$$

Notice that since $\mathsf{Opt}(\mathbf{X}', k') \leq 2 \cdot \mathsf{Opt}(\mathbf{Y}, k)$, we have that $\mathsf{Opt}(\mathbf{X}', k', B') \leq 2 \cdot \mathsf{Opt}(\mathbf{Y}, k, B)$. Indeed, if $\mathsf{Opt}(\mathbf{Y}, k) \leq B$, then $\mathsf{Opt}(\mathbf{X}', k') \leq 2B = B'$. Hence, $\mathsf{Opt}(\mathbf{Y}, k, B) = \mathsf{Opt}(\mathbf{Y}, k)$, $\mathsf{Opt}(\mathbf{X}', k', B') = \mathsf{Opt}(\mathbf{X}', k')$, and $\mathsf{Opt}(\mathbf{X}', k', B') \leq 2 \cdot \mathsf{Opt}(\mathbf{Y}, k, B)$. If $\mathsf{Opt}(\mathbf{Y}, k) > B$, then $\mathsf{Opt}(\mathbf{Y}, k, B) = B + 1$. In this case, $2 \cdot \mathsf{Opt}(\mathbf{Y}, k, B) = 2B + 2 > \mathsf{Opt}(\mathbf{X}', k', B')$ because $\mathsf{Opt}(\mathbf{X}', k', B') \leq B' + 1 = 2B + 1$. Finally, since $\mathsf{cost}_p(S_1, \ldots, S_t, X'_1, \ldots, X'_{k'}) = \mathsf{cost}_p(X'_1, \ldots, X'_{k'})$ and $\mathsf{Opt}(\mathbf{X}', k', B') \leq$

$2 \cdot \mathsf{Opt}(\mathbf{Y}, k, B)$, we conclude that

$$\frac{\mathsf{cost}_p^B(S_1, \ldots, S_t, X'_1, \ldots, X'_{k'})}{\mathsf{Opt}(\mathbf{Y}, k, B)} \leq 2 \cdot \frac{\mathsf{cost}_p^B(X_1, \ldots, X'_{k'})}{\mathsf{Opt}(\mathbf{X}', k', B')}. \tag{6.15}$$

Then the solution-lifting algorithm computes the equal k-clustering $\{X_1, \ldots, X_k\}$ for the equal k-clustering $\{Y_1, \ldots, Y_k\} = \{S_1, \ldots, S_t, X'_1, \ldots, X'_{k'}\}$ of $\mathbf{Y}$ by setting $X_i = \{\mathbf{x}_h \mid \mathbf{y}_h \in Y_i\}$ for $i \in \{1, \ldots, k\}$. Combining (6.14) and (6.15), we obtain (6.13).

This concludes the description of the reduction and solution-lifting algorithms, as well as the proof of their correctness. To argue that the reduction algorithm is a polynomial-time algorithm, we observe that the algorithms from Lemmata 18 and 19 run in a polynomial time. Trivially, the greedy construction of $S_1, \ldots, S_t$, $\mathbf{X}$, and k' can be done in a polynomial time. Therefore, the reduction algorithm runs in a polynomial time. The solution-lifting algorithm is also easily implementable to run in a polynomial time. □

6.2 Kernelization

In this section, we study (exact) kernelization of clustering with equal sizes. In Subsection 6.2.1 we prove Theorem 10 claiming that decision version of the problem, Decision ℓ_p-Equal k-Median Clustering, does not admit a polynomial kernel being parameterized by B only. We also show in Subsection 6.2.2 that the technical lemmata developed in the previous section for approximate kernel, can be used to prove that Decision ℓ_p-Equal k-Median Clustering parameterized by k and B admits a polynomial kernel.

6.2.1 Kernelization Lower Bound

In this subsection, we show that it is unlikely that Decision ℓ_p-Equal k-Median Clustering admits a polynomial kernel when parameterized by B only. We prove this for the ℓ_0 and ℓ_1-norms. Our lower bound holds even for points with binary coordinates, that is, for points from $\{0, 1\}^d$. For this, we use the result of Dell and Marx [30] about kernelization lower bounds for the Perfect r-Set Matching problem.

A hypergraph $\mathcal{H}$ is said to be *r-uniform* for a positive integer r, if every hyperedge of $\mathcal{H}$ has size r. Similarly to graphs, a set of hyperedges M is a *matching* if the hyperedges in M are pairwise disjoint, and M is *perfect* if every vertex of $\mathcal{H}$ is *saturated* in M, that is, included in one of the hyperedges of M. Perfect r-Set Matching asks, given a r-uniform hypergraph $\mathcal{H}$, whether $\mathcal{H}$ has a perfect matching. Dell and Marx [30] proved

the following kernelization lower bound. We use the Proposition 2 and Corollary 1 that we restate here.

Proposition 2. *[[30]] Let $r \geq 3$ be an integer and let ε be a positive real. If* NP $\subseteq$ coNP / poly, *then* Perfect r-Set Matching *does not have kernels with $\mathcal{O}(\left(\frac{|V(\mathcal{H})|}{r}\right)^{r-\varepsilon})$ hyperedges.*

We need a weaker claim.

Corollary 1. Perfect r-Set Matching *admits no polynomial kernel when parameterized by the number of vertices of the input hypergraph unless* NP $\subseteq$ coNP / poly.

Proof. To see the claim, it is sufficient to observe that the existence of a polynomial kernel for Perfect r-Set Matching parameterized by $|V(\mathcal{H})|$ implies that the problem has a kernel such that the number of hyperedges is polynomial in $|V(\mathcal{H})|$ with the degree of the polynomial that does not depend on r contradicting Proposition 2. □

We show the kernelization lower bound for ℓ_0 and ℓ_1 using the fact that optimum medians can be computed by the *majority rule* for a collection of binary points. Let X be a collection of points of $\{0,1\}^d$. We construct $\mathbf{c} \in \{0,1\}^d$ as follows: for $i \in \{1,\ldots,d\}$, consider the multiset $S_i = \{\mathbf{x}[i] \mid \mathbf{x} \in X\}$ and set $\mathbf{c}[i] = 0$ if at least half of the elements of S_i are zeros, and set $\mathbf{c}[i] = 1$ otherwise. It is straightforward to observe the following.

Observation 8. *Let X be a collection of points of $\{0,1\}^d$ and let $\mathbf{c} \in \{0,1\}^d$ be a vector constructed by the majority rule. Then for the ℓ_0 and ℓ_1-norms, $\mathbf{c}$ is an optimum median for $\mathbf{X}$.*

We also use the following lemma which is a special case of Lemma 10 in Chapter 5.

Lemma 21. *Let $\{X_1,\ldots,X_k\}$ be an equal k-clustering of a collection of points $\mathbf{X} = \{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ from $\{0,1\}^d$, and let $\mathbf{c}_1,\ldots,\mathbf{c}_k$ be optimum medians for $X_1,\ldots,X_k$, respectively. Let also $\mathbf{C} \subseteq \{\mathbf{c}_1,\ldots,\mathbf{c}_k\}$ be the set of medians coinciding with some points of $\mathbf{X}$. Suppose that every collection of the same points of $\mathbf{X}$ has size at most $\frac{n}{k}$. Then there is an equal k-clustering $\{X'_1,\ldots,X'_k\}$ of $\mathbf{X}$ such that $\mathsf{cost}_0(X'_1,\ldots,X'_k,\mathbf{c}_1,\ldots,\mathbf{c}_k) \leq \mathsf{cost}_0(X_1,\ldots,X_k,\mathbf{c}_1,\ldots,\mathbf{c}_k)$ and for every $i \in \{1,\ldots,k\}$, the following is fulfilled: if $\mathbf{c}_i \in \mathbf{C}$, then each $\mathbf{x}_h \in \mathbf{X}$ coinciding with $\mathbf{c}_i$ is in X'_i.*

Now we are ready to prove Theorem 10, we restate it here.

Theorem 10. *For the ℓ_0 and ℓ_1-norms,* Decision ℓ_p-Equal k-Median Clustering *has no polynomial kernel when parameterized by B unless* NP $\subseteq$ coNP / poly, *even if the input points are binary, that is, are from* $\{0,1\}^d$.

Proof. Notice that for any binary vector $\mathbf{x} \in \{0,1\}^d$, $\mathsf{dist}_0(\mathbf{x}) = \mathsf{dist}_1(\mathbf{x})$. Since we consider only instances where the input points are binary, we can assume that the medians of clusters are binary as well by Observation 8. Then it is sufficient to prove the theorem for one norm, say ℓ_0. We reduce from Perfect r-Set Matching. Let $\mathcal{H}$ be an r-uniform hypergraph. Denote by $v_1, \ldots, v_n$ the vertices and by $E_1, \ldots, E_m$ the hyperedges of $\mathcal{H}$, respectively. We assume that n is divisible by r, as otherwise $\mathcal{H}$ has no perfect matching. We also assume that $r \geq 3$ because for $r \leq 2$, Perfect r-Set Matching can be solved in a polynomial time [66].

