# Three small discoveries in the field of (in-)approximability.

A doctoral dissertation
by
**Jan Marcinkowski**

Uniwersytet Wrocławski

Wydział Matematyki
i Informatyki

Wrocław-Tylicz, AD 2021

## Streszczenie

W niniejszej rozprawie doktorskiej prezentujemy wyniki dotyczące tematu *aproksymowalności* wybranych *optymalizacyjnych problemów kombinatorycznych*:

1. W rozdziale 2 pokazujemy, że trywialny algorytm 2-aproksymacyjny dla problemu Minimum Maximal Matching jest najlepszym, jaki można uzyskać (także w szczególnym przypadku grafów dwudzielnych). Istnienie algorytmu $(2-\gamma)$-aproksymacyjnego (dla dowolnej stałej $\gamma$) przeczyłoby hipotezie *Unique Games Conjecture* (lub jej mocniejszym wariantom, w przypadku dwudzielnym).
2. W rozdziale 3 prezentujemy algorytm aproksymacyjny dla problemu rozstrzygania wyborów w ordynacji zwanej Proportional Approval Voting (a także w całej nieskończonej rodzinie powiązanych ordynacji wyborczych). Pokazujemy też, że algorytm ten jest najlepszą możliwą aproksymacją, jaką może uzyskać algorytm działający w czasie wielomianowym (o ile $\mathbf{P} \neq \mathbf{NP}$), a nawet w czasie parametryzowanym rozmiarem wybieranego zgromadzenia (o ile prawdziwa jest hipoteza Gap-ETH).
3. W rozdziale 4 konstruujemy algorytm o stałym współczynniku aproksymacji dla problemu Capacitated $k$-Median, który działa w czasie parametryzowanym liczbą $k$. Problem ten nie może być rozwiązany dokładnie w takiej złożoności (o ile $\mathbf{FPT} \neq \mathbf{W}[2]$), a algorytm aproksymacyjny o stałym współczynniku działający w czasie wielomianowym — chociaż nie jest wykluczony — pozostaje niedosięgniony.

## Abstract

In this thesis we present the results on *approximability* of selected *combinatorial optimisation problems*:

1. In Chapter 2 we are showing that a trivial 2-approximation algorithm for the Minimum Maximal Matching problem is the limit of what can be expected (also for bipartite graphs). Existence of a $(2-\gamma)$-approximation algorithm (for any constant $\gamma$) would contradict the *Unique Games Conjecture* (or its stronger variants, in the bipartite case).
2. In Chapter 3 we present an approximation algorithm for deciding winners in the Proportional Approval Voting system (and an infinite family of related electoral systems). We additionally show, that a better polynomial-time approximation would imply $\mathbf{P} = \mathbf{NP}$, and a better approximation running in time parametrised by the committee size would contradict Gap-ETH.
3. In Chapter 4 we construct a constant-factor approximation algorithm for the Capacitated $k$-Median problem, working in time parametrised by $k$. The problem in question cannot be solved exactly within such a complexity (unless $\mathbf{FPT} = \mathbf{W}[2]$), and a polynomial-time approximation—although not strictly out of question—remains out of reach.

# Contents

**Notations** v

**Introduction** 1
1 Computational tasks and their complexity . . . 1
2 Approximation algorithms . . . 2
3 This dissertation . . . 3
4 Acknowledgements . . . 4

1 **Gap problems** 5
1 Label Cover . . . 5
2 Unique Games and variations . . . 6

2 **The Minimum Maximal Matching problem** 9
1 Related work . . . 9
2 Our method . . . 12
3 General graphs . . . 12
3.1 The vertex cover instance . . . 13
3.2 Supplementing $H$ with edges . . . 15
4 Bipartite graphs I: Bipartisation and path covers . . . 16
4.1 Covering with paths . . . 17
5 Bipartite graphs II: From Strong UGC to MMM . . . 17
6 Bipartite graphs III: Hardness based on SSEH . . . 20
6.1 Reducing from SSEH to UGC . . . 21
6.2 Max Uncut Hypergraph Bisection . . . 23
6.3 From MUCHB to Bipartite MMM . . . 26
7 Notes . . . 31

3 **Proportional Approval Voting** 32
1 Algorithmic question . . . 32
2 The approximation algorithm . . . 34

2.1 Step 1: Three-valued $x^*$ . . . . . . . . . . . . . . . . . . . . . . . . 36
2.2 Step 2: Getting rid of ones . . . . . . . . . . . . . . . . . . . . . . . 37
2.3 Step 3: Comparing Binomial and Poisson . . . . . . . . . . . . . . 38
2.4 Step 4: Changing the mean of Poisson . . . . . . . . . . . . . . . 38
3 Hardness of Approximation . . . . . . . . . . . . . . . . . . . . . . . . 40
3.1 Comparing Binomial and Poisson . . . . . . . . . . . . . . . . . . . 41
3.2 The Reduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 43
4 Determining performance of the Greedy Algorithm . . . . . . . . . . . 47
5 Notes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 49

**4 k-Medians clustering with capacities** 51
1 Related work . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 52
2 $l$-Centred instances . . . . . . . . . . . . . . . . . . . . . . . . . . . 53
3 Polynomial-time $\mathcal{O}(\log k)$-approximation for CkM . . . . . . . . . . . . 56
3.1 Solving CkM on trees . . . . . . . . . . . . . . . . . . . . . . . . . . 57
4 Constant-factor approximation . . . . . . . . . . . . . . . . . . . . . . 58
4.1 Mapping clients to open facilities . . . . . . . . . . . . . . . . . . . 58
4.2 Solving Uniform CkM on $l$-centred metrics . . . . . . . . . . . . . 59
4.3 Approximating non-uniform CkM . . . . . . . . . . . . . . . . . . . 60
4.4 Choosing the right $l$ . . . . . . . . . . . . . . . . . . . . . . . . . . 62
5 Notes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 63

**Bibliography** 64

# Notations

$[m]$ The set of (positive) natural numbers up to $m$: $\{1,\ldots,m\}$.

$\bar{x}$ Negation. For a boolean value, $\bar{b} = \neg b$. For a boolean vector, $\bar{x}_i = \overline{x_i} = \neg x_i$.

$\circ$ Denotes function composition. If $\pi$ is a permutation and $x$ is a vector, $x \circ \pi$ is the vector with permuted coordinates: $(x \circ \pi)_i = x_{\pi(i)}$.

$\oplus$ XOR. We slightly abuse the notation by writing $b \oplus x$ for a boolean value $b$ and a boolean vector $x$. The operation applies to every element of $x$, so $(b \oplus x)_i = b \oplus x_i$.

$\{x\}$ Fractional part of a real number $x$. $\{x\} = x - \lfloor x \rfloor$.

$\uplus$ Disjoint union. $X \uplus Y = X \cup Y$ and additionally asserts that $X \cap Y = \varnothing$.

$\otimes$ Tensor product. For two vectors $\mathbf{v}$ and $\mathbf{w}$, $\mathbf{v} \otimes \mathbf{w}$ is the same as their outer product $\mathbf{v}\mathbf{w}^\intercal$—a matrix where every element of $\mathbf{v}$ is combined with every element of $\mathbf{w}$.

For two graphs $G$ and $H$, $G \otimes H$ is a graph with the vertex set $V(G) \times V(H)$ and an edge connecting $\langle u_1, v_1 \rangle$ with $\langle u_2, v_2 \rangle$ only if $(u_1, u_2) \in E(G)$ and $(v_1, v_2) \in E(H)$. $G^{\otimes k} = G \otimes G \otimes \cdots \otimes G$ ($k$ times) is a graph in which each vertex corresponds to a vector of $k$ vertices of $G$ and two such vectors are connected if there is a connection in $G$ on each coordinate.

$\mathfrak{S}_m$ A set of permutations of $[m]$. In some other sources called $S_m$ (a symmetric group of degree $m$).

$\sim$ Sample from a distribution. When $S$ is a set, $s \sim S$ means sampling uniformly from $S$. When $x^*$ is a vector of real numbers between 0 and 1, $x \sim x^*$ means sampling each coordinate $x_i$ independently from $\mathrm{Ber}(x_i^*)$.

# Introduction

We are Computer Scientists. Our role is to investigate—with mathematical scrutiny—which problems can be solved with computers, and how efficiently it can be done. Computer Science was started by a realisation (now known as *Church-Turing Thesis*) that if any intelligent life in the universe has computing machines, they would have the same computing power as ours, and most likely—however their machines were built—it should be possible to emulate their computer with ours with only a polynomial overhead in number of operations (anything they compute in $n$ steps, we can compute in $\mathcal{O}(n^c)$ for some constant $c$). This has led us to a (rarely contested) agreement that every problem which can be solved in time polynomial in the description of problem instance is 'practical' (we say it is in class **P**), even though some problems may be more suited for computers on Earth and some other for those belonging to aliens.

## 1. Computational tasks and their complexity

The problems at the focus of Computer Science are combinatorial in nature, and usually their application to practical use of computers is indirect. Some important tasks turn out to be very simple.

**EXAMPLE** *(*Maximum Acyclic Undirected Subgraph*)*.

In the `MAUS` problem we are given and undirected graph, and must find the largest possible set of edges which form an acyclic subgraph. This can be solved very easily, in time almost proportional to the time needed to read the input graph, as the maximum acyclic subgraph is the same as a spanning forest.

A small change in the description of the task may however put it in a very different place on the Computational Complexity map.

**EXAMPLE** *(Maximum Acyclic Subgraph)*.

In the MAS problem we have the same task as in MAUS, but the input graph is directed. We will show that it is unlikely, that MAS $\in$ **P**, by reducing another classic problem called Vertex Cover (VC) to it. Thus by solving MAS we would also solve VC in polynomial time.

**Reduction 1** (Karp [Kar72])
Input: A VC instance—an undirected graph $G = \langle V, E\rangle$.
Output: A MAS instance $H$ with vertex set $V_H = V \times \{0,1\}$ and arc set $E_H = E_H^V \cup E_H^E$, where $E_H^V = \left\{(\langle u,0\rangle, \langle u,1\rangle) \mid u \in V\right\}$, and $E_H^E = \left\{(\langle u,1\rangle, \langle v,0\rangle) \mid (u,v) \in E\right\}$.

One can verify that if $G$ has a vertex cover (subset of $V$ hitting every edge) of size $k$, then removing $k$ edges from $E_H^V$ suffices to make digraph $H$ acyclic (this statement is called *the completeness* of the reduction). Moreover, if $H$ can be made acyclic by removing $l$ arcs, then there is a vertex cover of $l$ vertices in graph $G$ (this states *the soundness* of our reduction).

## 2. Approximation algorithms

Having built a web of reductions like the one above, Computer Scientists began to look for a way around the common difficulty (**NP**-hardness) of problems like MAS or VC. One way around is *approximation algorithms*. Such an algorithm has an *approximation ratio* of $\rho$ if it returns solutions whose value is within $\rho$ of the optimal value for the input instance.

**EXAMPLE** *(1/2-approximation for MAS)*.

We can number the vertices of the graph arbitrarily. This ordering determines two sets: of *forward* arcs (ones that point from lower to higher number), and *backward* arcs. Let us pick the larger of these sets and call it $A$. The graph $\langle V, A\rangle$ is clearly acyclic, since all arcs agree with the same linear ordering of vertices. Moreover $|A| \geqslant \frac{|E|}{2}$, so for any acyclic set $P$ of edges (including the optimal one), $|A| \geqslant \frac{|P|}{2}$.

In the field of approximation algorithms—which has been central to Computer Science since 1970s—we try to construct solutions with $\rho$ as close to 1 as possible. More generally, we can ask the question ‘what is the best possible approximation ratio' for a particular problem, or 'whether such a constant approximation ratio can be even guaranteed' by a polynomial-time algorithm. Many general techniques have been developed for constructing approximation algorithms, but in the end, the exact answer must be given individually for each computational task.

Giving negative answers to approximability questions has been enabled by a chain of results (starting with the **PCP** theorem [Aro+98]) and hypotheses (most notably the *Unique Games Conjecture* [Kho02]). They allow us to construct reductions like above, but proving *inapproximability*, i.e. impossibility of approximation within a certain ratio. For example, an inapproximability result is known for the Maximum Acyclic Subgraph.

**Theorem 2** (Guruswami, Manokaran, Raghavendra [GMR08])
*Assuming Unique Games Conjecture, for any $\gamma > 0$, no polynomial-time algorithm can $(\frac{1}{2} + \gamma)$-approximate MAS.*

# 3. This dissertation

In this thesis we tackle the *approximability question* for three combinatorial problems, each of very different nature. In Chapter 2, we describe our hardness results for a problem called Minimum Maximal Matching. It is a somewhat similar story to the Maximum Acyclic Subgraph—there is a trivial approximation algorithm with ratio equal to 2, and a considerable effort had been invested into finding better approximation algorithms (i.e., with smaller ratio). In our papers [DLM19] (joint work with Szymon Dudycz and Mateusz Lewandowski; published at IPCO 2019) and [DMM20] (joint work with Szymon Dudycz and Pasin Manurangsi; pending publication) we prove such an algorithm impossible, conditionally on the Unique Games Conjecture. Our results effectively close a long line of research on the approximability of Minimum Maximal Matching. While proving the inapproximability, we need to look deeply into previous results on hardness of approximation of VC and Balanced Biclique, as well as the relationship between *Small Set Expansion Hypothesis* and the Unique Games Conjecture. Our write-up may be helpful to those willing to understand them but struggling throughout the (sometimes obscure) writing of the original papers.

In Chapter 3 we revisit the task of counting votes in the Proportional Approval Voting and related electoral systems. This problem—although dressed in the robes of political sciences—is a computational task similar to Maximum Coverage, only with a generalised objective function[1]. In our paper [Dud+20] (joint work with Szymon Dudycz, Pasin Manurangsi and Krzysztof Sornat; published ad IJCAI 2020), we give a new approximation algorithm (one for entire family of voting systems), and show that it cannot be improved upon, even by an algorithm with super-polynomial (*parametrised*) complexity.

In Chapter 4 we take on a metric clustering problem called Capacitated $k$-Median. Although the problem has been around for at least three decades, the question of its approximability is still wide open: no constant-factor approximation algorithm is known, yet there are no good reasons to believe such an algorithm to be impossible. While we are not resolving this question, in our paper [Ada+19] (joint work with Marek Adamczyk, Jarosław Byrka, Syed Mohammad Meesum and Michał Włodarczyk; published at ESA 2019) we follow a recent fashion and allow our algorithm to run in super-polynomial time (parametrised by $k$). We show, that this relaxation of the rules gives way to a rather simple constant-factor approximation for the problem.

The Chapters 2, 3, and 4 are independent and can be read in any order. They are based on respective published articles, but offer a much wider (and—hopefully—more accessible) writing, not constrained by the space limitation of conference proceedings. Chapter 1 introduces the reader to the land of inapproximability results and conjectures, which become the starting point for our reductions in Chapters 2 and 3.

## 4. Acknowledgements

I am very grateful to my PhD supervisor Jarosław Byrka, who encouraged me to follow the path of science. He gave me freedom seldom experienced by graduate students. I thank my teachers, most notably Katarzyna Paluch, who attracted me to the world of Combinatorial Optimisation and also suggested that we work on `MMM`. I thank my collaborators for the time and the joy, among them Szymon Dudycz, with whom we have been learning maths together for the last twenty-four years.

I am indebted to my family. My parents have taught me to climb mountains—both real and intellectual—and we still enjoy hiking together. My wife Patrycja not only is my greatest companion but also a professional touching people's lives with her daily work, which makes me immensely proud. Our daughter Jadwiga makes sure I get up from bed every day.

I would also like to thank the hard-working Polish taxpayer who mines coal, strikes iron, and sows the land so that we can travel to conferences. I tried to pay this debt back by teaching with dedication and enthusiasm.

[1] Although a similar thing could be said about many optimisation problems.

# 1 Gap problems

Some of the results in this thesis are inapproximability proofs. When working on hardness of approximation, we introduce and reduce between *gap problems*, in which the algorithm is promised that the incoming instance will come from one of two sets separated by a large gap in a specific measure. A good approximation algorithm with regard to this measure would need to distinguish between instances coming from these sets. In this chapter we present a selection of important gap problems, known results and conjectures. They will serve as starting points in our reductions in the next chapters.

## 1. Label Cover

One problem that has turned out to be useful in the hardness of approximation is Label Cover, in which we colour (label) the vertices of a bipartite graph, attempting to maximise the number of satisfied edge constraints. The constraints are functions projecting the colour of the left vertex onto the right vertex. Hence this problem is sometimes called *projection games*[1].

**Problem 1** (Label Cover, $\mathsf{LC}(\delta, t, R)$)
Given a bipartite, bi-regular graph $G = (U \uplus V, E)$ with right degree $t$, two sets of labels $[L]$, $[R]$, and for each edge $e \in E$ a function *constraint* $\pi_e \colon [L] \to [R]$, distinguish between two cases:

- (YES) There exists a labelling $l \colon U \to [L]$, such that every right-side vertex $v$ is *satisfied*—i.e. for every two neighbours $u_1$, $u_2$ of $v$ we have $\pi_{(u_1,v)}(l(u_1)) = \pi_{(u_2,v)}(l(u_2))$.

[1] The *game* here is played between two *provers* and one *verifier*: The verifier samples a right-vertex and asks the provers to colour its random two neighbours. The provers are not allowed to communicate and do not know, which right-vertex had been selected, so if their colours agree, the verifier is satisfied they likely can colour the graph to satisfy large fraction of the constraints.

- (NO) For every labelling $l\colon U \to [L]$ there are at most $\delta|V|$ *weakly satisfied* right-side vertices $v$—i.e. there exist two distinct neighbours $u_1$, $u_2$ for which $\pi_{(u_1,v)}(l(u_1)) = \pi_{(u_2,v)}(l(u_2))$.

If we denote by $\mathrm{val}(\mathcal{L})$ the maximum fraction of right-side vertices satisfied by any labelling in a Label Cover instance $\mathcal{L}$, and by weak-val($\mathcal{L}$) the maximum fraction of weakly satisfied right-side vertices, we see that the task in $\mathrm{LC}(\delta, t, R)$ is to decide if $\mathrm{val}(\mathcal{L}) = 1$ or weak-val$(\mathcal{L}) < \delta$. The instances coming from the (YES) and (NO) sets are thus separated by a large gap in the val (and even weak-val) measure.

Two results concerning hardness of Label Cover will be relevant to our work. The first is now a classical theorem:

**Theorem 2** (Feige [Fei96])
*For every $\delta > 0$, every $t \in \mathbb{N}$, $t \geqslant 2$ exists sufficiently large $R \in \mathbb{N}$ such, that $\mathrm{LC}(\delta, t, R)$ is* **NP**-*hard.*

The second theorem is conditional—it relies on a conjecture called Gap-ETH[2]:

**Theorem 3** (Manurangsi [Man20])
*Assuming Gap-ETH, for every $\delta > 0$, $t \in \mathbb{N}$ $(t \geqslant 2)$, exists sufficiently large $R \in \mathbb{N}$, such that no $f(|U|) \cdot |V|^{o(|U|)}$-time algorithm can solve $\mathrm{LC}(\delta, t, R)$.*

## 2. Unique Games and variations

When reducing from Label Cover instances, it would sometimes be useful to constraint the input instance somehow without losing the hardness. For example, controlling the (at least one sided) degree of the bipartite graph may play a role, and we can see from the aforementioned theorems that one indeed may do that. Another idea, introduced and used by Khot, was to assume that the constraint functions are bijections (permutations). That has led to Unique Label Cover.

**Problem 4** (Unique Label Cover, $\mathrm{ULC}(\epsilon, t, R)$)
Given a bipartite graph $G = (U \cup V, E)$ with right-degree $t$, a set of labels $[R]$, and for each $e \in E$ a permutation $\pi_e\colon [R] \to [R]$ distinguish between two cases:

[2] Recently, significant effort has been directed at the question of approximability within parametrised complexity (rather than in polynomial time). Most of the results in that area are conditional on various assumptions. Gap-ETH [Din16; MR17] is one such conjecture, which has given a lot of results (e.g. [Cha+17]). Other assumptions from **FPT** world, like ETH or **FPT** $\neq$ **W**[1] are also used, but tend to be much harder to reduce from.

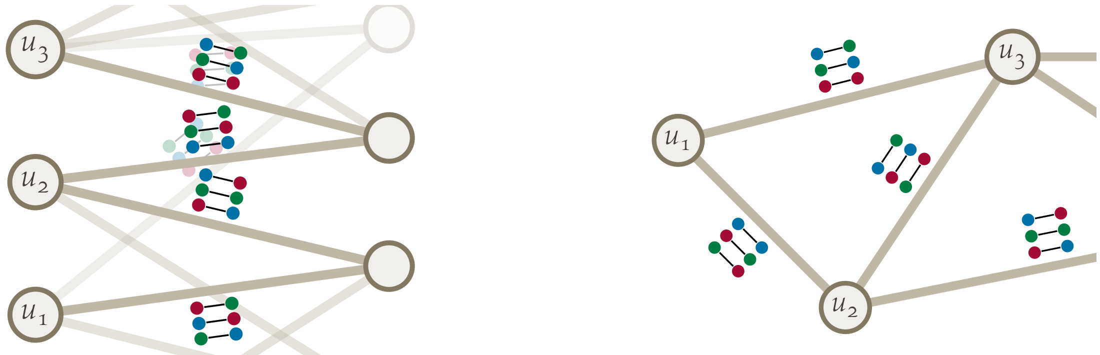



Figure 1: Unique Label Cover can be translated from one variant to another with a marginal loss in $\epsilon$.

- (YES) There exists a subset $V' \subset V$ of size $|V'| \geqslant (1-\epsilon)|V|$ and a labelling $l\colon U \to [R]$ such that for every vertex $v \in V'$ and its two neighbours $u_1$, $u_2$, $\pi_{(u_1,v)}(l(u_1)) = \pi_{(u_2,v)}(l(u_2))$.
- (NO) For every labelling $l$ and any subset $V' \subset V$ of size $|V'| \geqslant \epsilon|V|$, there is a vertex $v \in V'$ such that for any two neighbours $u_1$, $u_2$ of $v$, $\pi_{(u_1,v)}(l(u_1)) \neq \pi_{(u_2,v)}(l(u_2))$.

There is a slight difference here compared to the definition of Label Cover—in the (YES) case we do not have $\mathrm{val}(\mathcal{L}) = 1$, since checking if every edge-constraint can be satisfied in Unique Label Cover is very easy.

Complexity of the Unique Label Cover is still unresolved, but assuming it to be **NP**-hard has brought a ton of interesting consequences. This assumption is called *Unique Games Conjecture* [Kho02].

**Conjecture 5** (UGC [Kho02]). *For every $\epsilon > 0$ and $t \geqslant 2$ exists $R$ such that* $\mathsf{ULC}(\epsilon, t, R)$ *is* **NP**-*hard*.

It must be noted that these conjectures and theorems are quite robust. In ULC we could formulate the gap in terms of a fraction of satisfied constraints rather than (weakly) satisfied right-vertices and the conjecture remains equivalent. We could even allow the graph to be non-bipartite (Fig 1). However, some assumptions on a structure of the input graph have led to conjectures that are not yet known to be equivalent to UGC. One such extension of UGC, which we will be relying on, is the so-called *Strong Unique Games Conjecture*[3]. It is a strengthening of UGC, where in the (NO) case the graph $G$ is also promised to be a 'small set expander'; a precise definition is given below.

[3]Originally, the name Strong UGC was used to refer to Conjecture 5 itself [KR03], as it is described

**Problem 6** (ULC with weak expansion, $\mathsf{ULCE}(\epsilon, R, \delta)$)
Given a bipartite graph $G = (U \cup V, E)$, a set of labels $[R]$, and for each $e \in E$ a permutation $\pi_e \colon [R] \to [R]$ distinguish between two cases:

- (YES) There exists a subset $V' \subset V$ of size $|V'| \geqslant (1-\epsilon)|V|$ and a labelling $l : U \to [R]$ that satisfies every vertex $v \in V'$.
- (NO) For every labelling $l$ the number of weakly-satisfied right edges is smaller than $\epsilon|V|$. **Moreover**, for every $S \subset V$ of size $|S| = \delta|V|$, $|\Gamma(S)| \geqslant (1-\delta)|V|$, where $\Gamma(S)$ is a neighbourhood of the set $S$ in the graph $G$.

**Conjecture 7** (Strong UGC [BK09]). *For every $\epsilon > 0$ and $\delta > 0$ exists $R \in \mathbb{N}$ such that $\mathsf{ULCE}(\epsilon, R, \delta)$ is* **NP**-*hard.*

Another conjecture we base our results on is the *Small Set Expansion Hypothesis* (SSEH), which concerns the hardness of finding well-separated parts of a graph. Formally, in a $d$-regular graph $G = (V, E)$, an edge expansion of a subset $S$ of vertices is defined as

$$\Phi(S) = \frac{|E(S, V \setminus S)|}{d \cdot \min\{|S|, |V \setminus S|\}},$$

where $E(S, V \setminus S)$ is a set of edges crossing the cut $(S, V \setminus S)$. The small set expansion problem parametrised by $\delta, \eta \in (0,1)$ is defined as follows.

**Problem 8** ($\mathsf{SSE}(\delta, \eta)$)
Given a regular graph $G = (V, E)$ distinguish between two cases:

- (YES) There exists $S \subset V$ of size $|S| = \delta|V|$ such that $\Phi(S) \leqslant \eta$.
- (NO) For every $S \subset V$ of size $|S| = \delta|V|$, $\Phi(S) \geqslant 1 - \eta$.

The SSEH, introduced by Raghavendra and Steurer [RS10], hypothesises about hardness of this problem.

**Conjecture 9** (SSEH). *For every $\eta > 0$ exists $\delta = \delta(\eta) \in (0, \frac{1}{2}]$ such that $\mathsf{SSE}(\delta, \eta)$ is* **NP**-*hard.*

It turns out, surprisingly, that SSEH is also stronger than UGC [RS10]. We will see the intuition behind this fact in the next chapter. No direct relationship is however known between Strong-UGC and SSEH.