We construct the instance $(\mathbf{X}, k, B)$ of Decision ℓ_p-Equal k-Median Clustering, where $\mathbf{X}$ is a collection of $(r-1)n + rm$ points of $\{0,1\}^d$, where $d = 2rn$.

To describe the construction of $\mathbf{X}$, we partition the set $\{1, \ldots, 2rn\}$ of coordinate indices into n blocks $R_1, \ldots, R_n$ of size $2r$ each. For every $i \in \{1, \ldots, n\}$, we select an index $p_i \in R_i$ and set $R'_i = R_i \setminus \{p_i\}$. Formally,

- $R_i = \{2r(i-1)+1, \ldots, 2ri\}$ for $i \in \{1, \ldots, n\}$,
- $p_i = 2r(i-1)+1$ for $i \in \{1, \ldots, n\}$, and
- $R'_i = \{2r(i-1)+2, \ldots, 2ri\}$ for $i \in \{1, \ldots, n\}$.

The set of points $\mathbf{X}$ consists of $n+m$ blocks of equal points $V_1, \ldots, V_n$ and $F_1, \ldots, F_m$, where $|V_i| = r-1$ for each $i \in \{1, \ldots, n\}$ and $|F_i| = r$ for $i \in \{1, \ldots, m\}$. Each block V_i is used to encode the vertex v_i, and each block F_i is used to encode the corresponding hyperedge E_i. An example is shown in Figure 6.1.

For each $i \in \{1, \ldots, n\}$, we define the vector $\mathbf{v}_i \in \{0,1\}^{2rn}$ corresponding to the vertex v_i of $\mathcal{H}$:

$$\mathbf{v}_i[j] = \begin{cases} 1 & \text{if } j \in R_i, \\ 0 & \text{otherwise.} \end{cases}$$

Then V_i consists of $r-1$ copies of $\mathbf{v}_i$ that we denote $\mathbf{v}_i^{(1)}, \ldots, \mathbf{v}_i^{(r-1)}$.

For every $j \in \{1, \ldots, m\}$, we define the vector $\mathbf{f}_j \in \{0,1\}^{2rn}$ corresponding to the

hyperedge $E_j = \{v_{i_1^{(j)}}, \ldots, v_{i_r^{(j)}}\}$:

$$\mathbf{f}_j[h] = \begin{cases} 1 & \text{if } h = p_s \text{ for some } s \in \{i_1^{(j)}, \ldots, i_r^{(j)}\}, \\ 0 & \text{otherwise.} \end{cases}$$

Then F_j includes r copies of $\mathbf{f}_j$ denoted by $\mathbf{f}_j^{(1)}, \ldots, \mathbf{f}_j^{(r)}$.

To complete the construction of the instance of Decision ℓ_p-Equal k-Median Clustering, we define

- $k = n + m - \frac{n}{r}$,
- $B = (3r - 2)n$.

Recall that n is divisible by r and note that $\frac{(r-1)n+rm}{k} = r$.

It is straightforward to verify that the construction of $(\mathbf{X}, k, B)$ is polynomial.

We claim that the hypergraph $\mathcal{H}$ has a perfect matching if and only if $(\mathbf{X}, k, B)$ is a yes-instance of Decision ℓ_p-Equal k-Median Clustering. The proof uses the following property of the points of $\mathbf{X}$: for every $i \in \{1, \ldots, n\}$ and every $j \in \{1, \ldots, m\}$,

$$\mathsf{dist}_0(\mathbf{v}_i, \mathbf{f}_j) = \begin{cases} 3r - 2 & \text{if } v_i \in E_j, \\ 3r & \text{if } v_i \notin E_j. \end{cases} \tag{6.16}$$

For the forward direction, assume that $\mathcal{H}$ has a perfect matching M. Assume without loss of generality that $M = \{E_1, \ldots, E_s\}$ for $s = \frac{n}{r}$. Since M is a prefect matching, for every $i \in \{1, \ldots, n\}$, there is a unique $h_i \in \{1, \ldots, s\}$ such that $v_i \in E_{h_i}$. We construct the equal k-clustering $\{X_1, \ldots, X_k\}$ as follows.

For every $i \in \{1, \ldots, n\}$, we define $X_i = V_i \cup \{\mathbf{f}_{h_i}^{(t)}\}$, where t is chosen from the set $\{1, \ldots, r\}$ in such a way that $X_1, \ldots, X_n$ are disjoint. In words, we initiate each cluster X_i by setting $X_i := V_i$ for $i \in \{1, \ldots, n\}$. This way, we obtain n clusters of size $r - 1$ each. Then we consider the blocks of points $F_1, \ldots, F_s$ corresponding to the hyperedges of M and split them between the clusters $X_1, \ldots, X_n$ by including a single element into each cluster. It is crucial that each $X_i = V_i$ is complemented by an element of F_{h_i}, that is, by an element of the initial cluster corresponding to the hyperedge saturating the vertex v_i. Since M is a perfect matching, this splitting is feasible.

Notice that the first s blocks of points $F_1, \ldots, F_s$ are split between $X_1, \ldots, X_n$. The

$$\mathbf{X} = \left(\begin{array}{cc|cc|cc|cc|cc|cc||ccc|ccc|ccc|ccc}
1&1&0&0&0&0&0&0&0&0&0&0&1&1&1&0&0&0&1&1&1&0&0&0\\
\hline
1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
\hline
0&0&1&1&0&0&0&0&0&0&0&0&1&1&1&0&0&0&0&0&0&1&1&1\\
\hline
0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
\hline
0&0&0&0&1&1&0&0&0&0&0&0&1&1&1&0&0&0&1&1&1&0&0&0\\
\hline
0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
\hline
0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&1&1&1&0&0&0&1&1&1\\
\hline
0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
\hline
0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&1&1&1&1&1&1&0&0&0\\
\hline
0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
\hline
0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&1&1&1&0&0&0&1&1&1\\
\hline
0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0
\end{array}\right)$$

Figure 6.1: The construction of $\mathbf{X}$ for $\mathcal{H}$ with $V(\mathcal{H}) = \{v_1, \ldots, v_6\}$ and the hyperedges $E_1 = \{v_1, v_2, v_3\}$, $E_2 = \{v_4, v_5, v_6\}$, $E_3 = \{v_1, v_3, v_5\}$, and $E_4 = \{v_2, v_4, v_5\}$. The collection of the points $\mathbf{X}$ is shown here as a matrix, where each column is a point of $\mathbf{X}$. Note that $r = 3$ here. The blocks of $\mathbf{X}$ are shown by solid lines and the part of $\mathbf{X}$ corresponding to the vertices of $\mathcal{H}$ is separated from the part corresponding to hyperedges by a double line. The blocks of coordinates with indices $R_1, \ldots, R_6$ are separated by solid lines. The coordinates with the indices $p_1 = 1$, $p_2 = 7$, $p_3 = 13$, $p_4 = 19$, $p_5 = 25$, and $p_6 = 31$ are underlined by dashed lines.

remaining $m-s$ blocks $F_{s+1},\ldots,F_m$ have size r each and form clusters $X_{n+1},\ldots,X_k$. This completes the construction of $\{X_1,\ldots,X_k\}$.

To evaluate $\mathsf{cost}_0(X_1,\ldots,X_k)$, notice that the optimal median $\mathbf{c}_i=\mathbf{v}_i$ for $i\in\{1,\ldots,n\}$ by the majority rule. Then, by (6.16), $\mathsf{cost}_0(X_i)=\mathsf{dist}_0(\mathbf{v}_i,\mathbf{f}_{h_i})=3r-2$. Since the clusters $X_{n+1},\ldots,X_r$ consist of equal points, we have that $\mathsf{cost}_0(X_i)=0$ for $i\in\{1,\ldots,m-s\}$. Then $\mathsf{cost}(X_1,\ldots,X_k)=(3r-2)n=B$. Therefore, $(\mathbf{X},k,B)$ is a yes-instance of Decision ℓ_p-Equal k-Median Clustering.

For the opposite direction, let $\{X_1,\ldots,X_k\}$ be an equal k-clustering of $\mathbf{X}$ of cost at most B. Denote by $\mathbf{c}_1,\ldots,\mathbf{c}_k$ the optimal medians constructed by the majority rule. Observe that the choice of a median by the majority rule described above is not symmetric because if i-th coordinates of the points in a cluster have the same number of zeros and ones, the rule selects the zero value for the i-coordinate of the median. We show the following claim.

Claim 6.2.1. *For every $i\in\{1,\ldots,k\}$, either $\mathbf{c}_i\in\{\mathbf{v}_1,\ldots,\mathbf{v}_n\}$ or $\mathbf{c}_i[j]=0$ for all $j\in R'_1\cup\ldots\cup R'_n$. Moreover, the medians of the first type, that is, coinciding with one of $\mathbf{v}_1,\ldots,\mathbf{v}_n$, are distinct.*

Proof of Claim 6.2.1. Suppose that $\mathbf{c}_i[h]\neq 0$ for some $h\in R'_j$, where $j\in\{1,\ldots,n\}$. Observe that, by the construction of $\mathbf{X}$, for every point $\mathbf{x}\in\mathbf{X}$, $\mathbf{x}[h]=1$ only if $\mathbf{x}\in V_j$. Since $\mathbf{c}_i$ is constructed by the majority rule, we obtain that more than half of elements of X_i are from V_j and $\mathbf{c}_i=\mathbf{v}_j$. To see the second part of the claim, notice that $|V_j|=r-1$ and, therefore, at most one cluster X_i of size r can have at least half of its elements from V_j. □

By Claim 6.2.1, we assume without loss of generality that $\mathbf{c}_i=\mathbf{v}_i$ for $i\in\{1,\ldots,\ell\}$ for some $\ell\in\{0,\ldots,r\}$ ($\ell=0$ if there is no cluster with the median from $\{\mathbf{v}_1,\ldots,\mathbf{v}_n\}$) and $\mathbf{c}_i[j]=0$ for $j\in R'_1\cup\ldots\cup R'_n$ whenever $i\in\{\ell+1,\ldots,k\}$. Because the medians $\mathbf{c}_1,\ldots,\mathbf{c}_\ell$ are equal to points of $\mathbf{X}$, by Lemma 21, we can assume that $V_i\subset X_i$ for $i\in\{1,\ldots,\ell\}$.

Claim 6.2.2. *$\ell=n$.*

Proof of Claim 6.2.2. The proof is by contradiction. Assume that $\ell<n$. Consider the elements of $n-\ell$ blocks $V_{\ell+1},\ldots,V_n$. Let p be the number of elements of $V_{\ell+1}\cup\ldots\cup V_n$ included in $X_1,\ldots,X_\ell$ and the remaining $q=(r-1)(n-\ell)-p$ elements are in $X_{\ell+1},\ldots,X_k$. By the definition of $\mathbf{v}_1,\ldots,\mathbf{v}_n$, if a point $\mathbf{v}_h^{(t)}\in V_h$ for some $h\in\{\ell+1,\ldots,n\}$ is in X_i for some $i\in\{1,\ldots,\ell\}$, then $\mathsf{dist}_0(\mathbf{v}_h^{(t)},\mathbf{c}_i)=\mathsf{dist}_0(\mathbf{v}_h,\mathbf{v}_i)_0=4r$. Also we have that if $\mathbf{v}_h^{(t)}\in V_h$ for some $h\in\{\ell+1,\ldots,n\}$ is in X_i for some $i\in\{\ell+1,\ldots,r\}$, then $\mathsf{dist}_0(\mathbf{v}_h^{(t)},\mathbf{c}_I)=\mathsf{dist}_0(\mathbf{v}_h,\mathbf{c}_I)\geq|R'_h|=2r-1$. By (6.16), if the unique point $X_i\setminus V_i$

is $\mathbf{f}_h^{(t)} \in F_h$ for some $h \in \{1,\ldots,m\}$, then $\mathsf{dist}_0(\mathbf{f}_h^{(t)},\mathbf{c}_i) = \mathsf{dist}_0(\mathbf{f}_h,\mathbf{v}_i) \geq 3r-2$. Then $\sum_{i=1}^{\ell}\mathsf{cost}_0(X_i) \geq 4rp+(3r-2)(\ell-p)$ and $\sum_{i=\ell+1}^{k}\mathsf{cost}_0(X_i) \geq (2r-1)q$. Recall also that $r \geq 3$ and, therefore, $r+2 \leq 2r-1$ and $(r+2)(r-1) > 3r-2$. Summarizing, we obtain that

$$\begin{aligned}\mathsf{cost}_0(X_1,\ldots,X_k) &= \sum_{i=1}^{\ell}\mathsf{cost}_0(X_i)+\sum_{i=\ell+1}^{k}\mathsf{cost}_0(X_i)\\ &\geq \big(4rp+(3r-2)(\ell-p)\big)+\big((2r-1)q\big)\\ &= (3r-2)\ell+(r+2)p+(2r-1)q\\ &\geq (3r-2)\ell+(r+2)(p+q) = (3r-2)\ell+(r+2)(r-1)(n-\ell)\\ &> (3r-2)n = B,\end{aligned}$$

but this contradicts that $\mathsf{cost}_0(X_1,\ldots,X_k) \leq B$. This proves the claim. □

By Claim 6.2.2, we obtain that $\mathbf{c}_i = \mathbf{v}_i$ and $X_i \subset V_i$ for $i \in \{1,\ldots,n\}$. For every $i \in \{1,\ldots,n\}$, $X_i \setminus V_i$ contains a unique point. Clearly, this is a point from $F_1 \cup \cdots \cup F_m$. Denote by $\mathbf{f}_{h_i}^{(t_i)}$ the point of $X_i \subset V_i$ for $i \in \{1,\ldots,n\}$. By (6.16), $\mathsf{dist}_0(\mathbf{c}_i,\mathbf{f}_{h_i}^{(t_i)}) = \mathsf{dist}_0(\mathbf{c}_i,\mathbf{f}_{h_i}) \geq 3r-2$ for every $i \in \{1,\ldots,n\}$. This means that

$$\begin{aligned}B \geq& \mathsf{cost}_0(X_1,\ldots,X_k) = \sum_{i=1}^{n}\mathsf{cost}_0(X_i)+\sum_{i=n+1}^{k}\mathsf{cost}_0(X_i) \geq \sum_{i=1}^{n}\mathsf{cost}_0(X_i)\\ \geq& (3d-2)n = B.\end{aligned}$$

Therefore, $\sum_{i=n+1}^{k}\mathsf{cost}_0(X_i) = 0$. Hence, $k-n = m-s$ clusters $X_{n+1},\ldots,X_k \subseteq F_1 \cup \cdots \cup F_m$, where $s = \frac{n}{r}$, consists of equal points. Without loss of generality, we assume that $F_{s+1},\ldots,F_m$ form these clusters. Then the elements of $F_1,\ldots,F_s$ are split to complement $V_1,\ldots,V_n$ to form $X_1,\ldots,X_n$. In particular, for every $i \in \{1,\ldots,n\}$, there is $\mathbf{f}_{h_i}^{(t_i)} \in X_i$ for some $h_i \in \{1,\ldots,m\}$ and $t_i \in \{1,\ldots,r\}$.

We claim that $M = \{E_1,\ldots,E_s\}$ is a perfect matching of $\mathcal{H}$. To show this, consider a vertex $v_i \in V(\mathcal{H})$. We prove that $v_i \in E_{h_i}$. For sake of contradiction, assume that $v_i \notin E_{h_i}$. Then $\mathsf{dist}_0(\mathbf{f}_{h_i}^{(t_i)},\mathbf{c}_i) = \mathsf{dist}_0(\mathbf{f}_{h_i},\mathbf{v}_i) = 3r$ by (6.16) and

$$\begin{aligned}\mathsf{cost}_0(X_1,\ldots,X_k) &= \sum_{j=1}^{n}\mathsf{cost}_0(X_j) \geq \sum_{j=1}^{n}\mathsf{dist}_0(\mathbf{f}_{h_j}^{(t_j)},\mathbf{c}_i)\\ &= \sum_{j=1}^{n}\mathsf{dist}_0(\mathbf{f}_{h_i},\mathbf{v}_i) \ \geq (3r-2)n+2 > B;\end{aligned}$$

a contradiction with $\mathsf{cost}_0(X_1,\ldots,X_k) \leq B$. Hence, every vertex of $V(\mathcal{H})$ is saturated

by some hyperedge of M. Since $|M| = s = \frac{n}{r}$, we have that the hyperedges of M are pairwise disjoint, that is, M is a matching. Since every vertex is saturated and M is a matching, M is a perfect matching.

This concludes the proof of our claim that $\mathcal{H}$ has a perfect matching if and only if $(\mathbf{X}, k, B)$ is a yes-instance of Decision ℓ_p-Equal k-Median Clustering.