---

here. In this thesis we follow the nomenclature of [Bha+16], which refers to Conjecture 7 as Strong UGC

# 2 The Minimum Maximal Matching problem

An *edge dominating set* of an undirected graph is a subset $D \subset E$ of edges, such that every edge must either be in $D$ or adjacent to some edge in $D$. A Minimum Edge Dominating Set (EDS) problem asks to find such a set of smallest cardinality. An *independent* edge dominating set, in which no two edges have a common vertex, is also called a *maximal matching*—the domination criterion means that no edge may be added to it while maintaining the independence. In a natural algorithmic question of Minimum Maximal Matching (MMM) we are asked to find such an independent edge dominating set of smallest cardinality.

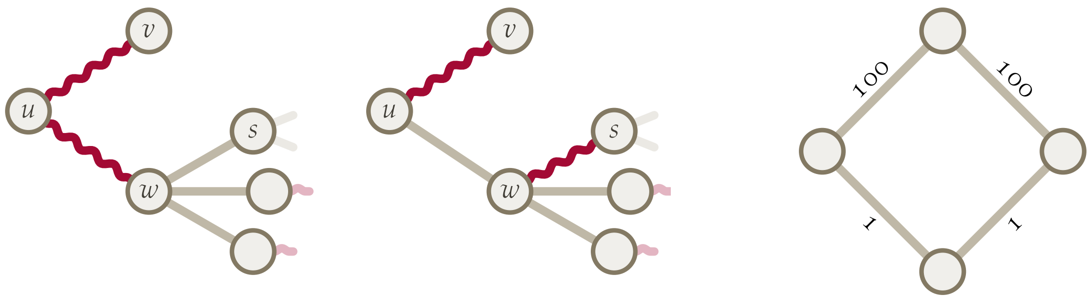
Both these problems also have counterparts in graphs with weights on edges, wEDS and wMMM, where we aim to find the lightest (independent) edge dominating set rather than smallest.

In this chapter we are going to present four inapproximability results which appeared in two papers [DLM19; DMM20], however for the first of this results we will present a different (simpler) proof then the published one.

## 1. Related work

The first traceable discussion of our problems goes back to 1969, when it was shown that every inclusion-wise minimal edge dominating set can be transformed into a maximal matching of the same cardinality (see Fig. 1 (a)). The EDS and MMM problems (in unweighted variants) are therefore equivalent [Gup69].

**Algorithms and Complexity** A series of polynomial-time algorithms for MMM on selected families of graphs have been developed: for trees [MH77; YG80], claw-free chordal graphs, locally connected claw-free graphs, the line graphs of total graphs, the line graphs of chordal graphs [HK93], bipartite permutation graphs, co-triangulated graphs [Sri+95]. For another families of graphs the problem has been shown to be




(a) Every *Edge Dominating Set* can be transformed into a Maximal Matching of the same size: Let $(u,v)$, $(u,w)$ be two adjacent EDS-edges. We find a neighbour $s$ of $w$ with no incident EDS-edge and replace $(u,w)$ with $(w,s)$. If no such $s$ exists, the edge $(u,w)$ can be dropped from the EDS.

(b) The equivalence between MMM and EDS does not hold in the weighted graphs. The lightest maximum matching weighs 101, while the Minimum Edge Dominating Set weighs only 2.

Figure 1: The equivalence of MMM and EDS.

**NP**-hard: first by Yannakakis and Gavril on planar or bipartite graphs of maximum degree 3 [YG80]; then planar bipartite graphs, line and total graphs, perfect claw-free graphs, and planar cubic graphs [HK93], and many other.

**Approximation algorithms** By endpoint-counting, no matching can be more than twice as large as any other maximal matching. This gives an immediate 2-approximation algorithm for MMM: find any maximal matching. Attempts to beat this approximation ratio resulted in $2 - c\frac{\log|V|}{|V|}$ [GLR08] (where $c$ is an arbitrary constant), and $2 - \frac{1}{\chi'(G)}$ ($\chi'(G)$ is an edge-colouring number of $G$ and is bounded by maximum degree plus one) [MKI11].

EDS was one of problems exemplifying Baker's framework for constructing PTAS on planar graphs [Bak83]. Better-than-two constant approximation ratios has been achieved for dense graphs [SV12], as well as for cubic graphs [Bat20].

**Weighted variants** The equivalence between EDS and MMM does not hold when edges carry weights (see example in Fig. 1 (b)). All of the aforementioned approximation algorithms (with the exception of Baker's method, but including the trivial 2-approximation) do not apply for wEDS and wMMM. A series of papers by Fujito, Parekh and others concluded in a 2-approximation algorithm for wEDS [Car+00; FN02; Par02]. wEDS has been shown to be as hard to approximate as weighted Vertex Cover [FN02;

Par02], hence, realistically, no further progress should be expected. Surprisingly, wMMM is impossible to approximate within any polynomially computable factor [Fuj01][1].

**Negative results** Chlebík and Chlebíkova proved that it is impossible to approximate the unweighted variant of EDS (MMM) with any constant ratio better then $\frac{7}{6}$ unless $\mathbf{P} = \mathbf{NP}$ [CC06]. That bound was later improved to 1.18 [Esc+15]. Additionally, a simple $\frac{3}{2}$-hardness conditional on *Unique Games Conjecture* were known by a straightforward reduction from Vertex Cover.

We enter the story with our results, by showing that the Unique Games Conjecture implies that the MMM cannot be approximated within a constant factor better then 2.

**Theorem 1**
*Assuming* Unique Games Conjecture *(Conjecture 1.5), there is no polynomial-time algorithm that approximates MMM (or equivalently EDS) on unweighted graphs within factor $2 - \varepsilon$ for any constant $\varepsilon > 0$, unless* $\mathbf{P} = \mathbf{NP}$.

A special case of MMM, particularly natural for matching experts is, when the input graph is bipartite. We were not able to carry our negative bound directly to the bipartite graphs, but we managed to obtain three 'weaker' results.

**Theorem 2**
*Assuming* Unique Games Conjecture *(Conjecture 1.5), there is no polynomial-time algorithm that approximates MMM (or equivalently EDS) on unweighted, bipartite graphs within factor $\frac{4}{3} - \varepsilon$ for any constant $\varepsilon > 0$, unless* $\mathbf{P} = \mathbf{NP}$.

**Theorem 3**
*Assuming* Strong Unique Games Conjecture *(Conjecture 1.7), there is no polynomial-time algorithm that approximates MMM (or equivalently EDS) on unweighted, bipartite graphs within factor $2 - \varepsilon$ for any constant $\varepsilon > 0$, unless* $\mathbf{P} = \mathbf{NP}$.

**Theorem 4**
*Assuming* Small Set Expansion Hypothesis *(Conjecture 1.9), there is no polynomial-time algorithm that approximates MMM (or equivalently EDS) on unweighted, bipartite graphs within factor $2 - \varepsilon$ for any constant $\varepsilon > 0$, unless* $\mathbf{P} = \mathbf{NP}$.

The rest of this chapter will be devoted to presenting the proofs of these theorems.

[1] A fact we were unaware of when working on our results. But it makes the unweighted MMM all the more intriguing.

## 2. Our method

Our proofs take advantage of a close relationship between the INDEPENDENT SET problem and MMM. In short, the set of vertices left unmatched by a maximal matching must form an independent set. In [KR03] it was proven that it is **NP**-hard (under UGC) to distinguish (YES) graphs with independent set of size $(\frac{1}{2} - \epsilon)|V|$ from (NO) those with no independent set of size $\epsilon|V|$. In the proof of Theorem 1 presented in our paper [DLM19] this reduction was modified by only adding edges, so that in the (YES) case there was a perfect matching between vertices not included in the independent set. This strategy aimed for simplicity, as typically, when proving correctness of gap reductions, analysing the (NO) case is significantly more difficult than the (YES) case. When we only add edges, no independent set can be created, so there is no need to look at the (NO) part at all. In the proof of Theorem 1 in this thesis we do the same trick, just starting from a different construction for Vertex Cover [BK09], which will have an advantage of producing unweighted graphs right away (in the paper [DLM19] we had to exploit the structure of the resulting weighted graph to get rid of the weights).

In bipartite graphs, we are aiming to employ the same strategy. However, while maximum independent sets in bipartite graphs can be found in polynomial time, we notice that vertices left out by any maximal matching must form a *balanced bi-independent set*—an independent set with the same number of vertices on both sides of bipartition. We must hence look at existing hardness of approximation proofs for balanced bi-independent set [Bha+16; Man17][2] and carefully add edges to them.

## 3. General graphs

At the heart of many UGC-based reductions lies a *dictatorship test*. A dictatorship is a function $f : \{0,1\}^R \to \{0,1\}$ that depends only on one coordinate, so $f(x) = x_i$ or $f(x) = \overline{x_i}$. Dictatorships are an error-correcting code—there are $2^{2^R}$ functions, but only $2R$ dictatorships. Importantly, there are ways to only query $f$ on some inputs and decide whether it is (similar to) a dictatorship, and decode it if it is. In an inapproximability reduction we are given $t$ functions $f^{(1)}, \ldots, f^{(t)}$ and must test if they all are dictatorships of the same coordinate.

The test $F_{\epsilon,t}$ coined by Bansal and Khot [BK09] works as follows:

1. Pick a random input $x \in \{0,1\}^R$;

[2] These results talk about *Balanced Biclique* which is more natural, but since bi-independent set fits our needs more, we will silently take graph complement when recalling their reductions.

2. Pick a random $S \subset [R]$ of size $\epsilon R$. Define the sub-cubes:

$$C_{x,S} = \left\{ z \in \{0,1\}^R \mid z_j = x_j \ \forall j \notin S \right\},$$
$$C_{\bar{x},S} = \left\{ z \in \{0,1\}^R \mid z_j = \overline{x_j} \ \forall j \notin S \right\};$$

3. Accept if all $f^{(l)}$-s are equal to some $b$ on all inputs in $C_{x,S}$ and $\bar{b}$ on all inputs in $C_{\bar{x},S}$.

**Completeness** If all the $f^{(l)}$-s are the dictatorship of the same coordinate $i$, this coordinate will be included in the set $S$ with probability $\epsilon$. When $i \notin S$, the test will pass.

**Soundness** For any function $f : \{0,1\}^R \to \{0,1\}$ and any integer $s \leqslant R$ one can determine the set $L_s[f] \subset [R]$ of $s$ *most low-degree influential coordinates*[3]. Bansal and Khot prove the following

**Theorem 5** ([BK09, Theorem 6.5, simplified])
*For every $\epsilon, \delta > 0$ there are $t, s \in \mathbb{N}$ such, that if the test $F_{\epsilon,t}$ accepts a collection $f^{(1)}, \ldots, f^{(t)}$ of functions $\{0,1\}^R \to \{0,1\}$ (for any $R$) with probability larger than $\delta$, then there are two of the tested functions $f^{(l_1)}$, $f^{(l_2)}$ and a coordinate $i$ for which*

$$i \in L_s[f^{(l_1)}] \cap L_s[f^{(l_2)}].$$

## 3.1. The vertex cover instance

We now use the test $F_{\epsilon,t}$ to translate a Unique Label Cover instance $G = (U \uplus V, E, \{\pi_e\}_{e \in E})$ into a graph $H$. Intuitively, every vertex of $H$ will correspond to the $V$-vertex performing the dictatorship test on the colouring of its $t$ neighbours. For every such vertex there are $2^{(1-\epsilon)R}\binom{R}{\epsilon R}$ possible sub-cubes $C_{x,S}$, and 2 possible values of the bit $b$. Every edge will correspond to inconsistent runs of the test.

**Reduction 6**
Input: A ULC$(\epsilon, t, R)$ instance $G = (U \uplus V, E, \{\pi_e\}_{e \in E})$.
Output: An undirected graph $H = (V_H, E_H)$ with $V_H = V \times 2^{(1-\epsilon)R}\binom{R}{\epsilon R} \times \{0,1\}$.
For any left-side vertex $u \in U$ and its two (not necessarily distinct) neighbours

[3] We will not delve into *how* this set is defined, as that would require us to explore the topic of Fourier analysis of boolean functions, which we managed to avoid in our proofs. Luckily, we do not need that to understand how dictatorship test translates into reductions.

$v_1, v_2 \in V$ we add an edge in $H$ between $\left\langle v_1, C_{x_1,S_1}, b_1 \right\rangle$ and $\left\langle v_2, C_{x_2,S_2}, b_2 \right\rangle$ if

- $b_1 \neq b_2$ and exists $x \in \{0,1\}^R$ such that $x \circ \pi_{(u,v_1)} \in C_{x_1,S_1}$[4] and $x \circ \pi_{(u,v_2)} \in C_{x_2,S_2}$; or
- $b_1 = b_2$ and exists $x \in \{0,1\}^R$ such that $\bar{x} \circ \pi_{(u,v_1)} \in C_{x_1,S_1}$ and $x \circ \pi_{(u,v_2)} \in C_{x_2,S_2}$.

Two properties are proved:

**Lemma 7** (Completeness [BK09, Section 4.1]). *If $G$ was a (YES) instance of* ULC*, there is an independent set of size $(\frac{1}{2} - 2\epsilon)|V_H|$ in $H$.*

*Proof.* Take a labelling $l\colon U \to R$, as in ULC. $l$ naturally extends to the set $V'$ of satisfied right vertices. Define

$$\mathcal{IS} = \left\{ \langle v, C_{x,S}, b \rangle \mid v \in V', l(v) \notin S, x_{l(v)} = b \right\}.$$

To see that this is indeed an independent set, define functions $\left\{ f_u : x \mapsto x_{l(u)} \right\}_{u \in U}$. Every vertex in $\mathcal{IS}$ corresponds to an accepting run of $F_{\epsilon,t}$ on $t$ of these functions. □

One can see, that when defining $\mathcal{IS}$ we could just as well have written $x_{l(v)} \neq b$ instead of $x_{l(v)} = b$, which would have corresponded to $\left\{ f_u : x \mapsto \overline{x_{l(u)}} \right\}_{u \in U}$. There are in fact two independent sets of size $(1 - 2\epsilon)|V_H|$ in the (YES) case. The freedom of picking one of the two sets is what the 'free bit' from the title of [BK09] refers to.

**Lemma 8** (Soundness [BK09, Section 4.2]). *If $G$ was a (NO) instance of* ULC*, no independent set of size $\epsilon|H|$ exists in $H$.*

*Proof (sketch).* Assume contrapositive, that there is an independent set $\mathcal{S}$ in $H$ of size $2\delta|H|$. We will use this assumption to colour the vertices of ULC instance $G$: Imagine that for every $u \in U$ there is a boolean function $f^{(u)}\colon \{0,1\}^R \to \{0,1\}$. For each such function we can identify the set $L_s[f^{(u)}]$ and pick one of the $s$ colours at random.

For $v \in V$, let

$$V_H[v] = \left\{ \langle v', C_{x,S}, b \rangle \in V_H \mid v' = v \right\}.$$

By Markov's inequality, the set $V'$ of right vertices $v$ for which $|V_H[v] \cap \mathcal{S}| \geqslant \delta |V_H[v]|$ has size at least $\frac{1}{2}|V|$. Take any $v \in V'$ and its $t$ neighbours, $u_1, \ldots, u^t$. Functions $f^{(u_1)} \circ \pi_{(u_1,v)}, \ldots, f^{(u_t)} \circ \pi_{(u_t,v)}$ pass the dictatorship test $F_{\epsilon,t}$ with probability exceeding

[4] $x \circ \pi_{(u,v)}$ is the vector $x$ with coordinates permuted by permutation $\pi_{(u,v)}$. Since the coordinates correspond to colours, we can think that this permutation allows us to look at $x$ from the perspective of vertex $v$.

$\delta$, so by Theorem 5 there are two neighbours $w$, $w'$ such that $L_s[f^{(w)}] \cap L_s[f^{(w')}] \neq \emptyset$. Consequently, our random colouring weakly satisfies at least $\frac{1}{s^2}|V'| \geqslant \frac{2}{s^2}|V|$ right vertices of $G$. □

Since Independent Set and Vertex Cover are complements in any graph, the reduction gives us that it is **NP**-hard to distinguish between graphs with Minimum Vertex Cover of size $(\frac{1}{2}+\epsilon)|V_H|$ and $(1-\epsilon)|V_H|$ for any $\epsilon$.

## 3.2. Supplementing $H$ with edges

We wish to exhibit a similar gap for `MMM`:

**Problem 9** (`Gap-MMM`)
Given an undirected graph $H = (V_H, E_H)$ distinguish between two cases:

- (YES) There is a maximal matching of size $\frac{1}{2}(\frac{1}{2}-\epsilon)|V_H|$.
- (NO) There is no independent set of size $\epsilon|V_H|$.

In the (NO) case, when there are no independent sets of size $\epsilon|V_H|$, every maximal matching must match almost all of the vertices.

To show that $H$ is an instance of `Gap-MMM`, in the (YES) case we would need to find a matching between vertices of $V_H \setminus \mathcal{IS}$. This is however impossible, because—as we have said—inside $V_H \setminus \mathcal{IS}$ there is another independent set of size $|\mathcal{IS}|$. We thus need to amend our reduction and break this symmetry.

**Reduction 10**
Input: A `ULC`$(\epsilon, t, R)$ instance $G = (U \cup V, E, \{\pi_e\}_{e \in E})$.
Output: An undirected graph $H' = (V_{H'}, E_{H'})$ with $V_{H'} = V \times 2^{(1-\epsilon)R}\binom{R}{\epsilon R} \times \{0,1\}$.

For any left-side vertex $u \in U$ and its two (not necessarily distinct) neighbours $v_1, v_2 \in V$ we add an edge in $H'$ between $\left\langle v_1, C_{x_1,S_1}, b_1 \right\rangle$ and $\left\langle v_2, C_{x_2,S_2}, b_2 \right\rangle$ if exist $y_1, y_2 \in \{0,1\}^R$ such that

- $\overline{b_1 \oplus y_1(i)} \leqslant b_2 \oplus y_2(i)$ for all $i \notin S_1 \cup S_2$[5]; and
- $b_1 \oplus (y_1 \circ \pi_{(u,v_1)}) \in C_{x_1,S_1}$ and $b_2 \oplus (y_2 \circ \pi_{(u,v_2)}) \in C_{x_2,S_2}$.

We only added edges to the graph, compared to Reduction 6, so Lemma 8 is (even more) true for $H'$. We need to prove that in the (YES) case a large independent set still exists.

[5]This is a slight modification of Reduction 6, where this condition could be written as $\overline{b_1 \oplus y_1(i)} = b_2 \oplus y_2(i)$ for all $i \notin S_1 \cup S_2$. In our reduction we replace $=$ with $\leqslant$.

**Lemma 11.** *If $G$ was a (YES) instance of* ULC*, $H'$ has an independent set of size $(\frac{1}{2} - 2\epsilon)|V_{H'}|$.*

*Proof.* Take $\mathcal{IS}$ as defined in Lemma 7. For any $\langle v, C_{x,S}, b\rangle \in \mathcal{IS}$ and $(u,v) \in E$, if we have a vector $y \in \{0,1\}^R$ such that $b \oplus (y \circ \pi_{(u,v)}) \in C_{x,S}$ then $y(l(u)) = 0$ (since $x(\pi_{(u,v)}(l(u))) = b$). As $\overline{0} = 1 \not\leqslant 0$, no edge exists in $E_{H'}$ between two vertices in $\mathcal{IS}$. □

We would not have been able to prove the above lemma for the alternative definition of $\mathcal{IS}$ in Lemma 7, because $\overline{1} = 0 \leqslant 1$. In fact, the symmetry is broken—there are now many internal edges in $V_{H'} \setminus \mathcal{IS}$.

**Lemma 12.** *There is a perfect matching in $V_{H'} \setminus \mathcal{IS}$.*

*Proof.* We consider three groups of vertices $\langle v, C_{x,S}, b\rangle \in V_{H'} \setminus \mathcal{IS}$.

(G1) $v \in V'$, $l(v) \notin S$, $x_{l(v)} \neq b$: We pair such a vertex with $\langle v, C_{x',S}, b\rangle$, where

$$x'_i = \begin{cases} x_i = \overline{b} & \text{if } i = l(v), \\ \overline{x_i} & \text{otherwise.} \end{cases}$$

This edge exists in $E_{H'}$ because $\overline{1} = 0 \leqslant 1$.

(G2) $v \in V'$, $l(v) \in S$: We pair such a vertex with $\langle v, C_{\overline{x},S}, b\rangle$.

(G3) $v \notin V'$: We also pair such a vertex with $\langle v, C_{\overline{x},S}, b\rangle$.

□

This concludes the analysis of the (YES) case and together with Lemma 8 hands us that Gap-MMM is **NP**-hard (assuming UGC), which proves Theorem 1.

## 4. Bipartite graphs I: Bipartisation and path covers

In this section we will show that:

**Lemma 13.** *Assuming the Unique Games Conjecture, for any $\epsilon > 0$ it is* **NP***-hard to distinguish between balanced bipartite graphs of $2n$ vertices:*

- *(YES) with a maximal matching of size $n(\frac{1}{2} - \epsilon)$.*
- *(NO) with no maximal matching of size smaller then $n(\frac{2}{3} - \epsilon)$.*

Theorem 2 is a direct consequence of this lemma.

Our reduction is going to be very predictable—we start with graph $G$-an instance of `Gap-MMM` and produce two copies of each vertex $v$: $v^{(l)}$ and $v^{(r)}$. The vertices $u^{(l)}$ and $v^{(r)}$ will be connected in the resulting graph $H$ whenever $(u,v) \in E(G)$. It is easy to see that for every maximal matching $M \subset E(G)$ there is a maximal matching in $H$ of size $2|M|$—we take two edges corresponding to each $e \in M$.

## 4.1. Covering with paths

In order to analyse the (NO) case we need an alternative interpretation of graph $H$ and its matchings from the perspective of $G$. Let us think that the graph $G$ is bi-directed and an edge between $u^{(l)}$ and $v^{(r)}$ corresponds to an arc $(u,v)$ in $G$. For any matching $M$ in $H$, the corresponding set of arcs in $G$ forms directed paths and cycles (no vertex has more than one outgoing and one incoming arc). For a matching $M \subset E(H)$, $P(M)$ will be the set of these paths and cycles in $G$.

*Observation.* If $M$ is a maximal matching in $H$ then every $p \in P(M)$ has length $|p| \geqslant 2$.

*Proof.* Take any $p = (u,v) \in P(M)$ of length 1. Vertices $v^{(l)}$ and $u^{(r)}$ are unmatched in $M$ and are connected by an edge in $H$ (corresponding to the reverse arc $(v,u)$) which contradicts maximality of $M$. □

This observation will help us tie maximal matchings in $H$ and vertex covers in $G$.

**Lemma 14.** *For any maximal matching $M$ in $H$, exists a vertex cover $C$ in $G$ of size $|C| \leqslant \frac{3}{2}|M|$.*

*Proof.* For every $p \in P(M)$ we add $V(p)$ to $C$. When $p$ is a cycle, we add $|p|$ vertices. For a path we add at least $\frac{3}{2}|p|$ (the longer the path, the higher the ratio). $V(M)$ is a vertex cover of $H$, so $C = V(P(M))$ is a vertex cover in $G$. □

In the (NO) case graph $G$ had no vertex cover smaller than $(1-\epsilon)|V(G)|$, so there is no maximal matching of size smaller then $|V(G)|(1-\epsilon)\frac{2}{3}$.

# 5. Bipartite graphs II: From Strong UGC to MMM

In this section we will review the reduction from Unique Label Cover with Weak Expansion to Bi-Independent Set problem, that was presented in [Bha+16]. Then we will present our modification and prove its correctness. The main building block of their reduction is the graph $T$ that we are going to use as a gadget. It will be composed of identical layers and parametrised by a prime number $q > 2$ (see Fig 2).

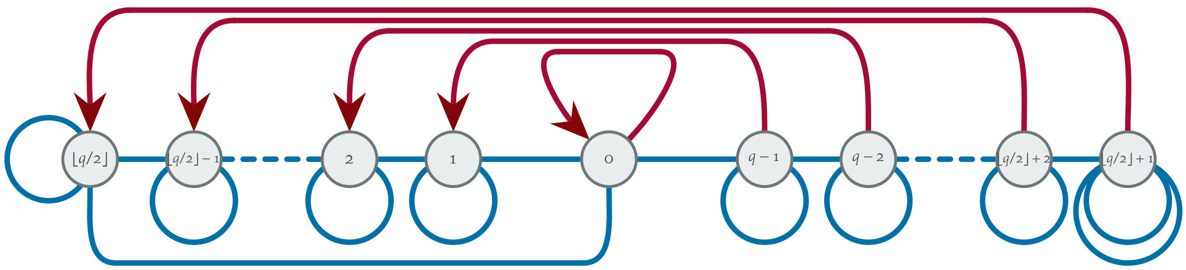



Figure 2: The gadget $T_q$ for a prime number $q > 2$ (blue) is an undirected 3-regular graph (counting edges, not endpoints). The modified gadget $T_q'$ is obtained by adding directed arcs from $i$ to $q - i$ for $i \notin \mathfrak{Lo}_q$ (red).

**Definition 15** ($T_q$). The vertex set is $\mathbb{Z}_q = \{0, 1, \ldots, q-1\}$. There are three kinds of (undirected) edges.

- $\{i, i+1 \pmod{q}\}$ for $i \neq \lfloor q/2 \rfloor$.
- A self-loop $\{i, i\}$ for $i \neq 0$ and an additional one for $\lfloor q/2 \rfloor + 1$.
- An edge $\{0, \lfloor q/2 \rfloor\}$.

The resulting graph is 3-regular and connected.

Let us additionally define the sets $\mathfrak{Hi}_q = \{\lfloor q/2 \rfloor + 1, \ldots, q-1\}$ and $\mathfrak{Lo}_q = \{1, \ldots, \lfloor q/2 \rfloor\}$. 0 belongs to neither. As one can see, there are no edges in $T_q$ between $\mathfrak{Hi}_q$ and $\mathfrak{Lo}_q$, so they form a balanced bi-independent set (even though $T_q$ is not bipartite). The graph $T = T_q^{\otimes R}$ has the vertex set $\mathbb{Z}_q^R$ (so every vertex is an R-element vector of numbers from $\mathbb{Z}_q$, every coordinate corresponds to one label colour in Unique Label Cover) and two vertices $x$ and $y$ are connected, if $x_i$ and $y_i$ are connected in $T_q$ for every coordinate $i \in R$. The reduction from [Bha+16] is now defined as follows.

**Reduction 16**
Input: An $\mathsf{ULCE}(\epsilon, R, \delta)$ instance $G = \langle U_G \uplus V_G, E_G, \{\pi_e\}_{e \in E} \rangle$.
Output: A bipartite graph $H = \langle U_H \uplus V_H, E_H \rangle$.

Let $U_H = V_H = V_G \times \mathbb{Z}_q^R$, so every vertex in $H$ corresponds to one vertex in $V_G$ and $R$ numbers in $\mathbb{Z}_q$. For any $u \in U_G$ and its two neighbours $v_1, v_2$ in $G$, we add an edge between $(v_1, x) \in U_H$ and $(v_2, y) \in V_H$ in $E_H$ whenever $x \circ \pi^{-1}_{(u,v_1)}$ is connected to $y \circ \pi^{-1}_{(u,v_2)}$ in $T$ ($x$ and $y$'s coordinates are permuted to the point of view of vertex $u$).

It is easy to see that if the input graph came from the (YES) case, then there is a bi-independent set of size $(\frac{1}{2} - \epsilon - \frac{1}{q})|U_H \uplus V_H|$ in $H$. To find it pick the labelling $l$ and the set $V' \subset V_G$ of vertices fully satisfied by the labelling $l$. For any $v \in V'$ and $x$ such that $x(l(v)) \in \mathfrak{Lo}_q$ pick $(v,x)$ into the left side of the bi-independent set. Similarly, pick $(v,y) \in V_H$ if $y(l(v)) \in \mathfrak{Hi}_q$. The reader can see, that in fact two symmetrical bi-independent sets exist in $H$. The (NO) case is analysed in the original paper.

**Lemma 17** (soundness, [Bha+16, Lemma 14 reinterpreted]). *For every constant $\gamma$, there exist constants $\epsilon, \delta$ such that if $G$ came from the NO case of* ULCE$(\epsilon, R, \delta)$ *then $H$ has no balanced bi-independent set larger than $\epsilon|U_H \uplus V_H|$.*

Now it is time to introduce our modification. Let us define a new gadget $T'_q$ by adding directed arcs $\{(i, q-i)\}_{i \notin \mathfrak{Lo}_q}$ (see Fig. 2). The undirected edge between two vertices is replaced by a pair of two directed arcs, the self-loop is replaced by a single arc. Consequently, the graph $T' = T_q'^{\otimes R}$ is also directed. Our reduction is similar to the one above.

**Reduction 18**
Input: An ULCE$(\epsilon, R, \delta)$ instance $G = \langle U_G \uplus V_G, E_G, \{\pi_e\}_{e \in E} \rangle$.
Output: A bipartite graph $H' = \left\langle U_H \uplus V_H, E'_H \right\rangle$.

Let $U_H = V_H = V_G \times \mathbb{Z}_q^R$, like before. For any $u \in U_G$ and its two neighbours $v_1, v_2$ in $G$, we add an edge between $(v_1, x) \in U_H$ and $(v_2, y) \in V_H$ in $E_H$ whenever there is an arc from $x \circ \pi^{-1}_{(u,v_1)}$ to $y \circ \pi^{-1}_{(u,v_2)}$ in $T'$.

**Lemma 19** (completeness). *If the graph $G$ comes from the (YES) case of* ULCE, *there is a bi-independent set $\mathcal{BIS} \subset U_H \uplus V_H$ of size $|\mathcal{BIS}| \geqslant (\frac{1}{2} - \epsilon - \frac{1}{q})|U_H \uplus V_H|$ in $H'$. Moreover, there is a perfect matching in $H'$ between $U_H \setminus \mathcal{BIS}$ and $V_H \setminus \mathcal{BIS}$.*

*Proof.* Take $l$ and $V'$ as in Problem 1.6. For every $v \in V'$ and $x \in \mathbb{Z}_q^R$ such that $x(l(v)) \in \mathfrak{Lo}_q$, we pick the vertex $(v,x) \in U_H$ into the set $\mathcal{BIS}$. Similarly, for every $v \in V'$ and $y$ with $y(l(v)) \in \mathfrak{Hi}_q$, we pick $(v,y) \in V_H$ into $\mathcal{BIS}$. To see that $\mathcal{BIS}$ is indeed a bi-independent set pick any $(v_1, x) \in U_H \cap \mathcal{BIS}$ and $(v_2, y) \in V_H \cap \mathcal{BIS}$. Since $v_1, v_2 \in V'$, for every $u$ in their common neighbourhood in $G$, $\pi_{(u,v_1)}(l(u)) = l(v_1)$ and $\pi_{(u,v_2)}(l(u)) = l(v_2)$. There is no arc from $x(l(v_1)) \in \mathfrak{Lo}_q$ to $y(l(v_2)) \in \mathfrak{Hi}_q$ in $T'_q$, so there is no edge between $(v_1, x) \in U_H$ and $(v_2, y) \in V_H$. The size of $\mathcal{BIS}$ is $|\mathcal{BIS}| \geqslant (1-\epsilon)|V_G|\left(\frac{\lfloor q/2 \rfloor}{q}\right)q^R \cdot 2 \geqslant \left(\frac{1}{2} - \epsilon - \frac{1}{q}\right)|U_H \uplus V_H|$.

Now we define a perfect matching $\mathcal{M}$ between the vertices not in $\mathcal{BIS}$. First, for every $v \in V'$ we will match the vertex $(v,x) \in U_H$ where $x(l(v)) \notin \mathfrak{Lo}_q$ with $(v,x') \in V_H$

where

$$x'(i) = \begin{cases} x(i) & \text{when } i \neq l(v), \\ q - x(i) & \text{when } i = l(v). \end{cases}$$

For any vertex $v \in V_G \setminus V'$ let $\mathcal{M}((v,x)) = (v,x)$. This edge exists since there was a loop $(i,i)$ in $T'_q$ for every $i \in \mathbb{Z}_q$. □

With Lemmas 19 and 17 we can conclude.

**Corollary.** *Assuming Strong Unique Games Conjecture, for any $\epsilon > 0$, it is* **NP***-hard given a bipartite graph $G = (U \cup V, E)$ (with $|U| = |V|$) to distinguish between two cases.*

- *(YES) There is a maximal matching in $G$ of size $(\frac{1}{2} - \epsilon)|U|$.*
- *(NO) There is no bi-independent set of size $\epsilon|U \cup V|$, so every maximal matching must have at least $(1-\epsilon)|U|$ edges.*

which gives us Theorem 3, as the $(2-\epsilon)$-approximation algorithm faced with a (YES) instance would produce a solution of size at most $(1-2\epsilon)|U|$ unequivocally qualifying it as a (YES) case.

# 6. Bipartite graphs III: Hardness based on SSEH

Our third reduction starts from the Small Set Expansion problem. A modification of the hardness of approximation proof of Maximum Bi-Independent Set from [Man17], it is more complicated than the one presented in Section 5. The reduction is two-step, with an intermediate problem called Max UnCut Hypergraph Bisection (MUCHB) defined below.

**Problem 20** (Max UnCut Hypergraph Bisection)
Given a hypergraph $H = (V_H, E_H)$ decide between cases:

- (YES) There is a bisection[6] $(V_H^0, V_H^1)$ of $V_H$ s.t. $|E_H(V_H^0)|, |E_H(V_H^1)| \geq (\frac{1}{2} - \epsilon)|E_H|$.
- (NO) For every set $T \subset V_H$ of size $|T| \leq \frac{|V_H|}{2}$, $|E_H(T)| \leq \epsilon|E_H|$.

We are only modifying the second step, but we need to describe the first one as well, as we are relying on the structure of the solution in the (YES) case. Before we are able to do that, let us give an overview of the reduction, to explain the origin of its various components.

[6] $(A,B)$ is a bisection of $X$ if $|A| = |B| = \frac{|X|}{2}$ and $A \cup B = X$.

## 6.1. Reducing from SSEH to UGC

In order to understand the reduction it makes sense to first see an outline of the proof that SSEH is strictly stronger than UGC, which was presented in [RS10]. This will help us understand the reduction of Manurangsi in the next subsection.

We begin with a graph $G = (V, E)$ — an instance of the Small Set Expansion problem. We will produce a bipartite graph $\mathcal{H}_1 = (\mathcal{U}_1 \cup \mathcal{V}_1, \mathcal{E}_1, \{\pi_e\}_{e \in \mathcal{E}_1})$ which shall constitute an instance of Unique Label Cover problem.

**Reduction 21** (SSEH to UGC: First attempt)
Set $R = \frac{1}{\delta}$ ($\delta$ is a parameter from SSE). Let $\mathcal{U}_1 = \mathcal{V}_1 = V^R$, so every vertex in $\mathcal{H}_1$ corresponds to a vector of $R$ vertices from $G$.

For $\mathbf{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_R \end{bmatrix}$ and $\mathbf{v} = \begin{bmatrix} v_1 \\ \vdots \\ v_R \end{bmatrix}$ we add $(\mathbf{u}, \mathbf{v})$ to $\mathcal{E}_1$ if there is a permutation $\pi$ such that $\pi(\mathbf{u})$ and $\mathbf{v}$ are neighbours in $G^{\otimes R}$. We set $\pi_{(\mathbf{u},\mathbf{v})} = \pi$.

The idea behind this reduction is that if there is a well-separated set $S \subset V$ of size $\delta|V|$ in $G$, then a vector of $\frac{1}{\delta}$ vertices should in expectation have exactly one vertex in $S$. We call a vector $\mathbf{v} \in V^R$ $S$-*singular* if $|\{v_1, \ldots, v_R\} \cap S| = 1$. It is clear that if $\mathbf{u} \in \mathcal{U}_1$ and $\mathbf{v} \in \mathcal{V}_1$ are both $S$-singular and $(\mathbf{u}, \mathbf{v}) \in \mathcal{E}_1$ then the constraint $\pi_{(\mathbf{u},\mathbf{v})}$ is satisfied (see Fig. 3). Unfortunately, only $\frac{1}{e}$-fraction of all the vectors will be $S$-singular, so Reduction 21 is clearly a failed attempt, as the easy part—completeness—is not working.

There is a simple way to overcome this problem. If we pick a large constant $k$ and sample $k$ distinct $R$-element vectors (blocks) of vertices, it is very likely that at least one of these vectors will be $S$-singular. This idea stands behind the second attempt at reducing SSEH to UGC.

**Reduction 22** (SSEH to UGC: Second attempt)
Let $R = \frac{1}{\delta}$ and $k$ be a constant depending on $\epsilon$ we wish to achieve in our UGC. Let $\mathcal{H}_2 = (\mathcal{U}_2 \cup \mathcal{V}_2, \mathcal{E}_2, \{\pi_e\}_{e \in \mathcal{E}_2})$ where $\mathcal{U}_2 = \mathcal{V}_2 = V^{R \times k}$ (every vertex in $\mathcal{H}_2$ corresponds to $k$ length-$R$ vectors of vertices of $G$).

For $\mathbf{u} = \begin{bmatrix} u_1^1 & \cdots & u_1^k \\ & \cdots & \\ u_R^1 & \cdots & u_R^k \end{bmatrix}$ and $\mathbf{v} = \begin{bmatrix} v_1^1 & \cdots & v_1^k \\ & \cdots & \\ v_R^1 & \cdots & v_R^k \end{bmatrix}$ we add $(\mathbf{u}, \mathbf{v})$ to $\mathcal{E}_2$ if there are $k$ permutations $\pi^1, \ldots, \pi^k$ such that $\pi^1(\mathbf{u}^1)$ and $\mathbf{v}^1$ are neighbours in $G^{\otimes R}$, and $\pi^2(\mathbf{u}^2)$ and $\mathbf{v}^2$ are neighbours in $G^{\otimes R}$ and so on[7]. We will set $\pi_{(\mathbf{u},\mathbf{v})} = \pi^{1,\ldots,k} = \pi^1|\pi^2|\ldots|\pi^k$ (concatenation of $k$ permutations).

[7] Equivalently we may write that $\pi^{1,\ldots,k}(\mathbf{u})$ and $\mathbf{v}$ are neighbours in $G^{\otimes(R \times k)}$.

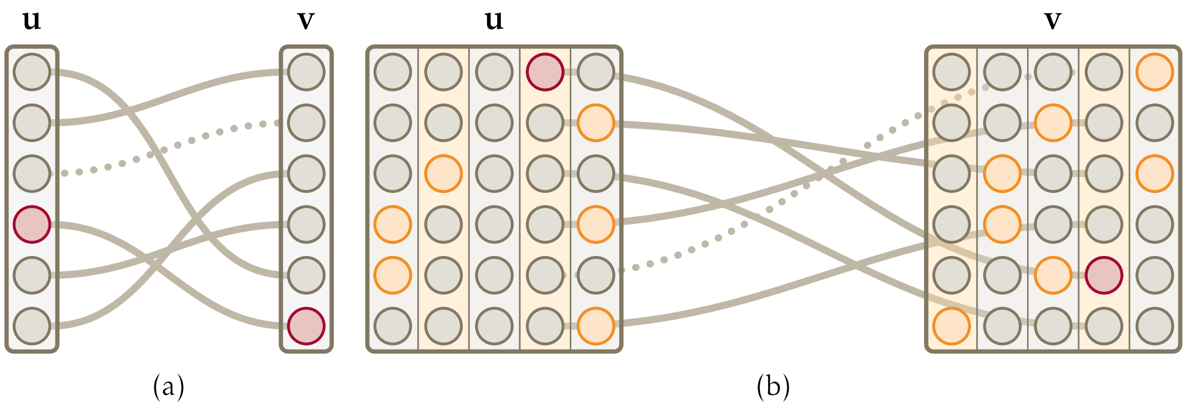



Figure 3: In Reduction 21 (a), a vertex of $\mathcal{H}_1$ corresponds to a block (vector) of $R$ vertices of $G$. Two blocks are connected if one can be permuted so that the blocks are point-wise connected in $G$. Here both are $S$-singular and neither of the connecting edges is cut by $S$, so the colouring $l_S$ with $l_S(\mathbf{u}) = 4$ and $l_S(\mathbf{v}) = 6$ satisfies the constraint. In a refined Reduction 23, $\epsilon R$ pairs connected by the permutation are not connected in $G$ (dotted line). In Reduction 22 (b), a vertex of $\mathcal{H}_2$ corresponds to $k$ blocks of $R$ vertices of $G$ each. In $\mathbf{u}$ second and fourth block are $S$-singular, in $\mathbf{v}$ first and fourth. Colouring $l_S$ with $l_S(\mathbf{u}) = (4,1)$, $l_S(\mathbf{v}) = (4,5)$ satisfies the constraint.

The reader may verify that the completeness of this reduction is satisfied. However, it turns out that the reduction presented above does not yet provide the desired soundness, and certain 'noises' have to be added.

For simplicity we describe them as a modification of the Reduction 21. Combining such a modified version of the reduction into *blocks* gives a correct instance of ULC.

**Reduction 23** (SSEH to UGC: First attempt + random noises, [Ste10, Reduction 6.5])
Let $R = \frac{1}{\delta}$. Let $\mathcal{U}_1 = \mathcal{V}_1 = V^R$, so every vertex in $\mathcal{H}_1$ corresponds to a vector of $R$ vertices from $G$.

For $\mathbf{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_R \end{bmatrix}$ and $\mathbf{v} = \begin{bmatrix} v_1 \\ \vdots \\ v_R \end{bmatrix}$ we add $(\mathbf{u},\mathbf{v})$ to $\mathcal{E}_1$ if there is set $T \subset [R]$ of size $\epsilon R$, and a permutation $\pi$ such that $(u_i, v_{\pi(i)}) \in E(G)$ whenever $i \notin T$. We set $\pi_{(\mathbf{u},\mathbf{v})} = \pi$.

Every edge in $\mathcal{E}_1$ would now correspond to an edge in $G^{\otimes R}$, if not for the sets $T$ in the reduction. Alternatively, sets $T$ could be generated by picking every coordinate with probability $\epsilon$. That would result in an edge-weighted instance of ULC—the weight of an edge equal to the probability of sampling the set $T$ associated with it.

## 6.2. Max Uncut Hypergraph Bisection

The first phase of the reduction was described in [Man17]. It begins with a graph $G = (V, E)$ and produces a weighted hypergraph $H = (V_H, E_H, w_H)$, an instance of MUCHB. We will present two equivalent formulations of this reduction—one will be combinatorial (easier to work with), the other (original) is probabilistic and we present it for the sake of completeness (as it defines the weight function and may be more intuitive).

**Reduction 24** (combinatorial)
Let $R = \Theta(\frac{1}{\delta})$, $k$, $l$ be constants. Let also $\Omega = \{0, 1, \perp\}$. The hypergraph is built as follows:

- The set of vertices $V_H = V^{R\times k} \times \Omega^{R\times k}$.
- Every $e \in E_H$ is associated with a tuple $\left\langle B^1, \dots, B^l, x, D \right\rangle$ where $B^1, \dots, B^l \in V^{R\times k}$, $x \in \Omega^{R\times k}$, $D \subseteq [R] \times [k]$.
- A hypervertex $v = (A_v, x_v)$ belongs to the hyperedge $e = \left\langle B_e^1, \dots, B_e^l, x_e, D_e \right\rangle$ if there is a permutation $\pi \in (\mathfrak{S}_R)^k$ such that:
  (i) $\exists p$ s.t. $A_v(i) = B_e^p(\pi(i))$ for all $i$ where $x_v(i) \neq \perp$.
  (ii) $x_v(i) = x_e(\pi(i))$ for all $i$ such that $\pi(i) \notin D_e$.

We will denote as $\pi_{v,e}$ the permutation that is used to show that $v \in e$. [8] The weight $w_H(e)$ of an edge is defined by a probabilistic process described below.

To understand how this construction is obtained, imagine first, how we might try to reduce UGC to MUCHB. The same dictatorship test as in Section 3 will be used, only in a slightly different way. Every hyperedge now corresponds to an accepting run of the dictatorship test. It contains all consistent inputs to the test.

**Reduction 25** (An incorrect reduction from ULC to MUCHB)
Input: A ULC$(\epsilon, t, R)$ instance $G = (U \cup V, E, \{\pi_e\}_{e\in E})$.
Output: A hypergraph $\mathcal{H} = (V_{\mathcal{H}}, E_{\mathcal{H}})$ with $V_{\mathcal{H}} = U \times \{0,1\}^R$. For $v \in V$, its $l$ neighbours $u^1, \dots, u^l$, a vector $x \in \{0,1\}^R$ and $S \subset [R]$ of size $\epsilon R$ we compose a hyperedge $e_{u^1,\dots,u^l,v,x,S} = \left\{(u^p, x') \mid p \in [l], x \circ \pi_{(u,v)} \in C_{x,S}\right\}$.

[8] This may be not unique. The choice of a permutation is arbitrary.

Clearly, this reduction has the desired completeness. The soundness is not satisfied, as our ULC instance could potentially be composed of two unconnected copies of a (NO) instance. It would still be a (NO) instance, but a resulting hypergraph would have a perfect bisection.

Instead our hypergraph $H$ from Reductions 24 and 26 is constructed by running a similar operation on an instance created in the previous subsection. To describe the probabilistic construction we need to introduce some notation.

- Let $\varepsilon_V$ be a small constant. For each $A \in V^{R\times k}$, $T_V(A)$ denotes a distribution on $V^{R\times k}$ where $(i,j)$-th coordinate is set to $A_{i,j}$ with probability $1-\varepsilon_V$ and a uniformly random vertex of $V$ otherwise.
- For $A \in V^{R\times k}$, $G^{\otimes(R\times k)}(A)$ denotes a distribution on $V^{R\times k}$ where $(i,j)$-th coordinate is a random neighbour of $A_{i,j}$ in $G$.
- Let $\beta$ be a small constant. $\Omega_\beta = \{0,1,\perp\}_\beta$ denotes a distribution over $\Omega = \{0,1,\perp\}$ where 0 and 1 are sampled with probability $\beta/2$ and $\perp$ is sampled with probability $1-\beta$.
- Let $\varepsilon_T$ be a small constant. $S_{\varepsilon_T}(R)$ is a distribution over subsets of $[R]$ of size $\varepsilon_T R$.
- For $x \in \Omega^{R\times k}$ and $A \in V^{R\times k}$ let $M_x(A) = \left\{A' \in V^{R\times k} \mid \forall i \in [R\times k].x_i = \perp \vee A_i = A'_i\right\}$.
- For a set $D \subseteq [R]\times[k]$ and $x \in \Omega_\beta^{R\times k}$ let $C_D(x) = \left\{x' \mid x'_{([R]\times[k])\setminus D} = x_{([R]\times[k])\setminus D}\right\}$.

**Reduction 26** (probabilistic)

1. Sample $A \sim V^{R\times k}$ and $\tilde{A}^1,\ldots,\tilde{A}^l \sim T_V(A)$.
2. Sample $B^1 \sim G^{\otimes(R\times k)}(\tilde{A}^1),\ldots,B^l \sim G^{\otimes(R\times k)}(\tilde{A}^l)$ and $\tilde{B}^1 \sim T_V(B^1),\ldots,\tilde{B}^l \sim T_V(B^l)$.
3. Sample $x \sim \Omega_\beta^{R\times k}$ and $D \sim S_{\varepsilon_T}(Rk)$.
4. Output a hyperedge $\Big\{(\pi(B'),\pi(x')) \mid \exists p \in [l], \pi = \pi^1|\ldots|\pi^k \in (\mathfrak{S}_R)^k, x' \in C_D(x),$ $B' \in M_{x'}(\tilde{B}^p)\Big\}$.

*Remark.* Reductions 24 and 26 produce a weighted hypergraph (with weights assigned to hyperedges). By copying the hyperedges with quantities proportional to their weights we can make the instance unweighted as defined in Problem 20.

Manurangsi [Man17] shows that for every $\varepsilon > 0$ there exist constants $R$, $k$, $l$, $\beta$, $\varepsilon_V$, $\varepsilon_T$ such that the following theorems hold:

**Theorem 27** ([Man17, Lemma 9])
*If $G$ came from a (YES) case of SSEH, then there is a bisection $(V_H^0, V_H^1)$ of $V_H$ such that $|E_H(V_H^0)|, |E_H(V_H^1)| \geqslant (1/2 - \epsilon)|E_H|$.*

**Theorem 28** ([Man17, Lemma 14])
*If $G$ came from a (NO) case of SSEH, then every subset $T \subset V_H$ with $|T| \leqslant \frac{|V_H|}{2}$ has $|E_H(T)| \leqslant \epsilon |E_H|$.*

Henceforth, we will assume that we use the same parameters $R$, $k$, $l$, $\beta$, $\varepsilon_V$, $\varepsilon_T$ as in [Man17].

### Completeness—The bisection

We present the construction of sets $V_H^0$ and $V_H^1$ from Theorem 27 (in the *(YES)* case). They will be important in our further proceedings. The following additional notation will be used:

- For a vector $A \in V^{R\times k}$ let $A^1, \dots, A^k \in V^R$ be its *blocks*. Let $W(A,x,j)$ denote the set of all coordinates $i$ in $j$-th block such that $x_i \neq \perp$ and $A_i^j \in S$, i.e., $W(A,x,j) = \left\{ i \in [R] \mid A_i^j \in S \wedge x_i \neq \perp \right\}$.
- Let $j^*(A,x) \in [k]$ denote the first *$x$-filtered $S$-singular* block, i.e., smallest $j$ with $|W(A,x,j)| = 1$. Note that if such block does not exist, we set $j^*(A,x) = -1$.
- Let $i^*(A,x) \in [R] \times [k]$ be the only element in $W(A,x,j^*(A,x))$. If $j^*(A,x) = -1$, let $i^*(A,x) = -1$. For a hypervertex $v = (A_v, x_v)$ we will sometimes write $i^*(v)$ instead of $i^*(A_v, x_v)$.

We will split all hypervertices into three sets: $T_0, T_1, T_\perp$[9]. Let $T_\perp$ be the set of hypervertices $(A,x)$, such that $j^*(A,x) = -1$. Then, for every hypervertex $(A,x)$ not in $T_\perp$ we look at the value $b = x_{i^*(A,x)}$ and include it in $T_0$ if $b = 0$ and in $T_1$ if $b = 1$. It is shown that $|E_H(T_0)|, |E_H(T_1)| \geqslant (\frac{1}{2} - \epsilon)|E_H|$.

### Not-disjointness relation between hypergraph edges

Following from the definition of the hypergraph (as in Reduction 24), we have the following property.

**Property 29.** *Two hyperedges $e_1 = \left\langle B_1^1, \dots, B_1^l, x_1, D_1 \right\rangle$ and $e_2 = \left\langle B_2^1, \dots, B_2^l, x_2, D_2 \right\rangle$ of the hypergraph are <u>not disjoint</u> if there exist two permutations $\pi_1, \pi_2 \in (\mathfrak{S}_R)^k$ satisfying the following conditions*

[9] $T_0$ and $T_1$ are called $T_0'$ and $T_1'$ respectively in [Man17].

*(1)* $x_1(\pi_1(i)) = x_2(\pi_2(i))$ *for* $i$ *such that* $\pi_1(i) \notin D_1 \wedge \pi_2(i) \notin D_2$;

*(2)* $\exists p_1, p_2 \in [l]$ *such that* $B_1^{p_1}(\pi_1(i)) = B_2^{p_2}(\pi_2(i))$ *for all* $i$ *satisfying*

   *(i)* $x_j(\pi_j(i)) = \perp \Rightarrow \pi_j(i) \in D_j$ *for* $j \in \{1,2\}$, *and*

   *(ii)* $\pi_1(i) \notin D_1 \vee \pi_2(i) \notin D_2$.

We will denote the *not-disjointness* relation by $\Lambda$. So $(e_1, e_2) \in \Lambda$ iff exists a hypergraph vertex $v \in e_1 \cap e_2$. Similarly, we define our relation $\Delta \subset E_H \times E_H$. Its definition is similar to the conditions of not-disjointness, but it is not symmetrical. It is also a superset of $\Lambda$.