Observe that $B = (3r-2)n$ in the reduction meaning that $B = \mathcal{O}(n^2)$. Since Decision ℓ_p-Equal k-Median Clustering is in NP, there is a polynomial reduction form Decision ℓ_p-Equal k-Median Clustering to Perfect r-Set Matching. Thus, if Decision ℓ_p-Equal k-Median Clustering has a polynomial kernel when parameterized by B, then Perfect r-Set Matching has a polynomial kernel when parameterized by the number of vertices of the input hypergraph. This leads to a contradiction with Corollary 1 and completes the proof of the theorem. □

6.2.2 Polynomial Kernel for $k + B$ Parameterization

In this subsection, we prove Theorem 11 that we restate here.

Theorem 11. *For every nonnegative integer constant p,* Decision ℓ_p-Equal k-Median Clustering *admits a polynomial kernel when parameterized by k and B, where the output collection of points has $\mathcal{O}(kB)$ points of $\mathbb{Z}^{d'}$ with $d' = \mathcal{O}(kB^{p+1})$ and each coordinate of a point takes an absolute value of $\mathcal{O}(kB^2)$.*

Proof. Let $(\mathbf{X}, k, B)$ be an instance of Decision ℓ_p-Equal k-Median Clustering with $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, where the points are from $\mathbb{Z}^d$. Recall that n is divisible by k.

Suppose $\frac{n}{k} \geq 4B+1$. Then we can apply the algorithm from Lemma 18. If the algorithm returns that there is no equal k-clustering of cost at most B, then the kernelization algorithm returns a trivial no-instance of Decision ℓ_p-Equal k-Median Clustering. Otherwise, if $\mathsf{Opt}(X, k) \leq B$, then the algorithm returns a trivial yes-instance.

Assume from now that $\frac{n}{k} \leq 4B$, that is, $n \leq 4Bk$. Then we apply the algorithm from Lemma 19. If this algorithm reports that there is no equal k-clustering of cost at most B, then the kernelization algorithm returns a trivial no-instance of Decision ℓ_p-Equal k-Median Clustering. Otherwise, the algorithm from Lemma 19 returns a collection of $n \leq 4Bk$ points $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ of $\mathbb{Z}^{d'}$ satisfying conditions (i)–(iii) of the lemma. By (i), we obtain that the instances $(\mathbf{X}, k, B)$ and $(\mathbf{Y}, k, B)$ of Decision ℓ_p-Equal k-Median Clustering are equivalent. By (ii), we have that the dimension

$d' = \mathcal{O}(k(B^{p+1}))$, and by (iii), each coordinate of a point takes an absolute value of $\mathcal{O}(kB^2)$. Thus, $(\mathbf{Y}, k, B)$ is a required kernel. □

6.3 APX-Hardness of ℓ_p-Equal k-Median Clustering

In this section, we prove APX-hardness of ℓ_p-Equal k-Median Clustering w.r.t. Hamming (ℓ_0) and ℓ_1 distances. The constructed hard instances consist of high-dimensional binary (0/1) points. As the ℓ_0 and ℓ_1 distances between any two binary points are the same, we focus on the case of ℓ_0 distances. Our reduction is from 3-Dimensional Matching (3DM), where we are given three disjoint sets of elements X, Y and Z such that $|X| = |Y| = |Z| = n$ and a set of m triples $T \subseteq X \times Y \times Z$. In addition, each element of $W := X \cup Y \cup Z$ appears in at most 3 triples. A set $M \subseteq T$ is called a matching if no element of W is contained in more than one triple of M. The goal is to find a maximum cardinality matching. We need Proposition 1 which we restate here.

Proposition 1. *[Restatement of Theorem 4.4 from [73]] There exists a constant $0 < \gamma < 1$, such that it is NP-hard to distinguish the instances of the 3DM problem in which a perfect matching exists, from the instances in which there is a matching of size at most $(1-\gamma)n$.*

Here γ should be seen as a very small constant close to 0. We use the construction described in Section 6.2.1, with a small modification.

We are given an instance of 3DM. Let $N = 3n$, the total number of elements. We construct a binary matrix A of dimension $6N \times (2N + 3m)$. For each element, we take 2 columns and for each triple 3 columns. The $6N$ row indexes are partitioned into N parts each of size 6. In particular, let $R_1 = \{1, \ldots, 6\}$, $R_2 = \{7, \ldots, 12\}$ and so on. For the i-th element, we construct the column a_i of length $6N$ which has 1 corresponding to the indexes in R_i and 0 elsewhere.

Recall that each element can appear in at most 3 triples. For each element x, consider any arbitrary ranking of the triples that contain it. The occurrence of x in a triple with rank j is called its j-th occurrence for $1 \leq j \leq 3$. For example, suppose x appears in three triples t_w, t_y and t_z. One can consider the ranking $1.t_w, 2.t_y, 3.t_z$. Then, the occurrence of x in t_y is called 2-nd occurrence. Let v_i^j be the j-th index of R_i for $1 \leq i \leq N, 1 \leq j \leq 3$. For each triple t with j_1-, j_2- and j_3-th occurrences of the elements p, q and r in t, respectively, we construct the column b_t of length $6N$ which has 1 corresponding to the indices $v_p^{j_1}, v_q^{j_2}$ and $v_r^{j_3}$, and 0 elsewhere.

The triple columns are defined in a different way in our reduction in Section 6.2.1 where for each triple and each element, a fixed index is set to 1. Now, we set different indices based on the occurrences of the element. This ensures that for two different triple columns b_s and b_t, their Hamming distance $d_H(b_s, b_t) = 6$. Note that $d_H(a_i, b_t) = 7$ if the element i is in triple t, otherwise $d_H(a_i, b_t) = 9$. Set cluster size to be 3 and the number of clusters k to be $(2N/3) + m$. We will prove the following lemma.

Lemma 22. *If there is a perfect matching, there is a feasible clustering of cost* $7N$. *If all matchings have size at most* $(1 - \gamma)n$, *any feasible clustering has cost at least* $7(1 - \gamma)N + (23/3)\gamma N$.

Note that it is sufficient to prove the above lemma for showing the APX-hardness of the problem. The proof of the first part of the lemma is exactly the same as in the previous construction. We will prove the second part. To give some intuition of the cost suppose there is a matching of the maximum size $(1 - \gamma)n$. Then we can cluster the matched elements and triples in the same way as in the perfect matching case by paying a cost of $7(1 - \gamma)N$. Now for each unmatched element, we put its two columns in a cluster. Now we have γN clusters with one free slot in each. One can fill in these slots by columns corresponding to $\gamma N/3$ unmatched triples. All the remaining unmatched triples form their own cluster. Now, consider an unmatched triple s whose 3 columns are used to fill in slots of unmatched elements p, q, and r. As this triple was not matched, it cannot contain all these three elements, i.e, it can contain at most 2 of these elements. Thus, for at least one element, the cost of the cluster must be 9. Therefore, the total cost of the three clusters corresponding to p, q, and r is at least $7 + 7 + 9 = 23$. The total cost corresponding to all $\gamma N/3$ unmatched triples is then $(23/3)\gamma N$. We will show that one cannot find a feasible clustering of lesser cost.

For our convenience, we will prove the contrapositive of the second part of the above lemma: if there is a feasible clustering of cost less than $7(1-\gamma)N+(23/3)\gamma N$, then there is a matching of size greater than $(1 - \gamma)n$. So, assume that there is such a clustering. Let $c_1, c_2, \ldots, c_k$ be the cluster centers.

By Lemma 21, we can assume that if a column f of A is a center of a cluster C, all the columns equal to f are in C. We will use this in the following. A center c_i is called an element center if c_i is an element column. Suppose the given clustering contains ℓ clusters with element centers for some ℓ. Without loss of generality, we assume that these are the first ℓ clusters.

Lemma 23. *If the cost of the given clustering is less than* $7(1-\gamma)N + (23/3)\gamma N$, $\ell > (1 - 2\gamma/9)N$.

Proof. Note that if a cluster center is an element column, then by Lemma 21 we can assume that both element columns are present in the cluster. Thus, in our case, each of the first ℓ clusters contains two element columns and some other column. Now, each of these ℓ other columns can be either a column of some other element or a triple column. Let ℓ_1 of these be element columns and ℓ_2 of these be triple columns, where $\ell = \ell_1 + \ell_2$. For each cluster corresponding to these ℓ_1 element columns, the cost is 12, as $d_H(a_i, a_j) = 12$ for all i, j. Similarly, for each cluster corresponding to the ℓ_2 triple columns, the cost is at least 7, as $d_H(a_i, b_t) \geq 7$ for all i, t.

Note that out of $2N$ element columns, $2\ell + \ell_1$ are in the first ℓ clusters. The rest of the element columns are in the other clusters. Now there can be two cases: such a column is in a cluster that contains (i) at least 2 element columns and (ii) exactly one element column.