**Definition 30.** Two hyperedges $e_1 = \left\langle B_1^1, \dots, B_1^l, x_1, D_1 \right\rangle$ and $e_2 = \left\langle B_2^1, \dots, B_2^l, x_2, D_2 \right\rangle$ of the hypergraph are in relation $\Delta$ if there exist two permutations $\pi_1, \pi_2 \in (\mathfrak{S}_R)^k$ satisfying the following conditions

(1) $x_1(\pi_1(i)) \preccurlyeq x_2(\pi_2(i))$ for $i$ such that $\pi_1(i) \notin D_1 \wedge \pi_2(i) \notin D_2$;

(2) $\exists p_1, p_2 \in [l]$ such that $B_1^{p_1}(\pi_1(i)) = B_2^{p_2}(\pi_2(i))$ for all $i$ satisfying

   (i) $x_j(\pi_j(i)) = \perp \Rightarrow \pi_j(i) \in D_j$ for $j \in \{1,2\}$, and

   (ii) $\pi_1(i) \notin D_1 \vee \pi_2(i) \notin D_2$.

$x_1 \preccurlyeq x_2$ is a point-wise inequality on vectors of $\Omega$, where $a \preccurlyeq a$ for all $a \in \Omega$ and $0 \preccurlyeq 1$.

## 6.3. From MUCHB to Bipartite MMM

In [Man17] the bipartite graph $\mathcal{G}^{\mathcal{BIS}} = (\mathscr{V}_L \cup \mathscr{V}_R, \mathscr{E}^{\mathcal{BIS}})$ is created the following way.

- $\mathscr{V}_L, \mathscr{V}_R = E_H$.
- $e_l \in \mathscr{V}_L$ and $e_r \in \mathscr{V}_R$ are connected if they are not disjoint (i.e. $(e_l, e_r) \in \Lambda$).

We are applying a modification to this construction. Instead of $\Lambda$ we are basing the graph on the relation $\Delta$ which will result in a superset of edges of $\mathscr{E}^{\mathcal{BIS}}$.

**Reduction 31** (MUCHB to MMM)
The bipartite graph $\mathcal{G}^{\mathcal{MMM}} = (\mathscr{V}_L \cup \mathscr{V}_R, \mathscr{E}^{\mathcal{MMM}})$ is built as follows.

- $\mathscr{V}_L, \mathscr{V}_R = E_H$.
- $e_l \in \mathscr{V}_L$ and $e_r \in \mathscr{V}_R$ are connected in $\mathscr{E}^{\mathcal{MMM}}$ if $(e_l, e_r) \in \Delta$.

### Completeness

In the (YES) case we need to find a maximal matching that leaves roughly half of the vertices of $\mathcal{G}^{\mathcal{MMM}}$ unmatched. On the next few pages we will be constructing the matching, as promised in the following lemma.

**Lemma 32** (Completeness). *If $G$ came from the (YES) case of SSEH, then in $\mathcal{G}^{\mathcal{MMM}}$ there is a maximal matching of size at most $(\frac{1}{2}+\epsilon)|\mathscr{V}_L|$.*

Our matching $M$ will be composed of two parts. In $M_1$, $\mathscr{V}_L \cap E_H(T_0)$ will be matched with $\mathscr{V}_R \cap E_H(T_1)$ while $\mathscr{V}_L \cap E_H(T_1)$ and $\mathscr{V}_R \cap E_H(T_0)$ will form a bi-independent set (this is why we needed to build the graph on an asymmetric relation). This will be the larger part, of size $(\frac{1}{2}-\epsilon)|\mathscr{V}_L|$. In $M_2$, the vertices in $\mathscr{V}_L \setminus (E_H(T_0) \cup E_H(T_1))$ and $\mathscr{V}_R \setminus (E_H(T_0) \cup E_H(T_1))$ will be paired.

**Constructing $M_1$.** The first part of our matching will be based on a bijection between $E_H(T_0)$ and $E_H(T_1)$. For a hyperedge $t = \left\langle B^1, \ldots, B^l, x, D\right\rangle$ we define its *witness indexes* $W(t) = \left\{i \in [R] \times [k] \mid \exists p \in [l], x' \in C_D(x), i = i^*(B^p, x')\right\}$. Now we construct a bijection $\phi\colon E_H(T_0) \to E_H(T_1)$. Let $t_0 = \left\langle B^1, \ldots, B^l, x_0, D\right\rangle$ be any hyperedge in $E_H(T_0)$. Then let a vector $x_1$ be as follows.

$$x_1(i) = \begin{cases} 1 & \text{if } i \in W(t_0), \\ x_0(i) & \text{otherwise.} \end{cases} \tag{1}$$

We set $\phi(t_0) \coloneqq t_1$, where $t_1 = \left\langle B^1, \ldots, B^l, x_1, D\right\rangle$. We verify the validity of this definition in the next lemmas. In Lemma 36 we prove that $\phi$ is a properly defined function, meaning that $t_1 \in E_H(T_1)$. Then in Lemma 37 we prove that $\phi$ is a bijection. To prove these Lemmas, we first need the following facts, explaining why $W(t)$ are the *witness indexes*.

**Lemma 33.** *Let $W'(t)$ be constructed in the following way. Consider any hypervertex $v$ in $t$. For every such $v$ we include $\pi_{v,t}(i^*(v))$ in $W'(t)$. Then $W(t) = W'(t)$.*

*Proof.* We will first show that $W(t) \subseteq W'(t)$. Consider any $i \in W(t)$. Let $p \in [l]$, $x' \in C_D(x)$ be such that $i = i^*(B^p, x')$. Then $(B^p, x') \in t$, so $i \in W'(t)$.

Next, we will show that $W'(t) \subseteq W(t)$. Let $v = (A_v, x_v)$ be any vertex in $t$. Let $p$ be such that $A_v(i) = B^p_e(\pi_{v,t}(i))$ for all $i$ where $x_v(i) \neq \perp$. We consider another vertex $v' = (\pi_{v,t}(A_v), \pi_{v,t}(x_v))$, which also is in $t$. $\pi$ permutes each block in $A_v$ separately, so it preserves value $i^*$, that is $i^*(v') = \pi_{v,t}(i^*(v))$. Now we claim that $i^*(v') = i^*(B^p, x)$. It is so, because if $A_v(\pi_{v,t}(j)) \neq B^p(j)$ then $x(\pi_{v,t}(j)) = \perp$. ∎

**Lemma 34.** *Let $t_0$ be any hyperedge in $E_H(T_0)$. Then $D_0 \cap W(t_0) = \varnothing$.*

*Proof.* Assume $\exists i \in D_0 \cap W(t_0)$. Let $v = (A, x) \in t_0$ be such that $\pi_{v,t_0}(i^*(v)) = i$. Define $x'(i) = 1$ and $x'(j) = x(j)$ for all $j \neq i$. The vertex $v' = (A, x')$ is still in $t_0$, because $i \in D_0$. $v'$ is however also in $T_1$ because $x'(i^*(v')) = 1$, which contradicts $t_0 \in E_H(T_0)$. □

**Lemma 35.** *For any $t = \left\langle B^1, \ldots, B^l, x, D \right\rangle \in E_H(T_0)$, for any $i \in W(t)$, $x(i) = 0$.*

*Proof.* For any $p \in [l]$ and $x' \in C_D(x)$, a vertex $v = (B^p, x')$ is in $t$, and, by consequence, in $T_0$. Thus, $x'(i^*(B^p, x')) = 0$. As $i^*(v) \notin D$, $x(i^*(B^p, x')) = 0$ as well. □

**Lemma 36.** *Let $t_0$ be any hyperedge in $E_H(T_0)$ and $t_1 = \phi(t_0)$. Let $v = (A, x)$ be any hypervertex in $t_1$. Then $v \in T_1$.*

*Proof.* $\pi_{v,t_1}(i^*(A, x)) \in W(t_1)$, so by (1) we have $x_1(\pi_{v,t_1}(i^*(A, x))) = 1$. $\pi_{v,t_1}(i^*(A, x)) \notin D_0$, so $x(i^*(A, x)) = x_1(\pi_{v,t_1}(i^*(A, x))) = 1$. Thus, $v \in T_1$. □

**Lemma 37.** *Function $\phi$ is a bijection.*

*Proof.* Observe that for any hyperedge $t_0 \in E_H(T_0)$ we have $W(t_0) = W(\phi(t_0))$. This follows from the fact that $i^*(\,\cdot\,, x)$ does not distinguish between zeros and ones in $x$. Now, it is easy to construct the inverse function to $\phi$. $\phi^{-1}$ will be almost the same as $\phi$, except that in the vector $x$ we will set $x(i)$ to be 0 instead of 1. Thanks to the observation, $\phi^{-1}$ changes the same coordinates in $x$ as $\phi$, so it is an inverse of $\phi$. □

The matching between $\mathscr{V}_L \cap E_H(T_0)$ and $\mathscr{V}_R \cap E_H(T_1)$ uses $\phi$ in a natural way: $M_1 = \left\{(t_0, \phi(t_0)) \mid t_0 \in \mathscr{V}_L \cap E_H(T_0)\right\}$. We need to show that all the edges of this matching are in graph $\mathcal{G}^{\mathcal{MMM}}$.

**Lemma 38.** *Let $t_0, t_1$ be any pair such that $\phi(t_0) = t_1$. Then $(t_0, t_1) \in \Delta$.*

*Proof.* We choose both $\pi_1, \pi_2$ to be the identity permutation. Both conditions of Definition 30 are satisfied:

(1) is satisfied, as for all $i$ either $x_0(i) = x_1(i)$ or $x_0(i) = 0$ and $x_1(i) = 1$, which is exactly the definition of $x_0(i) \preccurlyeq x_1(i)$.

(2) is satisfied, because $B_0^1 = B_1^1$.

□

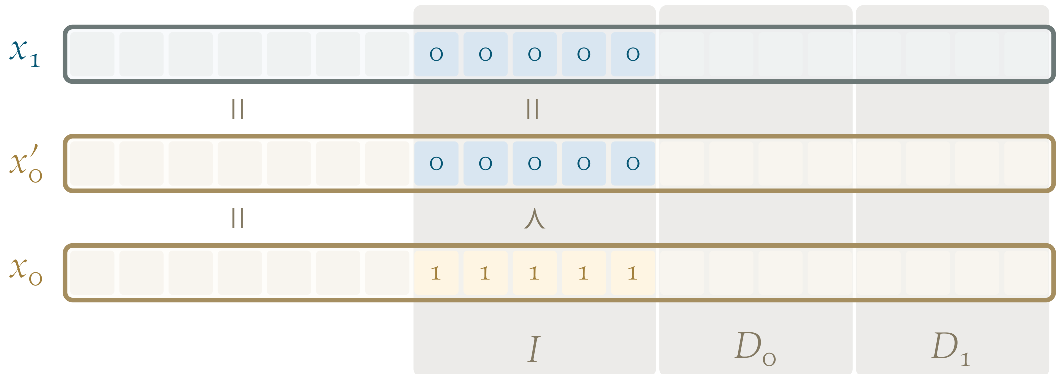



Figure 4: Vectors from the proof of Lemma 39. $x_1$ and $x_0$ are identical everywhere apart from positions in $D_0 \cup D_1 \cup I$. $x'_0$ is constructed from $x_0$ by replacing ones with noughts at indices of $I$. The vector $x_{v'}$ of $v' \in t_1 \cap t'_0$ must agree with $x_1$ everywhere except $D_1$, and with $x'_0$ everywhere except $D_0$.

**Constructing** $M_2$. As we said earlier, the vertices in $\mathscr{V}_L \setminus (E_H(T_0) \cup E_H(T_1))$ and $\mathscr{V}_R \setminus (E_H(T_0) \cup E_H(T_1))$ need to be taken care of. We are going to match them according to identity function: $M_2 = \left\{(e_l, e_r) \mid e_l \in \mathscr{V}_L \setminus (E_H(T_0) \cup E_H(T_1)), e_R \in \mathscr{V}_R \setminus (E_H(T_0) \cup E_H(T_1)) \text{ st. } e_l = e_r\right\}$ (every vertex is matched with its own copy). These pairs are in $\mathscr{E}^{\mathcal{MMM}}$ as the relation $\Delta$ is reflexive[10]. We define our matching $M = M_1 \cup M_2$.

**Maximality of** $M$. Now, all vertices left unmatched by $M$ are either in $\mathscr{V}_L \cap E_H(T_1)$ or $\mathscr{V}_R \cap E_H(T_0)$. In the following lemma we prove that they form a bi-independent set, and therefore our matching is maximal.

**Lemma 39.** *For every $t_1 \in E_H(T_1)$ and $t_0 \in E_H(T_0)$, $(t_1, t_0) \notin \Delta$.*

*Proof.* Assume that there is $t_1 = \left\langle B_1^1, \ldots, B_1^l, x_1, D_1 \right\rangle \in E_H(T_1)$ and $t_0 = \left\langle B_0^1, \ldots, B_0^l, x_0, D_0 \right\rangle$ such that $(t_1, t_0) \in \Delta$. We will show that there exists a hypervertex $v \in t_0 \cap T_1$, and consequently $t_0 \notin E_H(T_0)$.

The relations $\Delta$ and $\Lambda$ are very similar. If $(t_1, t_0) \in \Lambda$ ($t_1$ and $t_0$ are not disjoint), we have that $t_0 \cap T_1 \neq \varnothing$ right away. Otherwise, from the definition of $\Delta$ we have that there are two permutations $\pi_1$, $\pi_0$ such that $\forall i \notin D_1 \cup D_0 . x_1(\pi_1(i)) \preccurlyeq x_0(\pi_0(i))$ and the inequality is strict for at least one $i$ (since $(t_1, t_0) \notin \Lambda$), we shall denote the set of these

[10] In fact, we did not need to define $M_2$ to prove the completeness: Since $M_1$ matches almost half of vertices from each side, and (as we are proving in Lemma 39) another almost half forms a bi-independent set, extending $M_1$ greedily could extend it by at most $\epsilon|E_H|$ edges. That would give us the Lemma 32 with the matching size $(1 + 2\epsilon)|\mathscr{V}_L|$.

indices with strict inequality as $I$ (see Fig. 4). By permuting the indices in $t_1$ and $t_0$ we will assume that $\pi_1$ and $\pi_0$ are both the identity permutations. Let us consider another hyperedge $t_0' = \left\langle B_0^1, \ldots, B_0^l, x_0', D_0 \right\rangle$, where $x_0'$ is defined as follows:

$$x_0'(i) = \begin{cases} 0 & \text{if } i \in I, \\ x_0(i) & \text{otherwise.} \end{cases} \tag{2}$$

Clearly $(t_1, t_0') \in \Lambda$ (by Property 29), which means that $t_0' \notin E_H(T_0)$. Let $v' = (A_{v'}, x_{v'})$ be such that $v' \in t_0' \cap T_1$ (again, we will assume that $\pi_{v',t_0'}$ is the identity permutation). We will change $v'$ slightly, to 'reverse' changes made in (2). $x_v$ will be defined as follows:

$$x_v(i) = \begin{cases} 1 & \text{if } i \in I, \\ x_{v'}(i) & \text{otherwise.} \end{cases} \tag{3}$$

Now, a hypervertex $v = (A_{v'}, x_v)$ belongs to $t_0$ (with the witness permutation $\pi_{v,t_0} = \pi_0 \circ \pi_{v',t_0'}$). To finish the proof we need to show that $v \notin T_0$. Since $v' \in T_1$, we have that $x_{v'}(i^*(A_v, x_v)) = 1$. Then, from (3), also $x_v'(i^*(A_v, x_v')) = x_v(i^*(A_v, x_v))$ as $I \cap W(t_1) = \varnothing$, so $v$ is also in $T_1$. □

To finish the proof of Lemma 32 we bound the size of matching $M$.

$$|M| = |M_1| + |M_2| = |E_H(T_0)| + |\mathscr{V}_L \setminus (E_H(T_0) \cup E_H(T_1))| = |\mathscr{V}_L| - |E_H(T_1)| \leqslant \left(\frac{1}{2} + \epsilon\right)|\mathscr{V}_L|.$$

### Soundness

We recall Lemma from [Man17] about graph $\mathcal{G}^{\mathcal{BIS}}$ in the (NO) case of MUCHB.

**Lemma 40** ([Man17, Appendix A]). *If for every $T \subset V_H$ of size at most $\frac{|V_H|}{2}$ we have $|E_H(T)| \leqslant \epsilon |E_H|$, then in $\mathcal{G}^{\mathcal{BIS}}$ there is no bi-independent set of size $2 \cdot (\epsilon |E_H| + 1)$.*

As the set of edges of graph $\mathcal{G}^{\mathcal{MMM}}$ is a superset of the set of edges of $\mathcal{G}^{\mathcal{BIS}}$, analogous statement is also true for $\mathcal{G}^{\mathcal{MMM}}$.

**Lemma 41** (Soundness). *If $H$ came from the (NO) case of Problem 20, then in $\mathcal{G}^{\mathcal{MMM}}$ there is no bi-independent set of size $2 \cdot (\epsilon |E_H| + 1)$.*

Altogether Lemmas 32 and 41 give us the following, from which Theorem 4 immediately follows.

**Corollary.** *Assuming Small Set Expansion Hypothesis, for any $\epsilon > 0$ it is* **NP**-*hard given a bipartite graph $G = (U \cup V, E)$ (with $|U| = |V|$) to distinguish between two cases.*

- *(YES) There is a maximal matching in G of size* $(\frac{1}{2}+\epsilon)|U|$.
- *(NO) There is no bi-independent set of size* $\epsilon|U \cup V|$, *so every maximal matching must have at least* $(1-\epsilon)|U|$ *edges.*

# 7. Notes

The dictatorship test described in Section 3 is based on the 'It ain't over till it's over' theorem conjectured by Friedgut and Galai and proved by Mossel, O'Donnell and Oleszkiewicz [MOO05]. The reduction from Unique Label Cover to Vertex Cover, described in Section 3 comes from a paper by Bansal and Khot [BK09]. Its modification proving Theorem 1, described in Section 3.2 is original, albeit based on the same idea, as one in our paper with Szymon Dudycz and Mateusz Lewandowski [DLM19] published at IPCO 2019—only in that paper we were modifying a different reduction of Khot and Regev [KR03]. The proof of Theorem 2 described in Section 4 comes from our aforementioned paper [DLM19].

The original reduction from ULCE to Balanced Bi-Clique, revisited in Section 5, was presented by Bhangale, Gandhi, Hajiaghayi, Khadenkar and Kortsarz [Bha+16]. Our modification proving Theorem 3 comes from our paper with Szymon Dudycz and Pasin Manurangsi [DMM20].

The reduction from SSE to ULC described in Section 6 was presented by Raghavendra and Steurer [RS10] and expanded on in their paper with Tulsiani [RST12]. An accessible write-up of this reduction and other results on Small Set Expansion is Steurer's PhD thesis [Ste10].

The reduction from SSE to Balanced Bi-Clique was given by Manurangsi [Man17]. We have modified it with Dudycz and Manurangsi [DMM20] to prove Theorem 4.

The Minimum Maximal Matching problem was (on separate occasions) suggested to me and Szymon Dudycz by Katarzyna Paluch with expectation that we improve upon the trivial 2-approximation algorithm.

# 3 Proportional Approval Voting

In this chapter we are going to study a family of *multi-winner approval-based* electoral systems. In these systems the voters are electing a committee of $k$ members (e.g. a parliament) by checking boxes next to the candidates they approve of[1], without limitation on the number of approved candidates. An electoral system is a rule determining the result of elections based on the votes cast.

In late 19th century, a Danish mathematician and astronomer Thorvald N. Thiele proposed [Thi95] two such rules with a goal of making approval-voting results proportional. The first, Sequential proprotional approval voting was practical: The result is given by a well-defined algorithm. It was used in Sweden in early 20th century. The second rule, called Proportional approval voting just says that the winning committee must be one that maximises $\mathrm{scr}_{\mathbf{w}}(W) = \sum_{v \in V} \sum_{i=1}^{|W \cap A_v|} w_i$, out of all possible committees $W \subset C$ of size $k$ ($V$ is the set of voters; $C$ denotes the set of candidates; $A_v \subseteq C$ is a set of candidates supported by the voter $v$). The vector $\mathbf{w}$ was set by Thiele to $(1, 1/2, 1/3, \dots)$, but one can see that this choice is arbitrary and in fact there is a different, valid $\mathbf{w}$-Thiele electoral system for every vector $\mathbf{w}$ as long as it is non-increasing (the law of diminishing returns on the voter's happiness) and non-negative (the voter's happiness should not decrease when an additional candidate supported by him is voted in).

## 1. Algorithmic question

Finding a winning committee in a $\mathbf{w}$-Thiele electoral system becomes a computational task. It turns out only to be solvable in polynomial time when $\mathbf{w} = (1, 1, 1, \dots)$[2].

[1] As opposed to the *rank-based* systems like Single Transferrable Vote (used in Australia), where voters order the candidates from the most to the least preferred.

Every other **w**-Thiele electoral system is **NP**-hard to decide [SFL16].The researchers have thus put forward the question of approximability of these voting systems, with justification that, while it obviously would not have an application to political elections, such an algorithm may still be relevant for selection process less prone to controversy like participatory budgets [SFL16].

**Connection to classical problems** When $\mathbf{w} = (1, 0, 0, \dots)$, deciding the election becomes equivalent to the Maximum Coverage problem, where we are given a collection of sets $C = \left\{T_1, \dots, T_{|C|} \mid v \in T_c \Leftrightarrow c \in A_v\right\}$ over the same universe $V$ and must select $k$ of them to cover the most elements. Famously, the greedy algorithm gives an $(1 - 1/e)$-approximation [NWF78] and this is tight [Fei96] unless $\mathbf{P} = \mathbf{NP}$. Moreover, assuming Gap-ETH, even an algorithm running in time $f(k) \cdot (|C| + |V|)^{o(k)}$ (for any function $f$) cannot achieve a ratio smaller than $(1 - 1/e)$ [Man20]. Recently, Barman et al. [Bar+20] analysed the Multi $\ell$-Coverage—with an intermediate between $(1, 1, 1, \dots)$ and $(1, 0, 0, \dots)$—namely $\mathbf{w}_\ell = (1, \dots, 1, 0, \dots)$ ($\ell$ ones followed by noughts). They gave an algorithm with an approximation ratio of $1 - \frac{\ell^\ell}{e^\ell \ell!} \approx 1 - \frac{1}{\sqrt{2\pi\ell}}$. They also showed, this ratio cannot be beaten assuming UGC.

**Our results** We are presenting an approximation algorithm for a natural class of **w**-Thiele rules.

**Definition 1.** A non-increasing vector $\mathbf{x} = (x_1, x_2, \dots)$ is *geometrically dominant* if for every $i \in \mathbb{N}_+$, $x_i \cdot x_{i+2} \geqslant x_{i+1}^2$. Equivalently, $\mathbf{x}$ is geometrically dominant if either $\mathbf{x} = (1, 0, 0, \dots)$ or $\frac{x_i}{x_{i+1}} \geqslant \frac{x_{i+1}}{x_{i+2}}$ ($\mathbf{x}$ decreases slower than a geometric sequence).

All previously studied **w**-Thiele electoral systems are defined by geometrically dominant sequences **w**.

**Theorem 2**
*Let* **w** *be any geometrically dominant sequence. There exists a polynomial-time* $\alpha_{\mathbf{w}}$-*approximation*[3] *algorithm for* **w**-Thiele *rule where*

$$\alpha_{\mathbf{w}} = \mathop{\mathbb{E}}_{x \sim \mathrm{Poi}(1)} \left[ \sum_{i=1}^{x} w_i \right] = \sum_{x=1}^{\infty} \left( \frac{1}{ex!} \sum_{i=1}^{x} w_i \right).$$

[2] For simplicity, throughout this chapter we are assuming $w_1 = 1$. Every vector can be scaled so without any effect on the election.

[3] $\alpha_{\mathbf{w}}$ is the expected score of a voter, if the number of elected candidates from his ballot is sampled from the Poisson distribution with mean one.

| Problem Name | Sequence $\mathbf{w}$ | Our Ratio |
|---|---|---|
| Approval Chamberlin-Courant (aka. Maximum Coverage) | $(1,0,0,\dots)$ | $1 - 1/e \geqslant 0.6321$ |
| Proportional Approval Voting | $w_i = 1/i$ | $0.7965\dots$ |
| Saint-Laguë method | $w_i = 1/(2i-1)$ | $0.7394\dots$ |
| Penrose apportionment method | $w_i = 1/i^2$ | $0.7084\dots$ |
| $p$-Harmonic | $w_i = 1/i^p$ | $\sum_{x=1}^{\infty} \frac{1}{e \cdot x!} \cdot \left(\sum_{i=1}^{x} \frac{1}{i^p}\right)$ |
| $p$-Geometric | $w_i = p^i$ | $\frac{1}{1-p} \cdot \left(1 - \frac{1}{e^{1-p}}\right)$ |

Table 1: Our approximation ratios for $\mathbf{w}$-Thiele rules. They are tight unless **P=NP**. All listed $\mathbf{w}$-Thiele rules were known to have $(1 - 1/e)$-approximation algorithm [NWF78; SFL16]. To the best of our knowledge, none better than $(1 - 1/e)$-approximation algorithm was known for any of these rules.

The algorithm is a simple application of a well known linear programming rounding technique called *pipage rounding* due to Ageev and Sviridenko [AS04]. Our contribution is an analysis of this algorithm for $\mathbf{w}$-Thiele rules. The exact value of this approximation ratio for specific rules is presented in Table 1. Additionally we provide a family of hard instances for the greedy algorithm to show that our algorithm is indeed a stronger approximation, not just better analysed. We are also showing that our approximation ratio is tight.

**Theorem 3**
*Let* $\mathbf{w}$ *be any non-increasing sequence such that* $\lim_{i\to\infty} w_i = 0$. *For any* $\epsilon > 0$ *it is* **NP**-*hard to compute an* $(\alpha_{\mathbf{w}} + \epsilon)$-*approximate solution to* $\mathbf{w}$-Thiele. *Furthermore, assuming Gap-ETH, such an approximation cannot be computed even in time* $f(k) \cdot (|V| + |C|)^{o(k)}$.

Our inapproximability construction is a reduction from Label Cover (Problem 1.1), in principle the same as the reduction for Maximum Coverage of Feige. We do however need a more involved analysis to serve an arbitrary vector $\mathbf{w}$.

## 2. The approximation algorithm

Our algorithm is based on an LP rounding technique, which requires us to first write a linear programming relaxation of the problem. Since $\mathrm{scr}_{\mathbf{w}}$ only makes sense on subsets of candidates, we first extend $\mathrm{scr}_{\mathbf{w}}$ also to a fractional solution $x \in [0,1]^C$

specifying for each candidate fractionally, how much the candidate is selected.

$$\mathrm{scr}_{\mathbf{w}}(x) = \sum_{v \in V} \sum_{l=1}^{|A_v|} w_l \cdot \min\{1, \max\{0, x_{A_v} - l + 1\}\},$$

where $x_{A_v} = \sum_{c \in A_v} x_c$. Intuitively, for each voter $v$ this function adds the first $x_{A_v}$ elements of the sequence $\mathbf{w}$. The last element of the sequence may be counted in only fractionally. With this definition, we now compute the optimum fractional solution $x^*$ of the following linear program.

$$\begin{aligned} \text{maximize} \quad & \mathrm{scr}_{\mathbf{w}}(x) \\ \text{subject to} \quad & \sum_{c \in C} x_c = k \\ & x \in [0,1]^C \end{aligned} \tag{1}$$

*Remark* (Solving the LP). The program (1) is not really an LP, as its objective function is not linear. To solve it we need to rewrite it into an equivalent form:

$$\begin{aligned} \text{maximize} \quad & \sum_{v \in V} \sum_{l=1}^{|A_v|} w_l \cdot y_v^l \\ \text{subject to} \quad & \sum_{c \in C} x_c = k \\ & \sum_{l=1}^{|A_v|} y_v^l \leqslant \sum_{c \in A_v} x_c \quad \forall v \in V \\ & x \in [0,1]^C; y \in [0,1]^{V \times C}. \end{aligned}$$

This equivalence holds because $\mathbf{w}$ is a non-increasing sequence.

To round $x^*$, we will employ a framework called *pipage rounding* [AS04]. Specifically, we resort to the following result of Călinescu et al.

**Theorem 4** ([Căl+11, Lemma 3.5])
*Let $f$ be a monotone, submodular function and a polytope $B(\mathcal{M})$ be described by matroid constraints. Given vector $y^* = \arg\max\left\{f(x) \mid x \in B(\mathcal{M})\right\}$ the procedure* Pipage-Rounding *will return a solution $S \in \mathcal{M}$ of value $\mathbb{E}[f(S)] \geqslant \mathbb{E}_{\hat{y} \sim y^*}[f(\hat{y})]$.*

*Remark.* Applying the theorem directly results in a randomized algorithm. There is however a deterministic version of this theorem [Căl+07]. In order to use it, we have to be able—given a vector $x^*$—to efficiently compute $\mathbb{E}_{\hat{x} \sim x^*}[\mathrm{scr}_{\mathbf{w}}(\hat{x})]$.

This can be done with a simple dynamic programming procedure: Let $\mathsf{dp}_v[k][i]$ denote the probability that exactly $i$ candidates out of $c_1, \dots, c_k \in A_v$ have been selected. Clearly it can be computed using the identity

$$\mathsf{dp}_v[k][i] = \mathsf{dp}_v[k-1][i-1] \cdot x^*_{c_k} + \mathsf{dp}_v[k-1][i] \cdot (1 - x^*_{c_k}).$$

The function $\mathrm{scr}_{\mathbf{w}}$ is submodular (because $\mathbf{w}$ is non-increasing), and monotone (because $\mathbf{w}$ is non-negative). We can thus use Theorem 4 to round the optimum LP solution $x^*$. The resulting approximation ratio is then equal to

$$\rho = \frac{\mathbb{E}_{\hat{x} \sim x^*}[\mathrm{scr}_{\mathbf{w}}(\hat{x})]}{\mathrm{scr}_{\mathbf{w}}(x^*)}.$$

The remainder of this section is dedicated to bounding $\rho$.

Let the approximation ratio of a voter $v$ be $\rho_v := \frac{\mathbb{E}_{\hat{x} \sim x^*}[\mathrm{scr}_{\mathbf{w}}(\hat{x}, v)]}{\mathrm{scr}_{\mathbf{w}}(x^*, v)}$ where $\mathrm{scr}_{\mathbf{w}}(x, v)$ is the score of $v$, i.e., $\sum_{l=1}^{|A_v|} w_k \cdot \min\{1, \max\{0, x_{A_v} - l + 1\}\}$. Henceforth, we will focus on a single voter $v$ and show that $\rho_v \geqslant \alpha_{\mathbf{w}}$. This straightforwardly implies that $\rho \geqslant \alpha_{\mathbf{w}}$ as desired.

We next characterize $x^*$ that minimizes the ratio $\rho_v$, which in the end will lead us to the claimed bound.

## 2.1. Step 1: Three-valued $x^*$

Let $\tau = x^*_{A_v} = \sum_{c \in A_v} x^*_c$ be a sum of $x^*$ values for candidates approved by $v$. The score of the fractional solution $x^*$ is now $\mathrm{scr}_{\mathbf{w}}(x^*, v) = w_1 + \dots + w_{\lfloor \tau \rfloor} + \{\tau\} \cdot w_{\lfloor \tau \rfloor + 1}$.

The numerator $\mathbb{E}_{\hat{x} \sim x^*}[\mathrm{scr}_{\mathbf{w}}(\hat{x}, v)]$ of $\rho_v$ can be written as

$$\sum_{l=0}^{\infty} \left( (w_1 + \dots + w_l) \mathop{\mathbb{P}}_{\hat{x} \sim x^*} \left[ \sum_{c \in A_v} \hat{x}_c = l \right] \right).$$

We recall the following lemma due to Barman et al.

**Lemma 5** ([Bar+20][4]). *Let $x \in [0,1]^m$ be such that $x_1 + \dots + x_m = \tau$. Then, for any non-negative sequence $(a_1, a_2, \dots)$, there exist $q \in (0,1)$ and $x' \in \{0, q, 1\}^m$ such that $\sum_i x'_i = \tau$ and*

$$\sum_l \left( a_l \mathop{\mathbb{P}}_{\hat{x} \sim x} \left[ \sum_{i=1}^m \hat{x}_i = l \right] \right) \geqslant \sum_l \left( a_l \mathop{\mathbb{P}}_{\hat{x} \sim x'} \left[ \sum_{i=1}^m \hat{x}_i = l \right] \right).$$

Now, notice that the denominator $\mathrm{scr}_{\mathbf{w}}(x^*, v)$ in our ratio $\rho_v$ only depends on the sum $\tau = \sum_{c \in A_v} x^*_c$ and not the way values are distributed among $x^*_c$'s. Hence, from Lemma 5, $\rho_v$ is minimised by $x^*$ that only takes three values—0, $q$ and 1, although we do not yet know what the value of $q$ is[5].

## 2.2. Step 2: Getting rid of ones

Let $w(x)$ be a function $w\colon \mathbb{R}_{\geqslant 0} \to \mathbb{R}_{\geqslant 0}$ extending $\mathbf{w}$, defined by $w(x) := w_{\lfloor x \rfloor} \cdot (1 - \{x\}) + w_{\lfloor x \rfloor + 1} \cdot \{x\}$. Clearly, when $x$ is integral $w(x) = w_x$. For fractional $x$, the function just takes the weighted average of $w_{\lfloor x \rfloor}, w_{\lfloor x \rfloor + 1}$. The following observation is straightforward.

*Observation.* The function $w$ is convex on $\mathbb{R}_{\geqslant 1}$.

Using this observation, we can prove the following lemma, which allows us to only consider $x^*$ that does not contain 1.

**Lemma 6.** *Suppose $x^*_{\tilde{c}} = 1$ for a candidate $\tilde{c} \in A_v$. Then the approximation ratio for the voter $v$ decreases after removing $\tilde{c}$ from the fractional solution $x^*$. Formally, if $x^*_{\tilde{c}} = 1$, then*

$$\rho_v = \frac{\sum_{l=0}^{\infty} (w_l \, \mathbb{P}_{\hat{x} \sim x^*}[\sum_c \hat{x}_c \geqslant l])}{w_1 + \cdots + w_{\lfloor \tau \rfloor} + \{\tau\} \cdot w_{\lfloor \tau \rfloor + 1}} \geqslant \frac{\sum_{l=0}^{\infty} (w_l \, \mathbb{P}_{\hat{x} \sim x^*}[\sum_{c \neq \tilde{c}} \hat{x}_c \geqslant l])}{w_1 + \cdots + w_{\lfloor \tau - 1 \rfloor} + \{\tau\} \cdot w_{\lfloor \tau \rfloor}}.$$

*Proof.* It suffices to show that

$$\frac{\Delta^{\mathrm{int}}_{\tilde{c}}}{\Delta^{\mathrm{frac}}_{\tilde{c}}} := \frac{\sum_{l=0}^{\infty} (w_l \, \mathbb{P}_{\hat{x} \sim x^*}[\sum_c \hat{x}_c \geqslant l]) - \sum_{l=0}^{\infty} (w_l \, \mathbb{P}_{\hat{x} \sim x^*}[\sum_{c \neq \tilde{c}} \hat{x}_c \geqslant l])}{\left(w_1 + \cdots + w_{\lfloor \tau \rfloor} + \{\tau\} \cdot w_{\lfloor \tau \rfloor + 1}\right) - \left(w_1 + \cdots + w_{\lfloor \tau - 1 \rfloor} + \{\tau\} \cdot w_{\lfloor \tau \rfloor}\right)} \geqslant \rho_v.$$

In fact we will just show that $\frac{\Delta^{\mathrm{int}}_{\tilde{c}}}{\Delta^{\mathrm{frac}}_{\tilde{c}}} \geqslant 1$. Starting with the numerator.

$$\begin{aligned}
\Delta^{\mathrm{int}}_{\tilde{c}} &= \sum_{l=0}^{\infty} \left( w_l \cdot \left( \mathbb{P}_{\hat{x} \sim x^*}\left[ \sum_c \hat{x}_c \geqslant l \right] - \mathbb{P}_{\hat{x} \sim x^*}\left[ \sum_{c \neq \tilde{c}} \hat{x}_c \geqslant l \right] \right) \right) \\
&= \sum_{l=0}^{\infty} \left( w_l \cdot \left( \mathbb{P}_{\hat{x} \sim x^*}\left[ \sum_c \hat{x}_c \geqslant l \right] - \mathbb{P}_{\hat{x} \sim x^*}\left[ \sum_c \hat{x}_c \geqslant l + 1 \right] \right) \right)
\end{aligned}$$

[4] In [Bar+20, Lemma 2.5], $\tau$ is required to be an integer. It is however obvious from the proof that this is not necessary.

[5] Note that this reduction cannot be performed globally. It only makes sense for a single voter $v$ and his candidates $A_v$, as applying Lemma 5 to different sets $\{x^*_c\}_{c \in A_v}$ may result in different $q$ values for the same candidate.

$$= \sum_{l=0}^{\infty} \left( w_l \cdot \mathop{\mathbb{P}}_{\hat{x}\sim x^*} \left[ \sum_c \hat{x}_c = l \right] \right) = \mathop{\mathbb{E}}_{\hat{x}\sim x^*} \left[ w\left( \sum_c \hat{x}_c \right) \right] \geqslant w\left( \mathop{\mathbb{E}}_{\hat{x}\sim x^*} \left[ \sum_c \hat{x}_c \right] \right) = w(\tau) = \Delta_{\tilde{c}}^{\text{frac}}$$

Where the inequality follows from convexity of $w$. □

Hence, the fractional solution $x^*$ that minimizes the ratio $\rho_v$ of a single voter only has values 0 and $q$ where $q \in (0,1)$.

## 2.3. Step 3: Comparing Binomial and Poisson

The values $\{x_c^*\}_{c\in A_v}$ are now all either 0 or $q$ and sum up to $\tau$. It means that, when we sample $\hat{x} \sim x^*$, we now are performing $\frac{\tau}{q}$ coin tosses with an identical coin with bias $q$. The value of our integral solution which is equal to $\sum_l (w_l \cdot \mathbb{P}_{\hat{x}\sim x^*}[\sum_c \hat{x}_c \geqslant l])$ can be now rewritten as $\sum_l (w_l \cdot \mathbb{P}[\text{Bin}(\tau/q, q) \geqslant l]) = \mathbb{E}[w_1 + \cdots + w_{\text{Bin}(\tau/q,q)}]$.

Let $s_w(n) = w_1 + \cdots + w_n$. The function $s_w$ is monotone (as **w** is non-negative) and concave (as **w** is non-increasing). We recall another lemma by Barman et al.

**Lemma 7** ([Bar+20, Lemma 2.3]). *For any convex function $f$, any integer $N \geqslant 1$ and parameter $p \in [0,1]$ we have $\mathbb{E}[f(\text{Bin}(N,p))] \leqslant \mathbb{E}[f(\text{Poi}(Np))]$.*

By applying the above lemma with the function $s_{-w}(x) = -s_w(x)$ which is convex, we get that $\mathbb{E}[w_1 + \cdots + w_{\text{Bin}(\tau/q,q)}] \geqslant \mathbb{E}[w_1 + \cdots + w_{\text{Poi}(\tau)}]$ and hence

$$\rho_v \geqslant \frac{\mathbb{E}[w_1 + \cdots + w_{\text{Poi}(\tau)}]}{w_1 + \cdots + w_{\lfloor \tau \rfloor} + \{\tau\} \cdot w_{\lfloor \tau \rfloor + 1}}. \tag{2}$$

## 2.4. Step 4: Changing the mean of Poisson

A property of Poisson distributions is that sampling $X_i \sim \text{Poi}(\lambda_i)$ for $i = 1, \ldots, m$ and adding them is equivalent to sampling $Y \sim \text{Poi}(\sum_i \lambda_i)$. Hence,

$$\begin{aligned}\mathbb{E}[w_1 + \cdots + w_{\text{Poi}(\tau)}] = &\left( \sum_{l=1}^{\lfloor \tau \rfloor} \mathop{\mathbb{E}}_{y\sim \text{Poi}(l-1)} [w_{y+1} + \cdots + w_{y+\text{Poi}(1)}] \right) \\ &+ \mathop{\mathbb{E}}_{y\sim \text{Poi}(\lfloor \tau \rfloor)} [w_{y+1} + \cdots + w_{y+\text{Poi}(\{\tau\})}].\end{aligned} \tag{3}$$

We can bound each term in (3) as follows.

**Lemma 8.** *For any $l \in \{1, \ldots, \lfloor \tau \rfloor\}$, we have*

$$\mathop{\mathbb{E}}_{y\sim \text{Poi}(l-1)} [w_{y+1} + \cdots + w_{y+\text{Poi}(1)}] \geqslant \mathbb{E}[w_1 + \cdots + w_{\text{Poi}(1)}] \frac{w_l}{w_1}.$$

*Proof.* We start with the left side of the inequality

$$\mathop{\mathbb{E}}_{y\sim\mathrm{Poi}(l-1)}[w_{y+1}+\cdots+w_{y+\mathrm{Poi}(1)}] = \sum_{y=0}^{\infty}\mathbb{E}[w_{y+1}+\cdots+w_{y+\mathrm{Poi}(1)}]\cdot\mathbb{P}[\mathrm{Poi}(l-1)=y]$$

By $\mathbf{w}$ being *geometrically dominant* we get

$$\begin{aligned}
&\sum_{y=0}^{\infty}\mathbb{E}[w_{y+1}+\cdots+w_{y+\mathrm{Poi}(1)}]\cdot\mathbb{P}[\mathrm{Poi}(l-1)=y]\\
&\qquad\geqslant \mathbb{E}[w_1+\cdots+w_{\mathrm{Poi}(1)}]\cdot\sum_{y=0}^{\infty}\frac{w_{y+1}}{w_1}\mathbb{P}[\mathrm{Poi}(l-1)=y]\\
&\qquad= \mathbb{E}[w_1+\cdots+w_{\mathrm{Poi}(1)}]\cdot\frac{\mathbb{E}\left[w_{\mathrm{Poi}(l-1)+1}\right]}{w_1}\\
&\qquad\geqslant \mathbb{E}[w_1+\cdots+w_{\mathrm{Poi}(1)}]\cdot\frac{w_{\mathbb{E}[\mathrm{Poi}(l-1)+1]}}{w_1} = \mathbb{E}[w_1+\cdots+w_{\mathrm{Poi}(1)}]\cdot\frac{w_l}{w_1}.
\end{aligned}$$

The last inequality coming from convexity of $\mathbf{w}$. □

Next, we will apply a similar reasoning to the last term of (3). Specifically, we will show the following:

**Lemma 9.** $\mathbb{E}_{y\sim\mathrm{Poi}(\lfloor\tau\rfloor)}[w_{y+1}+\cdots+w_{y+\mathrm{Poi}(\{\tau\})}] \geqslant \mathbb{E}[w_1+\cdots+w_{\mathrm{Poi}(1)}]\cdot\frac{w_{\lfloor\tau\rfloor+1}}{w_1}\cdot\{\tau\}$.

Before we attempt to prove it, it will be helpful to have the following lemma at hand.

**Lemma 10.** *Let* $f(\delta)=\frac{\mathbb{E}_{k\sim\mathrm{Poi}(\delta)}[w_1+w_2+\cdots+w_k]}{\delta}$. *For any* $\delta\in(0,1]$ $f(\delta)\geqslant f(1)$.

*Proof.* We start the proof with a simple calculus:

*Observation.* For any $g\colon\mathbb{N}\to\mathbb{R}$, $\frac{\mathrm{d}\,\mathbb{E}_{k\sim\mathrm{Poi}(x)}[g(k)]}{\mathrm{d}x} = \mathop{\mathbb{E}}_{k\sim\mathrm{Poi}(x)}[g(k+1)-g(k)]$.

Using the above we analyse the function $f(\delta)=\frac{\mathbb{E}_{k\sim\mathrm{Poi}(\delta)}[w_1+w_2+\cdots+w_k]}{\delta}$. Its first derivative is

$$f'(\delta)=\frac{\mathrm{d}f}{\mathrm{d}\delta}=\frac{\mathbb{E}_{k\sim\mathrm{Poi}(\delta)}[\delta w_{k+1}]-\mathbb{E}_{k\sim\mathrm{Poi}(\delta)}[w_1+\cdots+w_k]}{\delta^2}.$$

We are going to show that $f'(\delta)\leqslant 0$ for $\delta\in(0,1]$ by focusing on the numerator $f'_{\mathrm{num}}(\delta)$ (the denominator $\delta^2$ is always going to be positive). Obviously $f'_{\mathrm{num}}(0)=0$. Moreover $\frac{\mathrm{d}f'_{\mathrm{num}}}{\mathrm{d}\delta}=\delta\mathop{\mathbb{E}}_{k\sim\mathrm{Poi}(\delta)}[w_{k+2}-w_{k+1}]$ is never greater than 0 since the sequence $\mathbf{w}$ is non-increasing. □

Now we can come back to Lemma 9.

*Proof of Lemma 9.* Again we transform the left side of desired inequality and take advantage of *geometrical dominance* and convexity of sequence $\mathbf{w}$.

$$\mathop{\mathbb{E}}_{y\sim\mathrm{Poi}(\lfloor\tau\rfloor)}\left[w_{y+1}+\cdots+w_{y+\mathrm{Poi}(\{\tau\})}\right] \geqslant \mathbb{E}\left[w_1+\cdots+w_{\mathrm{Poi}(\{\tau\})}\right]\cdot\frac{w_{\lfloor\tau\rfloor+1}}{w_1}$$

Now, by applying Lemma 10 with $\delta = \{\tau\}$,

$$\mathbb{E}\left[w_1+\cdots+w_{\mathrm{Poi}(\{\tau\})}\right]\cdot\frac{w_{\lfloor\tau\rfloor+1}}{w_1} \geqslant \mathbb{E}\left[w_1+\cdots+w_{\mathrm{Poi}(1)}\right]\cdot\{\tau\}\cdot\frac{w_{\lfloor\tau\rfloor+1}}{w_1}.$$

□

Putting together (2), (3), Lemmas 8 and 9 allows us to conclude that the ratio $\rho_v$ is at least

$$\mathbb{E}\left[w_1+\cdots+w_{\mathrm{Poi}(1)}\right]\cdot\frac{w_1+\cdots+w_{\lfloor\tau\rfloor}+\{\tau\}\cdot w_{\lfloor\tau\rfloor+1}}{w_1\left(w_1+\cdots+w_{\lfloor\tau\rfloor}+\{\tau\}\cdot w_{\lfloor\tau\rfloor+1}\right)}$$

which is equal to $\mathbb{E}[w_1+\cdots+w_{\mathrm{Poi}(1)}] = \alpha_{\mathbf{w}}$ since $w_1 = 1$. Hence, our algorithm yields an $\alpha_{\mathbf{w}}$-approximate solution.

# 3. Hardness of Approximation

The hardness of approximation proof is based on a reduction, much like those in Chapter 2. We do however want this reduction, when applied to Theorem 1.3, to prove the second part of our Theorem 3. Our reduction must therefore not only be *complete* and *sound*, but also must be aware of the size of the instance and parameter $k$. We will prove the following:

**Theorem 11**
*Let* $\mathbf{w}$ *be such that* $\lim_{i\to\infty} w_i = 0$. *For any* $\epsilon > 0$, *there exist* $\delta > 0, t \in \mathbb{N}$ *and a reduction that takes in an instance* $\mathcal{L} = (A, B, E, [L], [R], \{\pi_e\}_{e\in E})$ *of* $\textsc{lc}(\delta, t, R)$ *and produces an election* $E = (V, C, \{A_v\}_{v\in V})$ *in* $\mathbf{poly}(|C|, |V|)$ *time such that:*

- *(Completeness) If* $\mathrm{val}(\mathcal{L}) = 1$, *then there exists* $W \subseteq C$ *such that* $\mathrm{scr}_{\mathbf{w}}(W) = |V|$.
- *(Soundness) If* $\text{weak-val}(\mathcal{L}) < \delta$, *then for all* $W \subseteq C$, *we have* $\mathrm{scr}_{\mathbf{w}}(W) = (\alpha_{\mathbf{w}} + \epsilon)\cdot|V|$.
- *(Size bound)* $|C| = |A|\cdot L$ *and* $|V| = |B|\cdot t^R$.
- *(Parameter)* $k = |A|$.

Plugging in Theorem 11 to Theorems 1.2, 1.3 immediately yields the desired hardness of approximation in Theorem 3.

The rest of this section is devoted to the proof of Theorem 11. Our reduction is in fact the same as Feige's [Fei96], which can be interpreted as the inapproximability for **w**-Thiele electoral system with Chamberlin-Courant rules. Hence, on the hardness front, our main contribution is in extending the analysis to work with more general sequences of weights. To this end, we provide several helpful auxiliary lemmas in Section 3.1. Then, in Section 3.2, we describe Feige's reduction together with our generalized analysis, and prove Theorem 11.

## 3.1. Comparing Binomial and Poisson

Similar to the analysis of our algorithm, we will need to move from (generalizations of) Binomial random variables to Poisson random variables. The difference between here and the algorithmic counterpart (Section 2.3) is that the sign is flipped. Specifically, in Section 2.3, we need $\mathbb{E}[w_1 + \cdots + w_{\mathrm{Bin}(N,p)}] \geqslant \mathbb{E}[w_1 + \cdots + w_{\mathrm{Poi}(Np)}]$. In this section, we show that in fact the left-hand side is not much more than the right-hand side.

**Lemma 12.** *For any sequence* $\mathbf{w}$ *such that* $\lim_{i\to\infty} w_i = 0$, *and any* $\gamma, \lambda \in \mathbb{R}_+$ *exists a boundary* $p_0 = p_0(\mathbf{w}, \gamma, \lambda)$ *such that, for any positive* $p < p_0$ *and any positive integer* $N < \lambda/p$ *we have*

$$\mathbb{E}\Big[w_1 + \cdots + w_{\mathrm{Bin}(N,p)}\Big] < \mathbb{E}\Big[w_1 + \cdots + w_{\mathrm{Poi}(Np)}\Big] + \gamma.$$

*Proof.* We will bound the following quantity:

$$\begin{aligned}&\mathbb{E}\Big[w_1 + \cdots + w_{\mathrm{Bin}(N,p)}\Big] - \mathbb{E}\Big[w_1 + \cdots + w_{\mathrm{Poi}(Np)}\Big] \\ = &\sum_{k=1}^{\infty}\Big(\mathbb{P}[\mathrm{Bin}(N,k) = k] - \mathbb{P}[\mathrm{Poi}(Nk) = k]\Big)\cdot(w_1 + \cdots + w_k).\end{aligned} \tag{4}$$

To this end we rely on a classic result that gives an upper bound on the total variation distance between the Binomial distribution and the corresponding Poisson distribution.

**Theorem 13** ([LeC65])
*For any* $n \in \mathbb{N}$ *and* $p \in \mathbb{R}_{\geqslant 0}$, $\sum_{k=0}^{\infty} |\mathbb{P}[\mathrm{Bin}(n,p) = k] - \mathbb{P}[\mathrm{Poi}(np) = k]| < 4p$.

Let $i^*$ be the smallest positive integer such that $w_{i^*} \leqslant 0.1\gamma/\lambda$. We let $p_0 = \frac{0.01\gamma^2}{i^*\lambda}$. We will separate the summands in (4) into two parts, based on whether $k \leqslant k_0 := \lfloor \frac{i^*\lambda\cdot 10}{\gamma} \rfloor$.