Claim 6.3.1. *The cost of each element column which is not in the first ℓ clusters is at least 5 in the first case.*

Proof. Consider such a column a_i and let c_j be the center of the cluster that contains a_i. Note that the only 1 entries in a_i are corresponding to the indices in R_i. We claim that at most one entry of c_j corresponding to the indices in R_i can be 1. This proves the original claim, as $|R_i| = 6$. Consider an index $z \in R_i$ such that $c_j[z] = 1$. As c_j is not an element column and the centers are defined based on the majority rule, there is a column e in the cluster with $e[z] = 1$. This must be a column of a triple that contains the element i. By construction, e does not contain 1 corresponding to the indices in $R_i \setminus \{z\}$. As the third column in the cluster is another element column (as we are in the first case), its entries corresponding to the indices in R_i are again 0. Hence, by the majority rule, at most one entry of c_j corresponding to the indices in R_i can be 1. □

Next, we consider case (ii).

Claim 6.3.2. *Consider a cluster that is not one of the first ℓ clusters and contains exactly one element column. Then, its cost is at least 5. Moreover, the cost of the element column is at least 4.*

Proof. Consider the element column a_i of the cluster and let c_j be the center of the cluster. Note that the only 1 entries in a_i are corresponding to the indices in R_i. Now, if the other two (triple) columns in the cluster are the same, there must be at most one

entry of them corresponding to the indices in R_i that is 1. This is true by the construction of triple columns. Hence, in this case, at most one entry of c_j corresponding to the indices in R_i can be 1 and the cost is at least 5. Otherwise, there can be two distinct triple columns b_s and b_t in the cluster and at most two indices $z_1, z_2 \in R_i$ such that $z_1 \neq z_2$ and $b_s[z_1] = b_t[z_2] = 1$. By construction of the triple columns, there are no other indices $z \in R_i \setminus \{z_1, z_2\}$ such that $b_s[z] = 1$ or $b_t[z] = 1$. Thus, by the majority rule, at most two entries of c_j corresponding to the indices in R_i can be 1. Hence, the cost of a_i is at least 4. Now, as b_s and b_t are distinct, the cost of either one of them must be at least 1. It follows that the cost of this cluster is at least 5. □

Now, again consider the $2N - 2\ell - \ell_1$ element columns that are not in the first ℓ clusters. Let κ be the number of clusters that are not the first ℓ clusters and contain exactly 1 element column. This implies that $2N - 2\ell - \ell_1 - \kappa$ element columns are contained in the clusters which are not the first ℓ clusters and contain at least 2 element columns. By Claim 6.3.1, the cost of each such column is at least 5. By Claim 6.3.2, the cost of each of the κ clusters defined above is at least 5.

It follows that the total cost of the clustering is $12\ell_1 + 7\ell_2 + (2N - 2\ell - \ell_1 - \kappa)5 + 5\kappa = 10N - 3\ell$, as $\ell = \ell_1 + \ell_2$. Now, given that the cost is less than $7(1-\gamma)N + (23/3)\gamma N$.

$$
\begin{aligned}
&10N - 3\ell < 7(1-\gamma)N + (23/3)\gamma N = 7N + 2\gamma N/3 \\
&3N - 3\ell < 2\gamma N/3 \\
&\ell > (1 - 2\gamma/9)N
\end{aligned}
$$

□

As before, let ℓ_2 be the number of clusters out of the first ℓ clusters such that ℓ_2 contains a triple column.

Claim 6.3.3. $\ell_2 > (1 - 2\gamma/3)N$.

Proof. Again consider the cost of the given clustering. The cost of the ℓ_2 clusters is at least 7. The cost of the remaining $\ell - \ell_2$ clusters is exactly 12 as before. Now, as

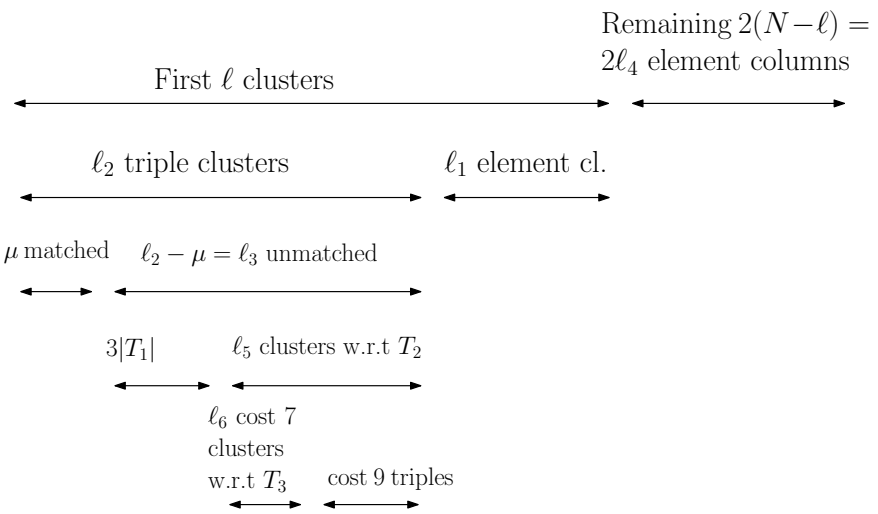

Figure 6.2: Hierarchy of the clusters. Illustration of the proof of Lemma 24.

$\ell > (1 - 2\gamma/9)N$ by Lemma 23,

$$7\ell_2 + 12((1 - 2\gamma/9)N - \ell_2) < 7(1 - \gamma)N + (23/3)\gamma N = 7N + 2\gamma N/3$$
$$7\ell_2 + 12N - 24\gamma N/9 - 12\ell_2 < 7N + 2\gamma N/3$$
$$5\ell_2 > 5N - 30\gamma N/9$$
$$\ell_2 > (1 - 2\gamma/3)N$$

□

We show that out of the ℓ_2 elements corresponding to these ℓ_2 clusters, more than $(1 - \gamma)N$ elements must be matched.

Lemma 24. *There is a matching that matches more than* $(1 - \gamma)N$ *elements.*

Proof. Consider the set of elements corresponding to the ℓ_2 clusters, each of which contains a triple column. Let M be a maximum matching involving these elements and triples that matches μ elements. We will show that $\mu > (1 - \gamma)N$. The total cost of the clusters corresponding to these matched elements is 7μ. Let ℓ_1 be the number of clusters out of the first ℓ clusters that contain all element columns (see Figure 6.2). The total cost of these clusters is $12\ell_1$. Note that $3\ell_1$ columns are involved in these clusters. For the remaining at least $2(N - \mu) - 3\ell_1$ element columns and correspondingly at least $N - \mu - 3\ell_1/2$ elements, the corresponding columns can either be in one cluster along with a triple column or split into two clusters. Let ℓ_3 be the number of such elements

whose columns are in one cluster along with a triple column. Also, let ℓ_4 be the remaining elements whose columns are split into two clusters (see Figure 6.2). By Claims 6.3.2 and 6.3.1, the cost of each split column is at least 4. Thus, the total cost corresponding to these ℓ_4 elements is at least $8\ell_4$. Now, we compute the cost corresponding to the ℓ_3 elements whose columns are in one cluster along with a triple column. Consider the set of triples involved in these clusters. Also, let T_1 be the set of triples whose three columns appear in these ℓ_3 clusters. The cost of such triple columns is at least $7+7+9=23$, as they are not a part of the maximum matching. Let ℓ_5 be the number of clusters among the ℓ_3 clusters where the triples in T_1 do not appear and T_2 be the set of associated triples. Each triple in T_2 thus appears in at most 2 clusters among the ℓ_3 clusters (see Figure 6.2). Let $T_3 \subseteq T_2$ be the set of triples each of which is only associated with the clusters of cost 7 and ℓ_6 be the number of these clusters. As these triples are not part of the maximum matching, each of them can cover at most two unmatched elements. Thus, the size of T_3 is at least $\ell_6/2$. Note that, by definition, at least one column of each such triple does not belong to the first ℓ clusters. We compute the cost of these triple columns. If such a triple column appears in all triple column clusters, the cost of the column is at least 3, by the construction of the triple columns and noting that two copies of the column cannot appear in the cluster. If such a triple is in a cluster with only one element column, its cost must be at least 2, as the element columns' at most one 1 entry can coincide with the 1 entries of the column. Now, if such a triple column appears in a cluster with two element columns, then the cost of the column is at least 1. However, the cost of the element columns must be at least 10. We charged each such element column a cost of 4 while charging the split columns corresponding to the ℓ_4 elements. So, we can charge $10-8=2$ additional cost to those element columns. Instead, we charge this to the triple column. Thus, its charged cost is $1+2=3$. Thus, the total cost corresponding to the triples in T_3 is at least $(\ell_6/2)\cdot 2$. The total cost of the clustering is at least,

$$
\begin{aligned}
&7\mu + 12\ell_1 + 8\ell_4 + (23/3)|T_1| + (\ell_5-\ell_6)((7+9)/2) + 7\ell_6 + (\ell_6/2)\cdot 2 \\
=&7\mu + 12\ell_1 + 8\ell_4 + (23/3)(\ell_3-\ell_5) + 8\ell_5 && (\text{as } 3|T_1| = \ell_3-\ell_5) \\
\geq&7\mu + 12\ell_1 + 8\ell_4 + (23/3)\ell_3 \\
\geq&7\mu + 12\ell_1 + (23/3)(\ell_3+\ell_4) \\
\geq&7\mu + 12\ell_1 + (23/3)(N-\mu-3\ell_1/2) && (\text{as } \ell_3+\ell_4 \geq N-\mu-3\ell_1/2) \\
=&7\mu + (23/3)(N-\mu) + \ell_1/2 \\
\geq&(23/3)N - (2/3)\mu && (\text{as } \ell_1 \geq 0)
\end{aligned}
$$