We bound the first part:

$$\begin{aligned}
&\sum_{k=1}^{k_0}\Big(\mathbb{P}[\mathrm{Bin}(N,p)=k]-\mathbb{P}[\mathrm{Poi}(Np)=k]\Big)\cdot(w_1+\cdots+w_k) \\
&\qquad\qquad \leqslant k_0\sum_{k=1}^{k_0}|\mathbb{P}[\mathrm{Bin}(N,k)=k]-\mathbb{P}[\mathrm{Poi}(Nk)=k]| \\
&\textit{(From Theorem 13)}\qquad \leqslant k_0\cdot 4p \\
&\textit{(From our choice of } k_0 \textit{ and } p_0\textit{)}\qquad \leqslant 0.4\gamma.
\end{aligned} \tag{5}$$

The second part can be bounded as follows:

$$\begin{aligned}
&\sum_{k=k_0+1}^{\infty}\Big(\mathbb{P}[\mathrm{Bin}(N,p)=k]-\mathbb{P}[\mathrm{Poi}(Np)=k]\Big)\cdot(w_1+\cdots+w_k) \\
&\qquad\qquad \leqslant \sum_{k=k_0+1}^{\infty}\mathbb{P}[\mathrm{Bin}(N,p)=k]\cdot(w_1+\cdots+w_k) \\
&\textit{(From our choice of } i^*\textit{)}\qquad \leqslant \sum_{k=k_0+1}^{\infty}\mathbb{P}[\mathrm{Bin}(N,p)=k]\cdot(i^*+(k-i^*)\frac{\gamma}{10\lambda}) \\
&\textit{(From choice of } k_0,\ k\frac{\gamma}{10\lambda}>i^*\textit{)}\qquad \leqslant \sum_{k=k_0+1}^{\infty}\mathbb{P}[\mathrm{Bin}(N,p)=k]\cdot(k\frac{\gamma}{5\lambda}) \\
&\qquad\qquad \leqslant \frac{\gamma}{5\lambda}\mathbb{E}[\mathrm{Bin}(N,p)]=\frac{\gamma}{5\lambda}\cdot Np \\
&\textit{(From assumption that } Np\leqslant\lambda\textit{)}\qquad \leqslant 0.2\gamma.
\end{aligned} \tag{6}$$

□

The distribution that will turn up in our reduction is in fact a bit different from the Binomial distribution. It can be described as a *grouped* Binomial Distribution, where the outcome of $N$ coins is decided in $l$ coin-flips.

**Lemma 14.** *Let* $\mathbf{w}$ *be such that* $\lim_{i\to\infty} w_i = 0$. *For any* $p \in \mathbb{R}_+$, *and* $l$ *non-negative integers* $m_1,\ldots,m_l$, *we have*

$$\mathop{\mathbb{E}}_{z_1,\ldots,z_l\sim\mathrm{Ber}(p)}[w_1+\cdots+w_{\sum_{i=1}^{l} z_i m_i}] \leqslant \mathbb{E}[w_1+\cdots+w_{\mathrm{Bin}(\sum_{i=1}^{l} m_i,p)}].$$

We will prove a much stronger statement, similar to that of Barman et al. (which we recalled before as Theorem 7).


Figure 1: A *Hypercube Set System*. The set $P_{5,1}$ contains all vectors with 1 at fifth position, so $\mathbf{u} \notin P_{5,1}$ but $\mathbf{v} \in P_{5,1}$.

**Proposition 15.** *For any convex function $f$, any parameter $p \in [0,1]$, and $l$ non-negative integers $m_1, \ldots, m_l$, we have* $\mathbb{E}_{z_1,\ldots,z_l \sim \mathrm{Ber}(p)}[f(\sum_{i=1}^{l} z_i m_i)] \geqslant \mathbb{E}[f(\mathrm{Bin}(\sum_{i=1}^{l} m_i, p))]$.

*Proof.* Similarly to [SS07, Theorem 3.A.20], let the independent random variables $Z_1, \ldots, Z_l \sim \mathrm{Ber}(p)$ and for each $i \in [l]$, $X_i^1 = \cdots = X_i^{m_i} = Z_i$ are copies of $Z_i$. We then have

$$\begin{aligned}
\mathop{\mathbb{E}}_{z_1,\ldots,z_l \sim \mathrm{Ber}(p)}[f(\textstyle\sum_{i=1}^{l} z_i m_i)] &= \mathbb{E}\left[f\left(X_1^1 + \cdots + X_1^{m_1} + \cdots + X_l^1 + \cdots + X_l^{m_l}\right)\middle| Z_1, \ldots, Z_l\right] \\
\text{(Jensen's Inequality)} \qquad &\geqslant \mathbb{E}\left[f\left(\mathbb{E}[X_1^1 | Z_1] + \cdots + \mathbb{E}[X_1^{m_1} | Z_1] + \cdots + \mathbb{E}[X_l^{m_l} | Z_l]\right)\right] \\
&= \mathbb{E}[f(\mathrm{Bin}(\textstyle\sum_{i=1}^{l} m_i, p))].
\end{aligned}$$

□

Finally we will need a simple property of a much-used function in our analysis.

**Lemma 16.** *The function $f(z) = \mathbb{E}[w_1 + \cdots + w_{\mathrm{Poi}(z)}]$ is concave.*

*Proof.* Similarly to the proof of Lemma 10, we calculate the second derivative of $f$ and get:

$$\frac{\mathrm{d}^2 f(z)}{\mathrm{d}z^2} = \mathop{\mathbb{E}}_{k \sim \mathrm{Poi}(z)}[w_{k+2} - w_{k+1}] \leqslant 0.$$

□

## 3.2. The Reduction

We can now describe the reduction and prove Theorem 11. As we had stated earlier, our reduction is the same as that of Feige [Fei96], which employs the following gadget (see Fig 1):

**Definition 17** (Hypercube Set System [Fei96]). For $t, R \in \mathbb{N}$, the $(t,R)$-*hypercube set system* is defined as $\mathcal{H} = (\mathcal{U}, \mathcal{P}_1, \ldots, \mathcal{P}_R)$ where the universe $\mathcal{U}$ is $[t]^R$ and each $\mathcal{P}_j = (P_{j,1}, \ldots, P_{j,t})$ is a partition of $\mathcal{U}$ into $t$ parts with $P_{j,i} = \left\{\mathbf{u} = (u_1, \ldots, u_R) \in [t]^R \mid u_j = i\right\}$.

With this gadget we can describe the reduction.

**Reduction 18**
Input: An $\mathsf{LC}(\delta, t, R)$ instance $\mathcal{L} = \langle A \cup B, E, [L], [R], \{\pi_e\}_{e \in E}\rangle$.
Output: An election $\mathfrak{E} = \langle V, C, \{A_v\}_{v \in V}, k\rangle$ with $V = B \times [t]^R$, $C = A \times [L]$, and $k = |A|$.
For each vertex $b \in B$ we fix an arbitrary order on its $t$ neighbours $a_1, \ldots, a_t$. A voter $(b, \mathbf{u}) \in V$ approves of a candidate $(a_i, \sigma) \in C$ if $a_i$ is ($i$-th) neighbour of $b$, and $\mathbf{u}(\pi_{a_i,b}(\sigma)) = i$ ($\mathbf{u}$ belongs to the set $P_{\pi_{a_i,b}(\sigma),i}$ in the $(t, R)$-hypercube set system).

This reduction is tuned by parameters $t$ and $\delta$ which we can feed into Theorems 1.2 and 1.3. We pick them based on $\epsilon$ expected in Theorem 11 as follows:

- Let $i^*$ denote the smallest positive integer such that $w_{i^*} < 0.1\epsilon$, and let $\vartheta = 10\frac{i^*}{\epsilon}$.
- Let $\gamma = 0.7\epsilon$, $p_0 = p_0(\mathbf{w}, \gamma, 10\vartheta)$ be as in Lemma 12.
- We select $t = \lceil 2/p_0 \rceil$ and $\delta = \frac{0.0001\epsilon}{\vartheta^3 t^2}$.

The reduction obviously runs in $\mathbf{poly}(|C|, |V|)$ time and produces an instance of size as stated in Theorem 11.

### Completeness

Suppose that there is an assignment $\phi \colon A \to [L]$ that satisfies every $b \in B$. Pick $W = \left\{(a, \phi(a)) \mid a \in A\right\}$. We claim that $\mathrm{scr}_{\mathbf{w}}(W) \geqslant |V|$: Specifically, we will observe that every voter had voted for at least one candidate in $W$. Consider a voter $(b, \mathbf{u}) \in B \times [t]^R$ and let $a_1, \ldots, a_t$ be $b$'s (ordered) neighbours. Since $\phi$ satisfies $b$, we have $\pi_{(a_1,b)}(\phi(a_1)) = \cdots = \pi_{(a_t,b)}(\phi(a_t)) = r \in [R]$. The value $i := \mathbf{u}(r)$ tells us, which of the $t$ candidates was supported by $(b, \mathbf{u})$—it was $(a_i, \phi(a_i))$.

### Soundness

Assume by contraposition that there exists a subset $W \subseteq C$ of $k$ candidates, such that $\mathrm{scr}_{\mathbf{w}}(W) \geqslant (\alpha_{\mathbf{w}} + \epsilon) \cdot |V|$. We will show that weak-val$(\mathcal{L}) \geqslant \delta$.

For every $b \in B$, let $V[b] = \{b\} \times [t]^R$. Let also $\deg_W(b) := |W \cap (\Gamma(b) \times [L])|$, where $\Gamma(b)$ denotes the set of neighbours of $b$ in $\mathcal{L}$. For every $a \in A$, let $W_a \subseteq [L]$ denote $\left\{\sigma \in [L] \mid (a, \sigma) \in W\right\}$. Finally, let $D = 10\vartheta t$.

We divide the set $B$ into three sets:

- Let $B_1$ denote the set of $b \in B$ such that $\deg_W(b) > D$.

- Let $B_2$ denote the set of $b \in B$ such that $\deg_W(b) \leqslant D$ and, for all distinct neighbours $a_1, a_2$ of $b$ and all $\sigma_1 \in W_{a_1}, \sigma_2 \in W_{a_2}$, we have $\pi_{(a_1,b)}(\sigma_1) \neq \pi_{(a_2,b)}(\sigma_2)$.
- Let $B_3 = B \setminus (B_1 \cup B_2)$.

We will next argue that $|B_3| \geqslant \Omega_{t,\epsilon}(|B|)$, which will then allow us to 'decode' back the assignment to the Label Cover instance $\mathcal{L}$[6]. To do so, we start by bounding the contribution of $B_1$ to $\mathrm{scr}_{\mathbf{w}}(W, B_1)$—this may be seen as an argument that it does not make sense for the committee $W$ to satisfy the voters very unevenly. The main idea is that when $\deg_W(b)$ is large, the *average score per candidate* becomes small, which gives the following upper bound on the desired quantity:

**Proposition 19.** $\sum_{b \in B_1} \sum_{v \in V[b]} \mathrm{scr}_{\mathbf{w}}(W, v) \leqslant 0.2\epsilon \cdot |V|$.

*Proof.* Let us fix a vertex $b \in B_1$. Notice that each candidate $(a, \sigma) \in \Gamma(b) \times [L]$ is approved by exactly $\frac{|V[b]|}{t}$ voters in $V[b]$ (and voters in $V[b]$ do not approve any candidate outside $\Gamma(b) \times [L]$). As a result, $\mathrm{scr}_{\mathbf{w}}(W, V[b])$ is of the form $\sum_{i=1}^{\infty} r_i w_i$ for some sequence $(r_i)_{i \in \mathbb{N}}$ of non-negative integers that satisfies $\sum_{i=1}^{\infty} r_i = \deg_W(b) \cdot \frac{|V[b]|}{t}$ and $|V[b]| \geqslant r_1 \geqslant r_2 \geqslant \cdots$ ($r_1$ voters having at least one candidate in $W$, $r_2$ having at least two, etc.). It is simple to see that among such sequences $\sum_{i=1}^{\infty} r_i w_i$ is maximized when $r_1 = \cdots = r_m = |V[b]|$ for $m = \lfloor \deg_W(b)/t \rfloor$, $r_{m+1} = |V[b]| \cdot \{\deg_W(b)/t\}$ and $r_{m+2} = r_{m+3} = \cdots = 0$. In other words, $\mathrm{scr}_{\mathbf{w}}(W, V[b])$ is upper bounded by

$$\begin{aligned}
\left(\sum_{i=1}^{m} |V[b]| \cdot w_i\right) + |V[b]| \cdot \{\deg_W(b)/t\} \cdot w_{m+1} &\leqslant \left(\frac{\deg_W(b) \cdot |V[b]|}{tm}\right) \cdot \left(\sum_{i=1}^{m} \cdot w_i\right) \\
\textit{(From definition of } i^*\textit{)} \quad &\leqslant \left(\frac{\deg_W(b) \cdot |V[b]|}{tm}\right) \cdot (i^* + (m - i^*)0.1\epsilon) \\
\textit{(From our choice of } D, \vartheta;\ i^* \leqslant 0.1\epsilon m\textit{)} \quad &\leqslant \left(\frac{\deg_W(b) \cdot |V[b]|}{tm}\right)(0.2\epsilon m) \\
&= \left(\frac{\deg_W(b) \cdot |V[b]|}{t}\right)(0.2\epsilon).
\end{aligned}$$

Summing the above inequality over all $b \in B_1$, we have

$$\sum_{b \in B_1} \mathrm{scr}_{\mathbf{w}}(W, V[b]) \leqslant 0.2\epsilon \left(\sum_{b \in B_1} \frac{\deg_W(b) \cdot |V[b]|}{t}\right)$$

[6] As we will see later, the bound $\deg_W(b) \leqslant D$ for $b \in B_3$ will play the same role as number $s$ in the decoding in Lemma 2.7.

$$= 0.2\epsilon \cdot \frac{|V|}{|B| \cdot t} \cdot \sum_{b \in B_1} \deg_W(b) \leqslant 0.2\epsilon \cdot \frac{|V|}{|B| \cdot t} \cdot \sum_{b \in B} \deg_W(b)$$

$$= 0.2\epsilon \cdot \frac{|V|}{|B| \cdot t} \cdot \sum_{(a,\sigma) \in W} |\Gamma(a)| \leqslant 0.2\epsilon \cdot \frac{|V|}{|B| \cdot t} \cdot |B| \cdot t = 0.2\epsilon \cdot |V|,$$

with the last inequality following from $|W| = |A|$ and bi-regularity of the LC instance.

□

Next, we bound the score contribution from $B_2$, using the auxiliary lemmas shown in the previous subsection.

**Proposition 20.** $\sum_{b \in B_2} \mathrm{scr}_{\mathbf{w}}(W, V[b]) \leqslant (\alpha_{\mathbf{w}} + 0.7\epsilon) \cdot |V|$.

*Proof.* Let us fix $b \in B_2$. Let $a_1, \ldots, a_t$ be its (ordered) neighbours. For each $\kappa \in [R]$, let $\tau^\kappa = |\{(a,\sigma) \in W \mid a \in N(b), \pi_{(a,b)}(\sigma) = \kappa\}|$. Moreover, if $\tau^\kappa \neq 0$, let $i^\kappa$ denote the index $i \in [t]$ of the neighbour s.t. $\pi_{(a_i,b)}(\sigma_i) = \kappa$ for some $\sigma_i \in W_i$. (Due to our definition of $B_2$, there is a unique such $i$.) If $\tau^\kappa = 0$, let $i^\kappa = \perp$.

Recall that a voter $(b, \mathbf{u})$ approves a candidate $(a_i, \sigma)$ if $u_{\pi_{(a_i,b)}(\sigma)} = i$. Hence, the number of candidates in $W$ approved by $(b, \mathbf{u}) \in V_b$ is

$$\sum_{i \in [t]} \sum_{\sigma \in W_{a_i}} \mathbb{1}[u_{\pi_{(a_i,b)}(\sigma)} = i] = \sum_{\kappa \in [R]} \tau^\kappa \cdot \mathbb{1}[u_\kappa = i^\kappa].$$

Hence, we may rearrange $\mathrm{scr}_{\mathbf{w}}(W, V[b])$ as

$$|[t]^R| \cdot \mathop{\mathbb{E}}_{u_1,\ldots,u_R \sim [t]} \left[ w_1 + \cdots + w_{\sum_{\kappa \in [R]} \tau^\kappa \cdot \mathbb{1}[u_\kappa = i^\kappa]} \right].$$

Observe that for a fixed vertex $b$, $\tau^\kappa$ are non-negative integer constants such that $\sum_{\kappa \in [R]} \tau^\kappa = \deg_W(b)$. $\mathbb{1}[u_1 = i^1], \ldots, \mathbb{1}[u_R = i^R]$ are independent random variables, identically distributed by $\mathrm{Ber}(1/t)$. We may apply Lemmas 14 and 12 to get

$$\mathrm{scr}_{\mathbf{w}}(W, V[b]) \leqslant |[t]^R| \cdot \mathbb{E}\left[ w_1 + \cdots + w_{\mathrm{Poi}(\deg_W(b)/t)} \right] + \gamma.$$

By summing this over $b \in B_2$, $\sum_{b \in B_2} \mathrm{scr}_{\mathbf{w}}(W, V[b])$ is at most

$$|[t]^R| \cdot \sum_{b \in B_2} \left( \mathbb{E}\left[ w_1 + \cdots + w_{\mathrm{Poi}(\deg_W(b)/t)} \right] + \gamma \right)$$

$$\text{(From Lemma 16)} \quad \leqslant |[t]^R| \cdot |B| \cdot \left( \mathbb{E}\left[ w_1 + \cdots + w_{\mathrm{Poi}\left(\frac{\sum_{b \in B} \deg_W(b)}{|B| t}\right)} \right] + \gamma \right)$$

$$\text{(By } \textstyle\sum_{b \in B} \deg_W(b) = |B| \cdot t\text{)} \quad = (\alpha_{\mathbf{w}} + 0.7\epsilon) \cdot |V|.$$

□

**Decoding the assignment to** Label Cover From the two propositions above and our assumption that $\mathrm{scr}_{\mathbf{w}}(W) \geqslant (\alpha_{\mathbf{w}} + \epsilon) \cdot |V|$, we must have $\sum_{b \in B_3} \mathrm{scr}_{\mathbf{w}}(W, V[b]) \geqslant 0.1\epsilon \cdot |V|$. Obviously, for $b \in B_3$ we have $\mathrm{scr}_{\mathbf{w}}(W, V[b]) \leqslant \deg_W(b) \cdot \frac{|V_b|}{t}$, which is at most $\frac{D \cdot |V|}{t \cdot |B|}$ for $b \in B_3$. Thus,

$$|B_3| \geqslant \frac{0.1\epsilon \cdot |V|}{D \cdot |V|/(t \cdot |B|)} = \frac{0.1\epsilon t|B|}{D} = \frac{0.01\epsilon|B|}{\vartheta}. \tag{7}$$

Now, consider the following (random) assignment: for every $a \in A$ such that $W_a \neq \emptyset$, let $\phi(a)$ be a random element of $W_a$. By definition, each $b \in B_3$ has distinct neighbours $a_1, a_2$ and $\sigma_1 \in W_{a_1}, \sigma_2 \in W_{a_2}$ such that $\pi_{(a_1,b)}(\sigma_1) = \pi_{(a_2,b)}(\sigma_2)$. Hence, the probability that $\phi$ weakly satisfies such $b$ is at least $\frac{1}{|W_{a_1}| \cdot |W_{a_2}|} \geqslant \frac{1}{D^2}$, where the inequality follows from $\deg_W(b) \leqslant D$. Thus,

$$\mathop{\mathbb{E}}_{\phi}[\text{weak-val}(\phi)] \geqslant \frac{|B_3|}{|B| \cdot D^2} \overset{(7)}{\geqslant} \frac{0.01\epsilon}{\vartheta \cdot D^2} = \frac{0.0001\epsilon}{\vartheta^3 t^2} = \delta.$$

This completes our proof of Theorem 11.

# 4. Determining performance of the Greedy Algorithm

As touched upon earlier, an important and natural algorithm for all covering problems is the greedy algorithm, in which at each of $k$ steps we select the set (candidate) maximising the utility increase. As we discussed, this algorithm not only is very efficient, but also provides tight approximation ratio for basic variants of the problem. It is hence tempting to question, if an LP-based algorithm like the one we presented in Section 2 does give any real improvement over the greedy algorithm, or is the gain in ratio only thanks to more thorough analysis? We answer this question by giving—for any *geometrically dominant* sequence $\mathbf{w}$ other than *geometric*—an instance for which the ratio of our algorithm is strictly larger than that of the greedy.

**Proposition 21.** *For any* geometrically dominant $\mathbf{w} = (w_1 = 1, w_2 = p, w_3, w_4, \dots)$,

$$\alpha_{\mathbf{w}} \geqslant \frac{1}{1-p} \cdot \left(1 - \frac{1}{e^{1-p}}\right).$$

*Proof.* $\alpha_{\mathbf{w}}$ is lower-bounded by $\alpha_{\mathbf{v}}$ for $\mathbf{v} = (1, p, p^2, \dots)$.

$$\alpha_{\mathbf{v}} = \mathop{\mathbb{E}}_{k \sim \mathrm{Poi}(1)}\left[\sum_{i=1}^{k} p^{i-1}\right] = \mathop{\mathbb{E}}_{k \sim \mathrm{Poi}(1)}\left[\frac{1-p^k}{1-p}\right] = \frac{1}{1-p}\left(1 - \mathop{\mathbb{E}}_{k \sim \mathrm{Poi}(1)}[p^k]\right) = \frac{1}{1-p}\left(1 - e^{p-1}\right),$$

where the last equality follows from the well-known fact that the probability-generating function (PGF) of $\mathrm{Poi}(\lambda)$ is $G(z) = e^{\lambda(z-1)}$. □

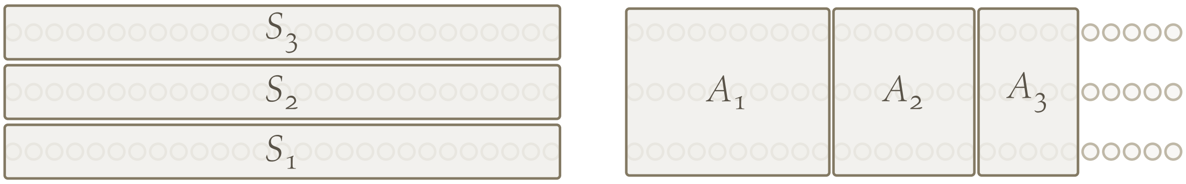



Figure 2: Tight instance for the greedy algorithm. The universe of $m \cdot k$ voters is covered by $2k$ candidates. The optimal solution is to pick $S_1, \dots, S_k$ and give every voter exactly one supported candidate in the committee. The greedy algorithm will pick $A_1, \dots, A_k$ instead. Compared to the tight example one would build for Maximum Coverage, here the sets $A_i$ must be larger, because even after picking $A_1, \dots A_l$ the sets $S_i$ still gain some score from the already "covered" voters.

Clearly, when there is an element $i$ of $\mathbf{w}$ with $w_i \cdot w_{i+2} > w_{i+1}^2$, the inequality in Proposition 21 becomes strict, as $\mathbf{v} < \mathbf{w}$. Let us now analyse the greedy algorithm for these $\mathbf{w}$-Thiele rules.

**Theorem 22**
*For any Thiele sequence $\mathbf{w} = (1, p, w_3 \dots)$, the approximation ratio of the greedy algorithm for $\mathbf{w}$-Thiele is at most $\frac{1}{1-p} \cdot \left(1 - \frac{1}{e^{1-p}}\right)$.*

We will prove this theorem by building a specific instance. The instance is defined below (see also Figure 2), and its construction is parametrised by numbers $m$, $k$ (the committee size), and $p$ (the second element of $\mathbf{w}$).

- The set of voters $V = \{v_{i,j}\}_{i \in [k], j \in [m]}$.
- One collection of $k$ candidates is $\{S_i\}_{i \in [k]}$. The candidate $S_i$ is supported by the voters $\{v_{i,j}\}_{j \in [m]}$. Every voter voting for exactly one $S_i$ gives
$$\mathrm{scr}_{\mathbf{w}}(\{S_1, \dots, S_k\}) = m \cdot k.$$
- An alternative collection of candidates is $\{A_i\}_{i \in [k]}$. $A_i$ is supported by $|A_i| = m \cdot \left(1 - \frac{1-p}{k}\right)^{i-1}$ voters. Moreover the overlap between $S_i$ and $A_l$ (the number of voters supporting both candidates) is equal to
$$|S_i \cap A_l| = \frac{|A_l|}{k} = \frac{m}{k}\left(1 - \frac{1-p}{k}\right)^{l-1}.$$
No voter is supporting both $A_i$ and $A_l$ for $i \neq l$ ($A_i \cap A_l = \varnothing$).

**Proposition 23.** *For any $l \in \{0,\dots,k-1\}$ if the candidates $A_1,\dots,A_l$ have been picked into the committee, the greedy algorithm will select $A_{l+1}$.*

*Proof.* To verify this we need to compare $\mathrm{scr}_{\mathbf{w}}(A_{l+1} \mid A_1,\dots,A_l)$—a score gain from picking $A_{l+1}$ once $A_1,\dots,A_l$ have been selected—to $\mathrm{scr}_{\mathbf{w}}(S_i \mid A_1,\dots,A_l)$ (for any $i \in [k]$). Since the electorates of $A_i$ candidates are pairwise disjoint, the former is very simple:

$$\mathrm{scr}_{\mathbf{w}}(A_{l+1} \mid A_1,\dots,A_l) = |A_{l+1}| = m \cdot \left(1 - \frac{1-p}{k}\right)^l.$$

To calculate the latter we notice that some of $S_i$'s voters are already satisfied by the already picked candidates. Those voters will only earn an additional score of $p$. The other voters supporting $S_i$ will gain 1 when he is picked.

$$\begin{aligned}\mathrm{scr}_{\mathbf{w}}(S_i \mid A_1,\dots,A_l) &= p \cdot |S_i \cap (A_1 \cup \dots \cup A_l)| + |S_i \setminus (A_1 \cup \dots \cup A_l)| \\ = p \cdot \frac{|A_1| + \dots + |A_l|}{k} + m - \frac{|A_1| + \dots + |A_l|}{k} &= m - \frac{1-p}{k}(|A_1| + \dots + |A_l|) \\ = m - m\frac{1-p}{k}\sum_{j=1}^{l}\left(1 - \frac{1-p}{k}\right)^{j-1} &= m \cdot \left(1 - \frac{1-p}{k}\right)^l.\end{aligned}$$

□

We have thus established that the greedy algorithm will indeed select $A_1,\dots,A_k$[7]. We also need to compute the score of the greedy solution,

$$\mathrm{scr}_{\mathbf{w}}(\{A_1,\dots,A_k\}) = |A_1| + \dots + |A_k| = m \cdot \left(1 - \left(1 - \frac{1-p}{k}\right)^k\right)\frac{k}{1-p},$$

which allows us to bound the approximation ratio of the greedy algorithm:

$$\frac{\mathrm{scr}_{\mathbf{w}}(\{A_1,\dots,A_k\})}{\mathrm{scr}_{\mathbf{w}}(\{S_1,\dots,S_k\})} = \frac{\frac{m \cdot k}{1-p}\left(1 - \left(1 - \frac{1-p}{k}\right)^k\right)}{m \cdot k} \xrightarrow[k \to \infty]{} \frac{1}{1-p} \cdot \left(1 - \frac{1}{e^{1-p}}\right).$$

# 5. Notes

The *pipage rounding* method has been introduced by Ageev and Sviridenko [AS04] with an application of optimising a function over bipartite matching constraints. It

[7] One may break ties by adding $k$ additional voters $u_1,\dots,u_k$ in the instance. $u_i$ would only approve the candidate $A_i$. This increases the score of the greedy solution only by $k$, with no impact on the approximation ratio.

was later extended by Călinescu, Chekuri, Pál and Vondrák [Căl+07] to any matroid constraints. In our algorithm in Section 2 we are applying this powerful method to a simple cardinality constraint, but in fact it could be replaced with any matroid constraint without effect on the approximation ratio.

The algorithm of Barman, Fawzi, Ghosal and Gurpinar [Bar+20] for the Multi $\ell$-Coverage problem is also using the *pipage rounding* framework. Their hardness is based on Unique Game Conjecture—an assumption we are able to get rid of with our reduction.

The results in this chapter are based on our paper with Szymon Dudycz, Pasin Manurangsi and Krzysztof Sornat, which appeared at IJCAI 2020 [Dud+20]. We left unanswered the question of approximation of **w**-Thiele rules with **w** sequence not *geometrically dominant*. This has been subsequently resolved by Barman, Fawzi and Fermé [BFF21].

# 4 k-Medians clustering with capacities

In this chapter we will focus on a topic of metric clustering. In the clustering problems we are given a discrete metric (defined by a distance function $d$ obeying the triangle inequality) and a number $k$. Our task is to select $k$ points (centres) in the metric and assign every other point to one of the $k$ clusters identified by the selected representatives. The objective may be to minimise a maximum distance from a point to its representative (this problem is called $k$-Center), the sum of distances from the points to their respective centres ($k$-Median) or the sum of squared distances ($k$-Means).

A common application is categorising data which is usually represented by long vectors within the metric defined by the Hamming distance. For $k$-Means, Data Scientists would typically use a library function implementing some local-search heuristic, like Lloyd's algorithm [Llo82] initialised with a seed given by a procedure called *k-means++* [AV07]. This algorithm—although practical—only gives an approximation guarantee of $\mathcal{O}(\log k)$.

Our Capacitated $k$-Median (`CkM`) problem adds another difficulty to the $k$-Median: Every point in the metric has an associated capacity and at most that many members may belong to its cluster. We describe it in the language of Facility Location.

**Problem 1** (Capacitated $k$-Median)
Given a metric $(\mathcal{F} \cup \mathcal{C}, d)$, and for every facility $f \in \mathcal{F}$ a capacity $u_f$, find a subset $S \subset \mathcal{F}$ of size $k$ and an assignment of clients $\phi\colon \mathcal{C} \to S$ such that:

- For every facility $f \in S$, its capacity is respected, i.e. $|\phi^{-1}(f)| \leqslant u_f$,
- The total distance from clients to their facilities, i.e.

  $$d(\phi) := \sum_{c \in \mathcal{C}} d(c, \phi(c)),$$

  is minimised over all possible choices of $S$ and $\phi$.

An instance of CkM is *uniform* if the capacities are equal for all the facilities.

*Remark*. The problem becomes marginally harder in a *client-weighted* variant, when each client location $i \in \mathcal{C}$ has cardinality $c_i$ of clients, which can be directed to different facilities. The additional complexity comes from the fact, that $c_i$ may be large compared to the size of the metric.

# 1. Related work

**Problems without capacities** There is an extensive body of work on approximability of the clustering problems: For the $k$-Center problem a simple local search procedure gives 2-approximation [Gon85], which cannot be improved unless **P** = **NP** [HN79]. Also for $k$-Means and $k$-Median, the first constant-factor approximation algorithms were using this method—giving the ratios of $(9+\varepsilon)$ [Kan+02] and $(3+\varepsilon)$ [Cha+99; Ary+04] respectively. Stronger results based on linear programming, and taking advantage of the similarity between clustering and Facility Location, have been attained later. Currently the best ratio for $k$-Means is 6.357 [Ahm+17], and for $k$-Median–$(2.675+\varepsilon)$ [LS13; Byr+15].

**Approximability of CkM** Capacitated $k$-Median is among few remaining natural optimisation problems for which it is not yet settled, whether a polynomial time algorithm may achieve a constant approximation ratio. The only proper approximation algorithm, which we describe in Section 3, has the ratio of $\mathcal{O}(\log k)$[1]. The difficulty lies with the natural linear programming formulation, which has an unbounded integrality gap (the ratio between the optimum fractional and integral solutions). All the known approximation algorithms overcome this problems by 'cheating': they either open more than $k$ facilities, or violate the capacities (but the comparison is made with an optimum solution which opens exactly $k$ facilities and does not violate the capacities). There is a long series of results fitting this pattern: Starting with 16-approximation algorithm for the *uniform* instances [Cha+99] that may violate the capacities by the factor of 3. Then, for general instances Chuzhoy and Rabani gave a 50-approximation algorithm violating the capacities 40 times [CR05]. Li used stronger LP-relaxations to obtain a constant-factor approximation algorithm which opens $(1+\varepsilon)$ facilities [Li15; Li16]. Similar ideas have been used to construct a constant-factor approximation which violates the capacities by a factor of $(1+\varepsilon)$ [BRU16; DL16].

[1] To the best of our knowledge, it had never been published before our paper came out, but it has been a known application of metric tree-embeddings, similar to the $\mathcal{O}(\log k)$-approximation for $k$-Median by Charikar et al. [Cha+98]. We are framing it in the language of our main tool.

**Our results** We cheat in a different way. Rather than violating the cardinality constraint or the capacities, we abuse the time complexity, by constructing a constant-factor approximation algorithm for CkM which runs in $\mathbf{FPT}(k)$ time.

**Theorem 2**
*For any $\epsilon > 0$ there is a $(7+\epsilon)$-approximation algorithm for* Capacitated $k$-Median *(Problem 1) running in time $2^{\mathcal{O}(k \log k)} \cdot \mathbf{poly}(|\mathcal{F}| + |\mathcal{C}|)$.*

This result makes sense from the point of view of parametrised complexity, as the $k$-Median problem (even without capacities) is $\mathbf{W}[2]$-hard, which makes it unlikely to admit exact $\mathbf{FPT}(k)$ algorithms.

**Proposition 3.** *The $k$-*Median *problem is $\mathbf{W}[2]$-hard when parametrised by $k$.*

*Proof.* In the Dominating Set problem we are given a graph $G$ and a parameter $k$, and must decide, whether there is a $S$ subset of $k$ vertices dominating entire graph (every vertex must either belong or have a neighbour in $S$). The Dominating Set problem is $\mathbf{W}[2]$-hard with respect to the parameter $k$ (see e.g. [Cyg+15]).

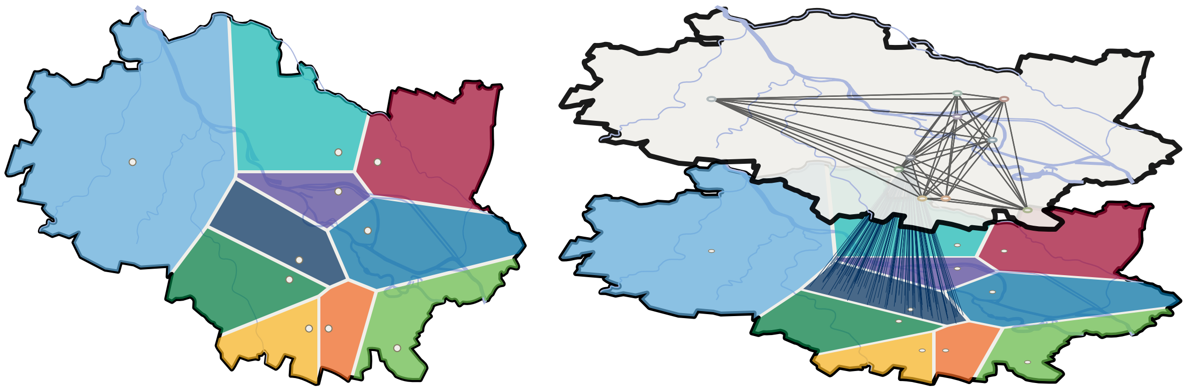
We construct $k$-Median instance as follows: let $\mathcal{F} = \mathcal{C} = V(G)$. The metric is induced by shortest paths in $G$. A dominating set of size $k$ in $G$ is equivalent to a $k$-Means solution of total cost equal to $|V(G)| - k$. □

Our algorithm from Theorem 2 is composed of two ingredients. The first is a reduction that allows us to solve the problem on simpler metrics called $l$-centred rather than general metrics. We describe this reduction in Section 2. In Section 3 we will familiarise ourselves with the $l$-centred instances by using them to construct a simple $\mathcal{O}(\log k)$-approximation algorithm for CkM. The second ingredient is a parametrised algorithm for CkM on the $l$-centred instances. In Section 4.2 we will present its simpler form—an exact algorithm for uniform instances. Finally in Section 4.3 we extend it to work on general instances (and pay a marginal price in the approximation ratio).

# 2. $l$-Centred instances

The $l$-centred metric is an alternative distance function defined on the same universe $\mathcal{F} \cup \mathcal{C}$. We start off by selecting a set $S \subset \mathcal{F}$ of $l$ facilities—a solution to $l$-Median (for a number $l \geqslant k$) on the universe $\mathcal{F} \cup \mathcal{C}$ (ignoring the capacities). This solution naturally determines a mapping $\phi^S : \mathcal{C} \cup \mathcal{F} \to S$, with every point mapped to the closest facility of $S$ (see Fig. 1 (a)).

The $l$-centred distance function $d^S$ will have two layers. For any $f, g \in S$ it is identical to the original metric, i.e. $d^S(f,g) = d(f,g)$. To travel anywhere from a point



(a) A set of $S \subset \mathcal{F}$ determines a mapping $\phi^S$ corresponding to the Voronoi diagram on $\mathcal{F} \cup \mathcal{C}$.

(b) To go from $a$ to $b$ in the $l$-centred metric, one must first go to $a$'s centre (to the top layer), then follow the original metric to the $b$'s centre and from there to $b$.

Figure 1: $l$-Centred metric defined by a set $S$ of Wrocław's hospitals.

$a \notin S$, we first route to $a$'s centre $\phi^S(a)$. Generally the new metric is defined as follows (see Figure 1 (b)):

$$d^S(a,b) := d(a, \phi^S(a)) + d(\phi^S(a), \phi^S(b)) + d(\phi^S(b), b). \tag{1}$$

The usefulness of $d^S$ follows from the fact, that it is a *metric embedding* of $d$.

**Proposition 4.** *For any $a, b \in \mathcal{F} \cup \mathcal{C}$, the inequalities hold*

$$d(a,b) \leqslant d^S(a,b) \leqslant 3d(a,b) + 4d(a, \phi^S(a)).$$

*Proof.* The left inequality is a trivial consequence of the triangle inequality. We focus on the right inequality:

$$\begin{aligned}
d^S(a,b) &= d(a,\phi^S(a)) + d(\phi^S(a), \phi^S(b)) + d(\phi^S(b), b) \\
\text{(By triangle inequality)} \quad &\leqslant d(a,\phi^S(a)) + d(\phi^S(a), a) + d(a,b) + d(b, \phi^S(b)) \\
&\quad + d(\phi^S(b), b) \\
\text{(Since } d(b,\phi^S(b)) \leqslant d(b,\phi^S(a))\text{)} \quad &\leqslant 2d(a,\phi^S(a)) + d(a,b) + 2d(b,\phi^S(a)) \\
\text{(By triangle inequality)} \quad &\leqslant 2d(a,\phi^S(a)) + d(a,b) + 2\Big(d(b,a) + d(a,\phi^S(a))\Big).
\end{aligned}$$

□

The cost on any solution $\phi$ for CkM can be measured on the metric $d^S$ instead of $d$. By summing over pairs $\left\langle c, \phi^S(c) \right\rangle$ for all clients $c \in \mathcal{C}$, Proposition 4 shows us, that the loss incurred by this swapping of the underlying metric depends on the quality of the solution $S$, which we used to define that metric.

**Corollary.** *For any mapping $\phi\colon \mathcal{C} \to \mathcal{F}$ we have*

$$d(\phi) \leqslant d^S(\phi) \leqslant 3d(\phi) + 4d(\phi^S), \tag{2}$$

*where $d(\varphi) = \sum_{c \in C} d(c, \varphi(c))$ is the total cost of mapping $\varphi$ on metric $d$.*

Fortunately, since the solution $S$ ignores the capacities, we can ensure that its cost is a constant approximation of the optimum solution to the Capacitated $k$-Median on $\mathcal{F} \cup \mathcal{C}$. Additionally we can count on a better approximation by picking $l$ significantly larger than $k$.

Altogether, we reduce the Capacitated $k$-Median on general metric $d$ to two different problems: The $k$-Median (without capacities) with the cardinality constraint relaxed from $k$ to $l$, and the CkM on the $l$-centred metric.

**Lemma 5.** *For any metric $d$ over the universe $\mathcal{F} \cup \mathcal{C}$ and capacities $\{u_f\}_{f \in \mathcal{F}}$ let $\phi^{OPT}$ denote the optimum solution to CkM on that instance. Let $S \subset \mathcal{F}$ with $\phi^S$ picking the closest centre in $S$ for every point (according to the metric $d$), and let $l = |S|$. Let also $\phi^{OPT(S)}$ denote the optimum solution to CkM on metric $d^S$.*
*If a solution $\phi$ respects the capacities, and satisfies*

$$d^S(\phi) \leqslant \alpha \cdot d^S(\phi^{OPT(S)}) \tag{Asm. 1}$$

*($\phi$ is an $\alpha$-approximation for CkM on the $l$-centred metric $d^S$), while $\phi^S$ satisfies*

$$d(\phi(S)) \leqslant \beta \cdot d(\phi^{OPT}) \tag{Asm. 2}$$

*(which holds when it is a $\beta$-approximation for $k$-Median on the metric $d$), then*

$$d(\phi) \leqslant \alpha \cdot (3 + 4\beta)\, d(\phi^{OPT})$$

*($\phi$ is an $\alpha \cdot (3 + 4\beta)$-approximation for CkM on metric $d$).*

*Proof.* Starting from the left side:

$$\begin{aligned} d(\phi) &\overset{(2)}{\leqslant} d^S(\phi) \overset{\text{(Asm. 1)}}{\leqslant} \alpha \cdot d^S(\phi^{\text{OPT}(S)}) \overset{(\clubsuit)}{\leqslant} \alpha \cdot d^S(\phi^{\text{OPT}}) \\ &\overset{(2)}{\leqslant} \alpha \cdot \left(3d(\phi^{\text{OPT}}) + 4d(\phi^S)\right) \overset{\text{(Asm. 2)}}{\leqslant} \alpha \cdot \left(3d(\phi^{\text{OPT}}) + 4\beta \cdot d(\phi^{OPT})\right), \end{aligned}$$

where the inequality ($\clubsuit$) follows from optimality of the solution $\phi^{\text{OPT}(S)}$ for CkM on metric $d^S$. □

# 3. Polynomial-time $\mathcal{O}(\log k)$-approximation for CkM

In this section we will show, how $l$-centred metrics can be used to obtain a $\mathcal{O}(\log k)$-approximation for CkM. As we have already said, constant-factor approximation algorithms do exist for $k$-Median (without capacities), with currently best ratio of $\beta = 2.675 + \varepsilon$. By plugging this $\beta$-approximation into Lemma 5, it is sufficient to have a $\mathcal{O}(\log k)$-approximation for CkM on $l$-centred metrics (with $l = k$).

A standard tool used to provide such an approximation in a metric setting is a randomised algorithm of Fakcharoenphol, Rao and Talwar (FRT) [FRT03] which embeds an arbitrary metric onto a tree.

**Theorem 6** (FRT [FRT03])
*For any metric $d$ over the ground set $X$, a randomised, polynomial-time (in $|X|$) procedure finds a metric $d^{\mathcal{T}}$ over the ground set $\mathcal{T} \supset X$, such that:*

- *$d^{\mathcal{T}}$ is a shortest-path metric on a weighted tree with vertex set $T$; $X$ are the leaves in that tree,*
- *For any $a, b \in X$,*
$$d(a,b) \leqslant d^{\mathcal{T}}(a,b),$$
*(so $d^{\mathcal{T}}$ is indeed a metric embedding of $d$),*
- *For any $a, b \in X$, the expected distortion—with respect to the random choice of the metric $\mathcal{T}$ by the algorithm—is bounded:*
$$\mathop{\mathbb{E}}_{\mathcal{T} \sim \mathbf{FRT}(X,d)} \left[ d^{\mathcal{T}}(a,b) \right] \leqslant \rho \cdot d(a,b),$$
*where $\rho = \mathcal{O}(\log |X|)$.*

We will apply Theorem 6 to the metric $d$ over the ground set $S$—the upper layer of our $l$-centred metric $d^S$. This new metric $d^{\mathcal{T}}$ can replace $d$ at the upper layer of $d^S$. To that end we define

$$d^T(a,b) := d(a, \phi^S(a)) + d^{\mathcal{T}}(\phi^S(a), \phi^S(b)) + d(\phi^S(b), b),$$

for any $a, b \in \mathcal{F} \cup \mathcal{C}$. The metric $d^T$ defined over the set $\mathcal{F} \cup \mathcal{C} \cup \mathcal{T}$, is induced by shortest paths on an edge-weighted tree. Let now $\phi^{\mathrm{OPT}(T)}$ denote the optimum solution co CkM on that metric. It turns out, that $\phi^{\mathrm{OPT}(T)}$ is an $\mathcal{O}(\log k)$-approximation for CkM on the original metric $d$.

**Proposition 7.** *The expected cost of $\phi^{OPT(T)}$—with respect to the random choice of $T$ in Theorem 6—measured on metric $d$ is bounded by*

$$\mathop{\mathbb{E}}_{T\sim\mathbf{FRT}(S,d)}\left[d(\phi^{OPT(T)})\right] \leqslant \mathop{\mathbb{E}}_{T\sim\mathbf{FRT}(S,d)}\left[d^{S,T}(\phi^{OPT(T)})\right] \leqslant \rho\cdot d^S(\phi^{OPT(S)}),$$

*where $\phi^{OPT(S)}$ is the optimum solution to CkM on the $l$-centred metric $d^S$, and $\rho = \mathcal{O}(\log k)$ comes from Theorem 6.*

*Proof.* The first inequality is given by the fact that $d^T$ is a metric embedding of $d$. Since $\phi^{\mathrm{OPT}(T)}$ is optimal on the metric $d^T$, we have

$$\mathop{\mathbb{E}}_{T\sim\mathbf{FRT}(S,d)}\left[d^T(\phi^{\mathrm{OPT}(T)})\right] \leqslant \mathop{\mathbb{E}}_{T\sim\mathbf{FRT}(S,d)}\left[d^T(\phi^{\mathrm{OPT}(S)})\right].$$

Furthermore, by summing over all clients $c\in\mathcal{C}$,

$$\mathop{\mathbb{E}}_{T\sim\mathbf{FRT}(S,d)}\left[d^T(\phi^{\mathrm{OPT}(S)})\right] = \sum_{c\in\mathcal{C}}\mathop{\mathbb{E}}_{T\sim\mathbf{FRT}(S,d)}\left[d^T(c,\phi^{\mathrm{OPT}(S)}(c))\right] \leqslant \sum_{c\in\mathcal{C}}\rho\cdot d^S(c,\phi^{\mathrm{OPT}(S)}(c)).$$

□

## 3.1. Solving CkM on trees

We are now left with the task of finding the optimum solution to CkM on an edge weighted tree $T$ (with a weight function $d^T$). We can additionally assume that the tree is binary and all the facilities and clients lie in the leaves (which—if it were not true already—could be ensured by at most trebling the number of nodes).

Consider a subtree $t$ dangling on an edge $e_t$, and imagine we have decided, which facilities shall be opened inside of that subtree (let this set be denoted as $R[t]\subset\mathcal{F}\cap t$). The total capacity of that subtree is equal to $u(R[t]) = \sum_{f\in R[t]} u_f$. If the number of clients inside $t$ is larger than the total capacity, $t$ has a negative balance of $|\mathcal{C}\cap t| - u(R[t])$ which must be routed up through the edge $e_t$. Each such client must pay $d^T(e_t)$ for passing this edge. If the subtree $t$ has a sufficient capacity to serve all its clients, it has a positive balance, and up to $u(R[t]) - |\mathcal{C}\cap t|$ clients may be rooted down through the edge $e_t$.

This reasoning lays out the dynamic programming for us. For any subtree $t$, number $k'\in\{0,\dots,k\}$ and the balance $b\in\{-|\mathcal{C}|,\dots,|\mathcal{C}|\}$, we define $\mathsf{dp}[t,k',b]$ to be the minimum cost of opening exactly $k'$ facilities inside $t$ and routing exactly $b$ clients down through $e_t$ ($b<0$ meaning, we are routing exactly $-b$ clients up). The cost of routing is counted to the top of the edge $e_t$, so it includes this edge.

For a leaf $c \in \mathcal{C}$, $\mathsf{dp}[c,0,-1] = d^T(e_c)$. We set $\mathsf{dp}[c,k',b] = \infty$ for every other value of $k'$ and $b$, as they are infeasible. Similarly, for a leaf $f \in \mathcal{F}$ we set

$$\mathsf{dp}[f,k',b] = \begin{cases} 0 & \text{if } k' = 0, \\ b \cdot d^T(e_f) & \text{if } k' = 1 \text{ and } b \in [0,u_f], \\ \infty & \text{otherwise.} \end{cases}$$

Finally, for $t$ with two subtrees $t_1$, $t_2$, computing $\mathsf{dp}[t,k',b]$ amounts to finding $k_1'$ and $b_1$ which minimize

$$\mathsf{dp}[t_1,k_1',b_1] + \mathsf{dp}[t_2,k'-k_1',b-b_1].$$

This can be trivially done by iterating over all $\mathcal{O}(k \cdot |\mathcal{C}|)$ choices. Once such a pair is found, we set

$$\mathsf{dp}[t,k',b] = \mathsf{dp}[t_1,k_1',b_1] + \mathsf{dp}[t_2,k'-k_1',b-b_1] + d^T(e_t) \cdot |b|.$$

At the root we can read the optimum solution to CkM on entire tree $T$, as it is equal to

$$\min\left\{\mathsf{dp}[T,k'0] \mid k' \in [k]\right\}.$$

# 4. Constant-factor approximation

In this section we will present the algorithm for Capacitated $k$-Median on $l$-centred metrics. The algorithm is a composition of ideas which we are going to gradually introduce—first, in Section 4.2, we describe the exact algorithm for the Uniform CkM, and only then we will put together the $(1+\epsilon)$-approximation algorithm for the non-uniform variant. Both algorithms will guess the $k$ facilities to open, and rely on a subroutine which (in polynomial time) optimally maps clients to the opened facilities. We start by describing this subroutine.

## 4.1. Mapping clients to open facilities

We are given a metric $d$ over the set $\mathcal{C} \cup \mathcal{F}^{\text{open}}$, in which we need to find the optimal mapping $\phi\colon \mathcal{C} \to \mathcal{F}^{\text{open}}$. Were there no capacities associated with the facilities, the task would be very simple—we would just map every client to the closest facility. It turns out however, that we can accomplish that task in polynomial time, even with the capacities present.

Intuitively, mapping clients $\mathcal{C}$ to $\mathcal{F}^{\text{open}}$ is very similar to minimum-cost $b$-matching in bipartite graphs. Every client has $b = 1$ (or $c_i$ in *client-weighted* variant of the problem), while every facility can be matched up to $u_f$ times. $b$-MATCHING can be found with LP-solvers, which is how we will find our mapping. We formulate our problem as an integer program:

$$\begin{aligned} \text{minimize} \quad & \sum_{c\in\mathcal{C}} \sum_{f\in\mathcal{F}^{\text{open}}} d(c,f)\cdot x_{c,f} \\ \text{subject to} \quad & \sum_{f\in\mathcal{F}^{\text{open}}} x_{c,f} = 1 \qquad \forall c\in\mathcal{C}, \\ & \sum_{c\in\mathcal{C}} x_{c,f} \leqslant u_f \qquad \forall f\in\mathcal{F}^{\text{open}}, \\ & x_{c,f}\in\{0,1\}. \end{aligned} \tag{3}$$

(In our integer program, $x_{c,f} = 1$ represents that $\phi(c) = f$.) Let us consider the linear-programming relaxation of (3), where the last constraint is replaced with $x_{c,f} \geqslant 0$. The constraint matrix of this LP is an incidence matrix of a complete bipartite graph on $\mathcal{C} \uplus \mathcal{F}^{\text{open}}$ and is known to be *totally unimodular*, which means that the relaxed LP has an optimal solution with integral values of $x_{c,f}$.

*Remark*. The solution is integral regardless of the right side of the inequalities in (3) (as long as they are integers), so LP-solving (which can be done in polynomial time) gives us the optimal mapping even in the *client-weighted* variant of CkM.