Now, we know a strict upper bound on this cost. Thus,

$$\begin{aligned}
&(23/3)N - (2/3)\mu < 7N + (2/3)\gamma N \\
&(23/3 - 7)N - (2/3)\gamma N < (2/3)\mu \\
&(2/3)N(1-\gamma) < (2/3)\mu \\
&\mu > (1-\gamma)N
\end{aligned}$$

□

We summarize the results of this section in the following theorem.

Theorem 12. *There exists a constant $\varepsilon_c > 0$, such that it is* NP*–hard to obtain a $(1+\varepsilon_c)$–approximation for ℓ_p-*Equal *k-*Median Clustering *with ℓ_0 (or ℓ_1) distances, even if the input points are binary, that is, are from $\{0,1\}^d$.*

Chapter 7

Discussions and Open Problems

In Chapter 4 of the thesis, we looked at variants of Discrete k-Median Clustering in a general metric space, where the candidate center set is either the same as the point set or selected from a prescribed finite set given as an input. Below, we briefly discuss these results and some future research directions.

- For Restricted k-Median Clustering, we designed an exact algorithm running in time $(1.89)^n \cdot n^{\mathcal{O}(1)}$ and showed that unless the Exponential Time Hypothesis fails, there is no algorithm for the problem running in time $2^{o(n)} \cdot n^{O(1)}$. Note that even if the distances satisfy the triangle inequality, the sum of *squares* of distances do not. However, our algorithm also works for k-Means, where we want to minimize the sum of squares of distances; or, even more generally, if we want to minimize the sum of z-th powers of distances, for some fixed $z \geq 1$. Our algorithm works for non-metric distance functions – it is easy to modify the construction of graph G to work with asymmetric distance functions, which are quite popular in the context of asymmetric travelling salesman problem. In particular, our algorithm and the hardness result also hold for k-Center. However, it is folklore that the *exact* versions of k-Center and Dominating set are equivalent. Thus, using the currently best-known algorithm for Dominating set by Iwata [51], it is possible to obtain an $\mathcal{O}^*((1.4689)^n)$ time algorithm for k-Center. Improving the running time (i.e., the base of the exponent) for the problem using the metric properties of distances remains an interesting future direction.

- We also studied k-Median Facility Location and designed an exact algorithm with running time $2^n \cdot (mn)^{\mathcal{O}(1)}$ and showed that an algorithm with running time $2^{(1-\epsilon)n} \cdot m^{\mathcal{O}(1)}$ for some fixed $\epsilon > 0$ is not possible for the problem unless the set cover conjecture fails. Observe that designing an algorithm for k-Median Facility

LOCATION with running time $2^m \cdot (mn)^{\mathcal{O}(1)}$ is trivial by simple enumeration. Hence, one of the natural questions that arise is the following. Is it possible to improve the base of the exponent by showing an algorithm with running time $(2-\epsilon)^m \cdot (mn)^{\mathcal{O}(1)}$ for some fixed $\epsilon > 0$ or this is not possible assuming some complexity-theoretic hypothesis, like SET COVER CONJECTURE, or Strong Exponential Time Hypothesis (SETH)?

In Chapter 5 of the thesis, we considered CATEGORICAL CAPACITATED k-MEDIAN CLUSTERING and its variants where sizes of the clusters satisfy special balance properties. Next, we briefly look at the results we obtained and some open problems.

- We designed an FPT algorithm for CATEGORICAL CAPACITATED k-MEDIAN CLUSTERING running in time $2^{(B \log B)}|\Sigma|^m \cdot (mn)^{\mathcal{O}(1)}$. CATEGORICAL CAPACITATED k-MEDIAN CLUSTERING is the most general among its considered variants BALANCED CATEGORICAL k-MEDIAN CLUSTERING and FACTOR-BALANCED CATEGORICAL k-MEDIAN CLUSTERING. Hence, the above problems also admit an FPT with a similar running time. It is natural to ask whether one can improve the dependence on B. We do not know the answer to this question even for the special case of CATEGORICAL k-MEDIAN CLUSTERING where the size of each cluster is the same. Also, besides the considered size constraints, it may be interesting to consider other variants. For example, in CATEGORICAL CAPACITATED k-MEDIAN CLUSTERING, the size constraints p and q are universal for all clusters. However, one may consider the case when the cluster sizes are given by individual constraints.

- We observed that CATEGORICAL CAPACITATED k-MEDIAN CLUSTERING, BALANCED CATEGORICAL k-MEDIAN CLUSTERING, and FACTOR-BALANCED CATEGORICAL k-MEDIAN CLUSTERING has no polynomial kernel when parameterized by B, unless $\mathsf{NP} \subseteq \mathsf{coNP}/\mathsf{poly}$, even if $\Sigma = \{0, 1\}$.

- We desiged a polynomial kernel for BALANCED CATEGORICAL k-MEDIAN CLUSTERING with $\mathcal{O}(k(B + \delta k))$ points from the space of dimension $O(B(B + k))$ over an alphabet of size at most $B + k$.

  - This leads to the question whether FACTOR-BALANCED CATEGORICAL k-MEDIAN CLUSTERING admits a polynomial kernel when parameterized by k and B, assuming that α is a fixed constant. A more general question is whether there are polynomial kernels for CATEGORICAL CAPACITATED k-MEDIAN CLUSTERING, BALANCED CATEGORICAL k-MEDIAN CLUSTERING,

and Factor-Balanced Categorical k-Median Clustering parameterized by k and B. Notice that Categorical k-Median Clustering has a polynomial kernel for this parameterization [37, Theorem 2].

– Are there polynomial *Turing kernels* and do these problems admit polynomial *lossy kernels*, that is, approximative kernels? (We refer to the book by Lokshtanov et al. [41] for the definition of the notions.)

In Chapter 6 of the thesis, we studied Parameterized ℓ_p-Equal k-Median Clustering. We initiated the study of lossy kernelization for clustering problems. The following are the results and some of the open problems arising from the chapter.

- We designed a factor 2-approximation polynomial size kernel for Parameterized ℓ_p-Equal k-Median Clustering. It is natural to ask whether the approximation factor can be improved or not. In particular, does the problem admit a *polynomial size approximate kernelization scheme* (PSAKS) that is a lossy kernelization analog of PTAS (we refer to the book by Fomin et al. [41] for the definition)?

- We proved that ℓ_p-Equal k-Median Clustering with ℓ_0 and ℓ_1 distances is APX-hard. This refutes the existence of PTAS and makes it natural to ask about PSAKS.

- We also believe it is interesting to consider the variants of the considered problems for means instead of medians. Here, the cost of a collection of points $\mathbf{X} \subseteq \mathbb{Z}^d$ is defined as $\min_{\mathbf{c}\in\mathbb{R}^d} \sum_{\mathbf{x}\in\mathbf{X}} (\mathsf{dist}_p(\mathbf{c},\mathbf{x}))^2$ for $p \geq 1$. Clearly, if $p = 1$, that is, in the case of the Manhattan norm, our results hold. However, for $p \geq 2$, we cannot translate our results directly because our arguments rely on the triangle inequality. We believe that lossy kernelization may be a natural tool for the lucrative area of approximation algorithms for clustering problems.

Bibliography

[1] M. R. Ackermann, J. Blömer, and C. Sohler, *Clustering for metric and nonmetric distance measures*, ACM Trans. Algorithms, 6 (2010). 2, 6

[2] P. K. Agarwal and C. M. Procopiuc, *Exact and approximation algorithms for clustering*, Algorithmica, 33 (2002), pp. 201–226. 26

[3] C. C. Aggarwal and C. K. Reddy, eds., *Data Clustering: Algorithms and Applications*, CRC Press, 2013. 1

[4] N. Alon and B. Sudakov, *On two segmentation problems*, Journal of Algorithms, 33 (1999), pp. 173–184. 9

[5] N. Alon, R. Yuster, and U. Zwick, *Color-coding*, J. ACM, 42 (1995), pp. 844–856. 54, 69

[6] S. Arora, P. Raghavan, and S. Rao, *Approximation schemes for euclidean* k*-medians and related problems*, in Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, Dallas, Texas, USA, May 23-26, 1998, J. S. Vitter, ed., ACM, 1998, pp. 106–113. 1, 4

[7] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit, *Local search heuristics for k-median and facility location problems*, SIAM J. Comput., 33 (2004), pp. 544–562. 3

[8] P. Awasthi, A. S. Bandeira, M. Charikar, R. Krishnaswamy, S. Villar, and R. Ward, *Relax, no need to round: Integrality of clustering formulations*, in Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science, ITCS 2015, Rehovot, Israel, January 11-13, 2015, T. Roughgarden, ed., ACM, 2015, pp. 191–200. 3

[9] M. Badoiu, S. Har-Peled, and P. Indyk, *Approximate clustering via core-sets*, in Proceedings on 34th Annual ACM Symposium on Theory of Computing, May 19-21, 2002, Montréal, Québec, Canada, J. H. Reif, ed., ACM, 2002, pp. 250–257. 2, 6

[10] D. Baker, V. Braverman, L. Huang, S. H.-C. Jiang, R. Krauthgamer, and X. Wu, *Coresets for clustering in graphs of bounded treewidth*, in International Conference on Machine Learning, PMLR, 2020, pp. 569–579. 5

[11] S. Bandyapadhyay, F. V. Fomin, and K. Simonov, *On coresets for fair clustering in metric and euclidean spaces and their applications*, in 48th International Colloquium on Automata, Languages, and Programming (ICALP), vol. 198 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, pp. 23:1–23:15. 9

[12] A. Banerjee and J. Ghosh, *Clustering with balancing constraints*, in Constrained clustering: advances in algorithms, theory, and applications, CRC Press, 2008, pp. 171–200. 2

[13] S. Basu, I. Davidson, and K. L. Wagstaff, eds., *Constrained clustering*, Chapman & Hall/CRC Data Mining and Knowledge Discovery Series, CRC Press, Boca Raton, FL, 2009. Advances in algorithms, theory, and applications. 1, 2

[14] A. Bhattacharya, R. Jaiswal, and A. Kumar, *Faster Algorithms for the Constrained k-means Problem*, Theory of Computing Systems, 62 (2018), pp. 93–115. 8

[15] S. Bhattacharya, P. Chalermsook, K. Mehlhorn, and A. Neumann, *New approximability results for the robust k-median problem*, in Algorithm Theory - SWAT 2014 - 14th Scandinavian Symposium and Workshops, Copenhagen, Denmark, July 2-4, 2014. Proceedings, R. Ravi and I. L. Gørtz, eds., vol. 8503 of Lecture Notes in Computer Science, Springer, 2014, pp. 50–61. 3

[16] J. Chalopin and D. Paulusma, *Packing bipartite graphs with covers of complete bipartite graphs*, Discret. Appl. Math., 168 (2014), pp. 40–50. 27

[17] M. Charikar, C. Chekuri, A. Goel, and S. Guha, *Rounding via trees: Deterministic approximation algorithms for group steiner trees and* k*-median*, in Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, Dallas, Texas, USA, May 23-26, 1998, J. S. Vitter, ed., ACM, 1998, pp. 114–123. 1, 5

[18] M. Charikar and S. Li, *A dependent ℓ_p-rounding approach for the k-median problem*, in Automata, Languages, and Programming - 39th International Colloquium, ICALP 2012, Warwick, UK, July 9-13, 2012, Proceedings, Part I, A. Czumaj, K. Mehlhorn, A. M. Pitts, and R. Wattenhofer, eds., vol. 7391 of Lecture Notes in Computer Science, Springer, 2012, pp. 194–205. 3

[19] M. Chrobak, C. Kenyon, and N. E. Young, *The reverse greedy algorithm for the metric* k*-median problem*, Inf. Process. Lett., 97 (2006), pp. 68–72. 3

[20] V. Cohen-Addad, A. de Mesmay, E. Rotenberg, and A. Roytman, *The bane of low-dimensionality clustering*, in Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018, A. Czumaj, ed., SIAM, 2018, pp. 441–456. 4, 6

[21] V. Cohen-Addad, H. Esfandiari, V. S. Mirrokni, and S. Narayanan, *Improved approximations for euclidean* k*-means and* k*-median, via nested quasi-independent sets*, in STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022, S. Leonardi and A. Gupta, eds., ACM, 2022, pp. 1621–1628. 3

[22] V. Cohen-Addad, F. Grandoni, E. Lee, and C. Schwiegelshohn, *Breaching the 2 LMP approximation barrier for facility location with applications to* k*-median*, in Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023, N. Bansal and V. Nagarajan, eds., SIAM, 2023, pp. 940–986. 3

[23] V. Cohen-Addad, A. Gupta, A. Kumar, E. Lee, and J. Li, *Tight FPT approximations for k-median and k-means*, in 46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece, C. Baier, I. Chatzigiannakis, P. Flocchini, and S. Leonardi, eds., vol. 132 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, pp. 42:1–42:14. 4, 6

[24] V. Cohen-Addad, Karthik C. S., and E. Lee, *On approximability of clustering problems without candidate centers*, in Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021, D. Marx, ed., SIAM, 2021, pp. 2635–2648. 4

[25] V. Cohen-Addad, P. N. Klein, and C. Mathieu, *Local search yields approximation schemes for k-means and k-median in euclidean and minor-free metrics*, SIAM J. Comput., 48 (2019), pp. 644–667. 4

[26] V. Cohen-Addad and J. Li, *On the fixed-parameter tractability of capacitated clustering*, in 46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece, C. Baier, I. Chatzigiannakis, P. Flocchini, and S. Leonardi, eds., vol. 132 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, pp. 41:1–41:14. 5

[27] S. A. Cook, *The complexity of theorem-proving procedures*, Proceedings of the third annual ACM symposium on Theory of computing, (1971). 16

[28] M. Cygan, H. Dell, D. Lokshtanov, D. Marx, J. Nederlof, Y. Okamoto, R. Paturi, S. Saurabh, and M. Wahlström, *On problems as hard as cnf-sat*, ACM Trans. Algorithms, 12 (2016). 17, 28

[29] M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh, *Parameterized Algorithms*, Springer, 2015. 15, 16, 38, 39, 54, 69

[30] H. Dell and D. Marx, *Kernelization of packing problems*, CoRR, abs/1812.03155 (2018). 14, 15, 98, 99

[31] H. Ding and J. Xu, *A unified framework for clustering constrained data without locality property*, Algorithmica, 82 (2020), pp. 808–852. 2, 8

[32] U. Feige, *NP-hardness of hypercube 2-segmentation*, CoRR, abs/1411.0821 (2014). 3, 5, 44

[33] D. Feldman and M. Langberg, *A unified framework for approximating and clustering data*, in Proceedings of the 43rd ACM Symposium on Theory of Computing, STOC 2011, San Jose, CA, USA, 6-8 June 2011, L. Fortnow and S. P. Vadhan, eds., ACM, 2011, pp. 569–578. 5, 6

[34] D. Feldman, M. Schmidt, and C. Sohler, *Turning big data into tiny data: Constant-size coresets for k-means, pca, and projective clustering*, SIAM Journal on Computing, 49 (2020), pp. 601–657. 5

[35] A. E. Feldmann, Karthik C. S., E. Lee, and P. Manurangsi, *A survey on approximation in parameterized complexity: Hardness and algorithms*, Algorithms, 13 (2020), p. 146. 3

[36] Q. Feng, Z. Zhang, Z. Huang, J. Xu, and J. Wang, *A unified framework of FPT approximation algorithms for clustering problems*, in 31st International Symposium on Algorithms and Computation, ISAAC 2020, December 14-18, 2020, Hong Kong, China (Virtual Conference), Y. Cao, S. Cheng, and M. Li, eds., vol. 181 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020, pp. 5:1–5:17. 4

[37] F. V. Fomin, P. A. Golovach, and F. Panolan, *Parameterized low-rank binary matrix approximation*, Data Min. Knowl. Discov., 34 (2020), pp. 478–532. 5, 6, 41, 42, 43, 50, 71, 77, 88, 115

[38] F. V. Fomin, P. A. Golovach, and K. Simonov, *Parameterized* k*-clustering: Tractability island*, J. Comput. Syst. Sci., 117 (2021), pp. 50–74. 5, 42, 45, 50, 52

[39] F. V. Fomin and D. Kratsch, *Exact Exponential Algorithms*, Texts in Theoretical Computer Science. An EATCS Series, Springer, 2010. 30

[40] F. V. Fomin, D. Kratsch, and G. J. Woeginger, *Exact (exponential) algorithms for the dominating set problem*, in Graph-Theoretic Concepts in Computer Science, J. Hromkovič, M. Nagl, and B. Westfechtel, eds., Berlin, Heidelberg, 2005, Springer Berlin Heidelberg, pp. 245–256. 17

[41] F. V. Fomin, D. Lokshtanov, S. Saurabh, and M. Zehavi, *Kernelization*, Cambridge University Press, Cambridge, 2019. Theory of parameterized preprocessing. 18, 115

[42] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications, Second Edition*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2020. 1

[43] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, 1979. 27, 44

[44] S. Ghiasi, A. Srivastava, X. Yang, and M. Sarrafzadeh, *Optimal energy aware clustering in sensor networks*, Sensors, 2 (2002), pp. 258–269. 2

[45] S. Guha and S. Khuller, *Greedy strikes back: Improved facility location algorithms*, Journal of Algorithms, 31 (1999), pp. 228–248. 3, 4, 5

[46] S. Guha, A. Meyerson, and K. Munagala, *Hierarchical placement and network design problems*, Proceedings 41st Annual Symposium on Foundations of Computer Science, (2000), pp. 603–612. 2

[47] G. Gupta and M. Younis, *Load-balanced clustering of wireless sensor networks*, in IEEE International Conference on Communications (ICC), vol. 3, IEEE, 2003, pp. 1848–1852. 2

[48] S. Har-Peled and S. Mazumdar, *On coresets for k-means and k-median clustering*, in Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, June 13-16, 2004, L. Babai, ed., ACM, 2004, pp. 291–300. 5

[49] F. Höppner and F. Klawonn, *Clustering with size constraints*, in Computational Intelligence Paradigms, Innovative Applications, L. C. Jain, M. Sato-Ilic, M. Virvou, G. A. Tsihrintzis, V. E. Balas, and C. Abeynayake, eds., vol. 137, Springer, 2008, pp. 167–180. 7

[50] R. Impagliazzo and R. Paturi, *Complexity of k-sat*, in Proceedings. Fourteenth Annual IEEE Conference on Computational Complexity (Formerly: Structure in Complexity Theory Conference) (Cat.No.99CB36317), 1999, pp. 237–240. 16

[51] Y. Iwata, *A faster algorithm for dominating set analyzed by the potential method*, in International Symposium on Parameterized and Exact Computation, Springer, 2011, pp. 41–54. 113

[52] E. Jack, *Paths, trees, and flowers*, Canadian Journal of Mathematics, 17 (1965), p. 449–467. 27, 29, 34

[53] A. K. Jain, M. N. Murty, and P. J. Flynn, *Data clustering: a review*, ACM Comput. Surv., 31 (1999), pp. 264–323. 1

[54] K. Jain, M. Mahdian, and A. Saberi, *A new greedy approach for facility location problems*, in Proceedings on 34th Annual ACM Symposium on Theory of Computing, May 19-21, 2002, Montréal, Québec, Canada, J. H. Reif, ed., ACM, 2002, pp. 731–740. 3

[55] K. Jain and V. V. Vazirani, *Approximation algorithms for metric facility location and k-median problems using the primal-dual schema and lagrangian relaxation*, J. ACM, 48 (2001), p. 274–296. 3

[56] D. G. Kirkpatrick and P. Hell, *On the complexity of general graph factor problems*, SIAM J. Comput., 12 (1983), pp. 601–609. 27

[57] J. M. Kleinberg and É. Tardos, *Algorithm design*, Addison-Wesley, 2006. 93, 94

[58] S. G. Kolliopoulos and S. Rao, *A nearly linear-time approximation scheme for the euclidean k-median problem*, SIAM J. Comput., 37 (2007), pp. 757–782. 4

[59] H. W. Kuhn, *The Hungarian method for the assignment problem*, Naval Res. Logist. Quart., 2 (1955), pp. 83–97. 22, 24

[60] A. Kumar, Y. Sabharwal, and S. Sen, *Linear-time approximation schemes for clustering problems in any dimensions*, J. ACM, 57 (2010), pp. 5:1–5:32. 2, 5, 6, 8, 9

[61] S. Li and O. Svensson, *Approximating k-median via pseudo-approximation*, in Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing, STOC '13, New York, NY, USA, 2013, Association for Computing Machinery, p. 901–910. 3

[62] T. Li, *A general model for clustering binary data*, in KDD'05, 2005, pp. 188–197. 2

[63] J.-H. Lin and J. S. Vitter, *ε-approximations with minimum packing constraint violation (extended abstract)*, in Proceedings of the Twenty-Fourth Annual ACM

Symposium on Theory of Computing, STOC '92, New York, NY, USA, 1992, Association for Computing Machinery, p. 771–782. 1

[64] D. Lokshtanov, D. Marx, and S. Saurabh, *Lower bounds based on the exponential time hypothesis*, Bull. EATCS, 105 (2011), pp. 41–72. 16, 34

[65] D. Lokshtanov, F. Panolan, M. S. Ramanujan, and S. Saurabh, *Lossy kernelization*, in Proceedings of the 49th Annual ACM Symposium on Theory of Computing (STOC), ACM, 2017, pp. 224–237. 9, 80

[66] L. Lovász and M. D. Plummer, *Matching theory*, AMS Chelsea Publishing, Providence, RI, 2009. 22, 24, 100

[67] P. J. Lynch, S. Horton, and S. Horton, *Web style guide: Basic design principles for creating web sites*, Universities Press, 1999. 2

[68] D. Marx, *Closest substring problems with small distances*, SIAM J. Comput., 38 (2008), pp. 1382–1410. 50, 52

[69] J. Matoušek, *On approximate geometric k -clustering*, Discrete & Computational Geometry, 24 (2000), pp. 61–84. 4

[70] N. Megiddo and K. J. Supowit, *On the complexity of some common geometric location problems*, SIAM J. Comput., 13 (1984), pp. 182–196. 3

[71] M. Naor, L. J. Schulman, and A. Srinivasan, *Splitters and near-optimal derandomization*, in FOCS 1995, IEEE Computer Society, 1995, pp. 182–191. 69

[72] R. Otter, *The number of trees*, Ann. of Math. (2), 49 (1948), pp. 583–599. 54, 69

[73] E. Petrank, *The hardness of approximation: Gap location*, Comput. Complex., 4 (1994), pp. 133–157. 13, 106

[74] C. Rösner and M. Schmidt, *Privacy preserving clustering with constraints*, in 45th International Colloquium on Automata, Languages, and Programming (ICALP 2018), Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018. 2

[75] D. B. Shmoys, É. Tardos, and K. Aardal, *Approximation algorithms for facility location problems (extended abstract)*, in Symposium on the Theory of Computing, 1997. 3

[76] C. Sohler and D. P. Woodruff, *Strong coresets for k-median and subspace approximation: Goodbye dimension*, in Proceedings of the 59th Annual Symposium on Foundations of Computer Science (FOCS), IEEE, 2018, pp. 802–813. 5

[77] L. Sweeney, *k-anonymity: A model for protecting privacy*, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10 (2002), pp. 557–570. 8

[78] D. Vallejo-Huanga, P. Morillo, and C. Ferri, *Semi-supervised clustering algorithms for grouping scientific articles*, in International Conference on Computational Science (ICCS), vol. 108 of Procedia Computer Science, Elsevier, 2017, pp. 325–334. 7, 42

[79] V. V. Vazirani, *Approximation algorithms*, Approximation Algorithms, (2001). 13

[80] D. P. Williamson and D. B. Shmoys, *The Design of Approximation Algorithms*, Cambridge University Press, 2011. 13

[81] R. A. Wright, L. B. Richmond, A. M. Odlyzko, and B. D. McKay, *Constant time generation of free trees*, SIAM J. Comput., 15 (1986), pp. 540–548. 54, 69

[82] Y. Yang and B. Padmanabhan, *Segmenting customer transactions using a pattern-based clustering approach*, in Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM), IEEE Computer Society, 2003, pp. 411–418. 2

[83] Z. Zhang, T. Li, C. Ding, and X. Zhang, *Binary matrix factorization with applications*, in ICDM'07, IEEE, 2007, pp. 391–400. 2

Graphic design: Communication Division, UiB / Print: Skipnes Kommunikasjon AS

uib.no

ISBN: 9788230859520 (print)
9788230853566 (PDF)