## 4.2. Solving Uniform CkM on $l$-centred metrics

Our parametrised algorithm for UNIFORM CkM is based on a simple observation, that once we know the number $k_s$ of facilities to open in a particular Voronoi cell $s \in S$ of the $l$-centred metric, then we can choose them greedily to be the closest $k_s$ facilities to the centre of the cell (see Figure 2 (a)). The reason is, that every client (even ones in the same cell) must connect to the facilities through the centre. It thus suffices to scan all configurations $(k_s)_{s\in S}$ with $\sum_{s\in S} k_s = k$, for each one open the facilities greedily and use our subroutine from Section 4.1 to find the mapping of clients to the facilities. There are at most $l^k$ possible configurations—some of them may be infeasible, when $k_s$ is larger then the number of facilities available in the cell $s$, but we may just disregard them when scanning over the configurations.

**Corollary.** UNIFORM CAPACITATED $k$-MEDIAN *on $l$-centred instances can be solved exactly in time* $l^k \cdot \mathbf{poly}(|\mathcal{F} \cup \mathcal{C}|)$.


(a) In the algorithm for Uniform CkM we guess the number of open facilities in every cell (2 in figure) and pick the closest facilities to the centre. This minimises the distance (as every client is routed through the centre in $l$-centred metric).

(b) In the general (non-uniform) case, every cell is divided into rings of geometrically increasing radii. We guess the number of open facilities in every ring, and pick the ones with largest capacity (facilities in the same ring have almost equal distance to the centre).

Figure 2: Determining the facilities to open.

## 4.3. Approximating non-uniform CkM

In general (non-uniform) case, our greedy facility-opening strategy will not suffice. Once we know, how many facilities need to be open in the cell $s \in S$, it might make sense to open a facility further from the centre, if it has higher capacity. Our configurations will therefore need to be more granular.

We begin by guessing the largest distance $d^S(c, \phi^{\mathrm{OPT}(S)}(c))$ between the client and its facility in the optimum solution (for the $l$-centred metric $d^S$). This quantity will be denoted as $D$. There are at most $|\mathcal{C}| \cdot |\mathcal{F}|$ choices for $D$. Consider the set $\mathcal{F}[s]$ of facilities in a single Voronoi cell $s \in S$. We will partition it into $B+1$ (with $B = \left\lceil \log_{1+\epsilon} \frac{|\mathcal{C}|}{\epsilon} \right\rceil$) *rings* $\mathcal{F}^0[s], \mathcal{F}^1[s], \ldots, \mathcal{F}^B[s]$ of geometrically growing diameters:

$$\mathcal{F}^i[s] := \begin{cases} \left\{ f \in \mathcal{F}[s] \mid d(s,f) \in [0, (1+\epsilon)^{-i} \cdot D] \right\} & \text{for } i = \left\lceil \log_{1+\epsilon} \dfrac{|\mathcal{C}|}{\epsilon} \right\rceil, \\ \left\{ f \in \mathcal{F}[s] \mid d(s,f) \in [(1+\epsilon)^{-(i+1)} \cdot D, (1+\epsilon)^{-i} \cdot D] \right\} & \text{for } i < \left\lceil \log_{1+\epsilon} \dfrac{|\mathcal{C}|}{\epsilon} \right\rceil. \end{cases}$$

The clients further than $D$ from their centre may be disregarded, since they will not appear in the optimum solution.

Once we know the number $k_s^i$ of facilities to open in a particular ring, we may again do it greedily—this time by opening ones with largest capacity.

**Proposition 8.** *Let $\left\langle \mathcal{F}^{OPT(S)}, \phi^{OPT(S)} \right\rangle$ be the optimum solution to* CkM *on l-centred metric $d^S$. For every ring $\mathcal{F}^i[s]$ let $k_s^i = |\mathcal{F}^i[s] \cap \mathcal{F}^{OPT(S)}|$. Let $\left\langle \mathcal{F}^{SOL}, \phi^{SOL} \right\rangle$ be determined by picking $k_s^i$ largest facilities in each ring $\mathcal{F}^i[s]$. Then the following inequality holds:*

$$d^S(\phi^{SOL}) \leqslant (1+\epsilon) \cdot d^S(\phi^{OPT}) + \epsilon \cdot D \leqslant (1+2\epsilon) \cdot d^S(\phi^{OPT}). \tag{4}$$

*Proof.* By definition,

$$\begin{aligned} d^S(\phi^{\mathrm{OPT}(S)}) &= \sum_{c \in \mathcal{C}} d^S(c, \phi^{\mathrm{OPT}(S)}(c)) \\ &= \sum_{c \in \mathcal{C}} d^S\left(c, \phi^S(\phi^{\mathrm{OPT}(S)}(c))\right) + \sum_{f \in \mathcal{F}^{\mathrm{OPT}(S)}} \left|\phi^{\mathrm{OPT}(S)^{-1}}(f)\right| \cdot d^S\left(\phi^S(f), f\right) \\ &= \sum_{c \in \mathcal{C}} d^S\left(c, \phi^S(\phi^{\mathrm{OPT}(S)}(c))\right) \\ &\qquad + \sum_{s \in S} \sum_{i=0}^{B} \sum_{f \in \mathcal{F}^{\mathrm{OPT}(S)} \cap \mathcal{F}^i[s]} \left|\phi^{\mathrm{OPT}(S)^{-1}}(f)\right| \cdot d^S\left(\phi^S(f), f\right). \end{aligned}$$

Let us analyse the additional cost incurred by replacing facilities from $\mathcal{F}^{\mathrm{OPT}(S)}$ with those from $\mathcal{F}^{\mathrm{SOL}}$. Define a bijection $\sigma \colon \mathcal{F}^{\mathrm{OPT}(S)} \to \mathcal{F}^{\mathrm{SOL}}$ which preserves every ring $F^i[s]$. We set $\phi(c) \coloneqq \sigma(\phi^{\mathrm{OPT}(S)}(c))$. The solution $\mathcal{F}^{\mathrm{SOL}}$ has picked the largest-capacity facilities in every ring, so $\phi$ does not violate any capacities. Now, we have

$$\begin{aligned} d^S(\phi) &= \sum_{c \in \mathcal{C}} d^S\left(c, \phi^S(\phi^{\mathrm{OPT}(S)}(c))\right) \\ &\qquad + \sum_{s \in S} \sum_{i=0}^{B} \sum_{f \in \mathcal{F}^{\mathrm{OPT}(S)} \cap \mathcal{F}^i[s]} \left|\phi^{\mathrm{OPT}(S)^{-1}}(f)\right| \cdot d^S\left(\sigma(\phi^S(f)), f\right). \end{aligned}$$

For every $f \in \mathcal{F}^i[s]$ with $i < B$,

$$d^S\left(\sigma(\phi^S(f)), f\right) \leqslant (1+\epsilon) \cdot d^S\left(\phi^S(f), f\right),$$

since the rings have geometrically increasing radii. The central ring, with $i = B$, has inner radius equal to 0, so that kind of proportion on distances does not hold. Its radius is however small, so we have

$$d^S\left(\sigma(\phi^S(f)), f\right) \leqslant \frac{\epsilon \cdot D}{|\mathcal{C}|},$$

for every facility $f \in \mathcal{F}^B[s]$. Even if all $|\mathcal{C}|$ clients were connected to the facilities in the central rings of the Voronoi cells, this cost would sum up to $\epsilon \cdot D$. Finally we observe, that $d^S(\phi^{\mathrm{SOL}}) \leqslant d^S(\phi)$, as $\phi^{\mathrm{SOL}}$ is the optimal mapping to $\mathcal{F}^{\mathrm{SOL}}$.

The right inequality of (4) is given by $D \leqslant d^S(\phi^{\mathrm{OPT}(S)})$. □

Proposition 8 lays out the algorithm for us. After guessing the upper-bound $D$, we scan all configurations $(k_s^i)_{s\in S\, i\in\{0,\dots,B\}}$ with $\sum_{s\in S}\sum_{i=0}^{B} k_s^i = k$, which determine the number of open facilities in every ring. For each such configuration we greedily open $k_s^i$ largest-capacity facilities in every ring $\mathcal{F}^i[s]$ and find the optimal mapping of clients to the facilities using our subroutine from Section 4.1. There are at most $\mathcal{O}\Big(l \cdot \frac{1}{\epsilon} \cdot \ln \frac{|\mathcal{C}|}{\epsilon}\Big)^k$ possible configurations and $|\mathcal{F}| \cdot |\mathcal{C}|$ choices of $D$.

**Corollary.** *Capacitated $k$-Median on $l$-centred instances can be $(1+\epsilon)$-approximated in time $\mathcal{O}\Big(l \cdot \frac{1}{\epsilon} \cdot \ln \frac{|\mathcal{C}|}{\epsilon}\Big)^k \cdot \mathbf{poly}(|\mathcal{F} \cup \mathcal{C}|)$.*

## 4.4. Choosing the right $l$

We now have the algorithms for $l$-centred instances, and only need to come up with an appropriate $l$-centred metric. One can see, that using a constant-factor approximation for $k$-Median with $l = k$ would be enough to achieve a (much larger) constant-factor for CkM. We can however optimise by taking $l > k$. Specifically, we will use a result of Lin and Vitter [LV92].

**Theorem 9** (Lin, Vitter [LV92, Theorem 2])
*For any $k$-Median instance $\langle \mathcal{F} \cup \mathcal{C}, d, k\rangle$, let $\mathcal{F}^{OPT}$ be the optimum solution to $k$-Median (without capacities). A set $U$ that satisfies:*

- $|U| \leqslant (1+\frac{1}{\varepsilon})k(\ln(|\mathcal{C} \cup \mathcal{F}|)+1)$,
- $\sum_{c\in\mathcal{C}} \min_{f\in U}\{d(c,f)\} \leqslant (1+\varepsilon)\sum_{c\in\mathcal{C}} \min_{f\in\mathcal{F}^{OPT}}\{d(c,f)\}$,

*can be found deterministically in time polynomial in $|\mathcal{C} \cup \mathcal{F}|$ and $\frac{1}{\varepsilon}$.*

By plugging Theorem 9 to construct the $l$-centred metric and running the algorithm from Section 4.3 on that metric, we obtain a $(7+\mathcal{O}(\epsilon))$-approximation for Capacitated $k$-Median with running time

$$\begin{aligned} &\mathcal{O}\left(\left(1+\frac{1}{\epsilon}\right)\cdot k \cdot (\ln(|\mathcal{C}\cup\mathcal{F}|)+1)\cdot\frac{1}{\epsilon}\cdot\ln\frac{|\mathcal{C}|}{\epsilon}\right)^k \cdot \mathbf{poly}(|\mathcal{F}\cup\mathcal{C}|) \\ &\quad = \mathcal{O}\Big(k \cdot \ln^2(|\mathcal{C}\cup\mathcal{F}|)\Big)^k \cdot \mathbf{poly}\left(\frac{1}{\epsilon^k}\right)\cdot \mathbf{poly}(|\mathcal{F}\cup\mathcal{C}|). \end{aligned} \tag{5}$$

We will use a standard trick in parametrised complexity to bound this running time.

**Proposition 10.** *For any $n$, $k$, $(\ln n)^{2k} \leqslant \max\{n, k^{\mathcal{O}(k)}\}$.*

*Proof.* We analyse two cases. First, if $\frac{\ln n}{2\ln\ln n} \leqslant k$, we have $\ln n = \mathcal{O}(k \ln k)$ and thus $(\ln n)^{2k} = k^{\mathcal{O}(k)}$. If $\frac{\ln n}{2\ln\ln n} > k$, then $(\ln n)^{2k} = \exp(2k \cdot \ln\ln n) < \exp(2\frac{\ln n}{2\ln\ln n} \cdot \ln\ln n) = n$. □

By applying Proposition 10 with $n = |\mathcal{F} \cup \mathcal{C}|$ to (5) we obtain the bound on complexity

$$\left(\frac{k}{\epsilon}\right)^{\mathcal{O}(k)} \cdot \mathbf{poly}(|\mathcal{F} \cup \mathcal{C}|),$$

thus proving Theorem 2.

# 5. Notes

The topic of approximation in parametrised time—a crossover of approximation algorithms and parametrised complexity—has gained a significant popularity in recent years. It has allowed to break some known (or conjectured) bounds on approximability by abusing the running time, like $(2-\varepsilon)$ ratio for $k$-Cut [GLL18]. At the same time, some problems have turned out to be resistant to such improvements. One such example is presented in Chapter 3 of this thesis.

The results in this chapter are based on our paper with Marek Adamczyk, Jarosław Byrka, Syed Mohammad Meesum and Michał Włodarczyk, which was presented at ESA 2019 [Ada+19]. After the preprint of our paper was announced, Xu et al. worked out a similar algorithm for Capacitated $k$-Means on euclidean metrics. Both our and their ratios have soon been improved by Cohen-Addad and Li [CL19], who obtained $(3+\varepsilon)$-approximation for Capacitated $k$-Median and $(9+\varepsilon)$-approximation for Capacitated $k$-Means. At the same conference, they also showed—together with Gupta, Kumar and Lee [Coh+19]—that an approximation ratio of $(1+\frac{2}{e}-\varepsilon)$ is impossible for $k$-Median in time parametrised by $k$, even without the capacities, if Gap-ETH holds. The same lower-bound had been known for algorithms running in polynomial time [Fei96].

In a subsequent paper, written together with Jarosław Byrka, Szymon Dudycz, Pasin Manurangsi and Michał Włodarczyk [Byr+20]—in which my involvement was lesser, and which is therefore not included in this dissertation—we analysed the approximability of $k$-Median problem parametrised by $|\mathcal{F}| - k$ rather then $k$. It turns out, that although $(1+\varepsilon)$-approximation is possible for $k$-Median without capacities, the bound of $(1+\frac{2}{e})$ still cannot be overcome for `CkM`.

# Bibliography


[Thi95] Thorvald Thiele. 'Om Flerfoldsvalg'. In: *Oversigt over det Kongelige Danske Videnskabernes Selskabs Forhandlinger (in Danish)*. 1895.

[LeC65] Lucien LeCam. 'On the Distribution of Sums of Independent Random Variables'. In: *Bernoulli 1713, Bayes 1763, Laplace 1813*. 1965. DOI: 10.1007/978-3-642-99884-3_10.

[Gup69] Ram Prakash Gupta. 'Independence and covering numbers of line graphs and total graphs'. In: *Proof Techniques in Graph Theory*. 1969.

[Kar72] Richard M. Karp. 'Reducibility Among Combinatorial Problems'. In: *Proceedings of a symposium on the Complexity of Computer Computations*. 1972. DOI: 10.1007/978-1-4684-2001-2_9.

[MH77] Sandra Lee Mitchell and Stephen Hedetniemi. 'Edge Domination in Trees'. In: *Proceedings of the Eighth Southeastern Conf. on Combinatorics, Graph Theory and Computing*. 1977.

[NWF78] George L. Nemhauser, Laurence A. Wolsey and Marshall L. Fisher. 'An analysis of approximations for maximizing submodular set functions - I'. In: *Math. Program.* (1978). DOI: 10.1007/BF01588971.

[HN79] Wen-Lian Hsu and George L. Nemhauser. 'Easy and hard bottleneck location problems'. In: *Discret. Appl. Math.* (1979). DOI: 10.1016/0166-218X(79)90044-1.

[YG80] Mihalis Yannakakis and Fanica Gavril. 'Edge Dominating Sets in Graphs'. In: *SIAM Journal on Applied Mathematics* (1980). DOI: 10.1137/0138030.

[Llo82] Stuart P. Lloyd. 'Least squares quantization in PCM'. In: *IEEE Trans. Inf. Theory* (1982). DOI: 10.1109/TIT.1982.1056489.

[Bak83] Brenda S. Baker. 'Approximation Algorithms for NP-Complete Problems on Planar Graphs (Preliminary Version)'. In: *24th Annual Symposium on Foundations of Computer Science, FOCS*. 1983. DOI: 10.1109/SFCS.1983.7.

[Gon85] Teofilo F. Gonzalez. 'Clustering to Minimize the Maximum Intercluster Distance'. In: *Theor. Comput. Sci.* (1985). DOI: 10.1016/0304-3975(85)90224-5.

[LV92] Jyh-Han Lin and Jeffrey Scott Vitter. '$\varepsilon$-Approximations with Minimum Packing Constraint Violation (Extended Abstract)'. In: *Proceedings of the 24th Annual ACM Symposium on Theory of Computing, STOC*. 1992. DOI: 10.1145/129712.129787.

[HK93] Joseph Douglas Horton and Kyriakos Kilakos. 'Minimum Edge Dominating Sets'. In: *SIAM J. Discret. Math.* (1993). DOI: 10.1137/0406030.

[Sri+95] Anand Srinivasan, K. Madhukar, P. Nagavamsi, C. Pandu Rangan and Maw-Shang Chang. 'Edge Domination on Bipartite Permutation Graphs and Cotriangulated Graphs'. In: *Inf. Process. Lett.* (1995). DOI: 10.1016/0020-0190(95)94093-8.

[Fei96] Uriel Feige. 'A Threshold of ln $n$ for Approximating Set Cover (Preliminary Version)'. In: *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, STOC*. 1996. DOI: 10.1145/237814.237977.

[Aro+98] Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan and Mario Szegedy. 'Proof Verification and the Hardness of Approximation Problems'. In: *J. ACM* (1998). DOI: 10.1145/278298.278306.

[Cha+98] Moses Charikar, Chandra Chekuri, Ashish Goel and Sudipto 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 (STOC)*. 1998. DOI: 10.1145/276698.276719.

[Cha+99] Moses Charikar, Sudipto Guha, Éva Tardos and David B. Shmoys. 'A Constant-Factor Approximation Algorithm for the $k$-Median Problem (Extended Abstract)'. In: *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing (STOC)*. 1999. DOI: 10.1145/301250.301257.

[Car+00] Robert D. Carr, Toshihiro Fujito, Goran Konjevod and Ojas Parekh. 'A $2\frac{1}{10}$-Approximation Algorithm for a Generalization of the Weighted Edge-Dominating Set Problem'. In: *Algorithms - ESA 2000, 8th Annual European Symposium*. 2000. DOI: 10.1007/3-540-45253-2_13.

[Fuj01] Toshihiro Fujito. 'On approximability of the independent/connected edge dominating set problems'. In: *Inf. Process. Lett.* (2001). DOI: 10.1016/S0020-0190(01)00138-7.

[FN02] Toshihiro Fujito and Hiroshi Nagamochi. 'A 2-approximation algorithm for the minimum weight edge dominating set problem'. In: *Discrete Applied Mathematics* (2002). DOI: 10.1016/S0166-218X(00)00383-8.

[Kan+02] Tapas Kanungo, David M. Mount, Nathan S. Netanyahu, Christine D. Piatko, Ruth Silverman and Angela Y. Wu. 'A local search approximation algorithm for k-means clustering'. In: *Proceedings of the 18th Annual Symposium on Computational Geometry*. 2002. DOI: 10.1145/513400.513402.

[Kho02] Subhash Khot. 'On the power of unique 2-prover 1-round games'. In: *Proceedings on 34th Annual ACM Symposium on Theory of Computing, STOC*. 2002. DOI: 10.1145/509907.510017.

[Par02] Ojas Parekh. 'Edge dominating and hypomatchable sets'. In: *Proceedings of the Thirteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*. 2002.

[FRT03] Jittat Fakcharoenphol, Satish Rao and Kunal Talwar. 'A tight bound on approximating arbitrary metrics by tree metrics'. In: *Proceedings of the 35th Annual ACM Symposium on Theory of Computing, STOC*. 2003. DOI: 10.1145/780542.780608.

[KR03] Subhash Khot and Oded Regev. 'Vertex Cover Might be Hard to Approximate to within $2-\varepsilon$'. In: *18th Annual IEEE Conference on Computational Complexity CCC*. 2003. DOI: 10.1109/CCC.2003.1214437.

[AS04] Alexander A. Ageev and Maxim Sviridenko. 'Pipage Rounding: A New Method of Constructing Algorithms with Proven Performance Guarantee'. In: *J. Comb. Optim.* (2004). DOI: 10.1023/B:JOCO.0000038913.96607.c2.

[Ary+04] Vijay Arya, Naveen Garg, Rohit Khandekar, Adam Meyerson, Kamesh Munagala and Vinayaka Pandit. 'Local Search Heuristics for k-Median and Facility Location Problems'. In: *SIAM J. Comput.* (2004). DOI: 10.1137/S0097539702416402.

[CR05] Julia Chuzhoy and Yuval Rabani. 'Approximating k-median with non-uniform capacities'. In: *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*. 2005.

[MOO05] Elchanan Mossel, Ryan O'Donnell and Krzysztof Oleszkiewicz. 'Noise stability of functions with low in.uences invariance and optimality'. In: *46th Annual IEEE Symposium on Foundations of Computer Science, FOCS*. 2005. DOI: 10.1109/SFCS.2005.53.

[CC06] Miroslav Chlebík and Janka Chlebíková. 'Approximation hardness of edge dominating set problems'. In: *J. Comb. Optim.* (2006). DOI: 10.1007/s10878-006-7908-0.

[AV07] David Arthur and Sergei Vassilvitskii. 'k-means++: the advantages of careful seeding'. In: *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*. 2007.

[Căl+07] Gruia Călinescu, Chandra Chekuri, Martin Pál and Jan Vondrák. 'Maximizing a Submodular Set Function Subject to a Matroid Constraint (Extended Abstract)'. In: *Integer Programming and Combinatorial Optimization, 12th International IPCO Conference*. 2007. DOI: 10.1007/978-3-540-72792-7_15.

[SS07] M. Shaked and J.G. Shanthikumar. *Stochastic Orders*. 2007. ISBN: 978-0-387-34675-5. DOI: 10.1007/978-0-387-34675-5.

[GLR08] Zvi Gotthilf, Moshe Lewenstein and Elad Rainshmidt. 'A $2-c\frac{\log n}{n}$ Approximation Algorithm for the Minimum Maximal Matching Problem'. In: *Approximation and Online Algorithms, 6th International Workshop, WAOA*. 2008. DOI: 10.1007/978-3-540-93980-1_21.

[GMR08] Venkatesan Guruswami, Rajsekar Manokaran and Prasad Raghavendra. 'Beating the Random Ordering is Hard: Inapproximability of Maximum Acyclic Subgraph'. In: *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS*. 2008. DOI: 10.1109/FOCS.2008.51.

[BK09] Nikhil Bansal and Subhash Khot. 'Optimal Long Code Test with One Free Bit'. In: *50th Annual IEEE Symposium on Foundations of Computer Science, FOCS*. 2009. DOI: 10.1109/FOCS.2009.23.

[RS10] Prasad Raghavendra and David Steurer. 'Graph expansion and the unique games conjecture'. In: *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC*. 2010. DOI: 10.1145/1806689.1806792.

[Ste10] David Steurer. 'On the Complexity of Unique Games and Graph Expansion'. PhD thesis. Princeton University, 2010.

[Căl+11] Gruia Călinescu, Chandra Chekuri, Martin Pál and Jan Vondrák. 'Maximizing a Monotone Submodular Function Subject to a Matroid Constraint'. In: *SIAM J. Comput.* (2011). DOI: 10.1137/080733991.

[MKI11] Yusuke Matsumoto, Naoyuki Kamiyama and Keiko Imai. 'An approximation algorithm dependent on edge-coloring number for minimum maximal matching problem'. In: *Inf. Process. Lett.* (2011). DOI: 10.1016/j.ipl.2011.02.006.

[RST12] Prasad Raghavendra, David Steurer and Madhur Tulsiani. 'Reductions between Expansion Problems'. In: *Proceedings of the 27th Conference on Computational Complexity, CCC*. 2012. DOI: 10.1109/CCC.2012.43.

[SV12] Richard Schmied and Claus Viehmann. 'Approximating edge dominating set in dense graphs'. In: *Theor. Comput. Sci.* (2012). DOI: 10.1016/j.tcs.2011.10.001.

[LS13] Shi Li and Ola Svensson. 'Approximating k-median via pseudo-approximation'. In: *Symposium on Theory of Computing Conference, STOC*. 2013. DOI: 10.1145/2488608.2488723.

[Byr+15] Jaroslaw Byrka, Thomas W. Pensyl, Bartosz Rybicki, Aravind Srinivasan and Khoa Trinh. 'An Improved Approximation for $k$-median, and Positive Correlation in Budgeted Optimization'. In: *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*. 2015. DOI: 10.1137/1.9781611973730.50.

[Cyg+15] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk and Saket Saurabh. *Parameterized Algorithms*. 2015. ISBN: 978-3-319-21274-6. DOI: 10.1007/978-3-319-21275-3.

[Esc+15] Bruno Escoffier, Jérôme Monnot, Vangelis Th. Paschos and Mingyu Xiao. 'New Results on Polynomial Inapproximabilityand Fixed Parameter Approximability of Edge Dominating Set'. In: *Theory Comput. Syst.* (2015). DOI: 10.1007/s00224-014-9549-5.

[Li15] Shi Li. 'On Uniform Capacitated $k$-Median Beyond the Natural LP Relaxation'. In: *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*. 2015. DOI: 10.1137/1.9781611973730.47.

[Bha+16] Amey Bhangale, Rajiv Gandhi, Mohammad Taghi Hajiaghayi, Rohit Khandekar and Guy Kortsarz. 'Bicovering: Covering Edges With Two Small Subsets of Vertices'. In: *43rd International Colloquium on Automata, Languages, and Programming, ICALP*. 2016. DOI: 10.4230/LIPIcs.ICALP.2016.6.

[BRU16] Jaroslaw Byrka, Bartosz Rybicki and Sumedha Uniyal. 'An Approximation Algorithm for Uniform Capacitated k-Median Problem with $1+\varepsilon$ Capacity Violation'. In: *Integer Programming and Combinatorial Optimization - 18th International Conference, IPCO*. 2016. DOI: 10.1007/978-3-319-33461-5_22.

[DL16] H. Gökalp Demirci and Shi Li. 'Constant Approximation for Capacitated k-Median with $(1+\varepsilon)$-Capacity Violation'. In: *43rd International Colloquium on Automata, Languages, and Programming, ICALP*. 2016. DOI: 10.4230/LIPIcs.ICALP.2016.73.

[Din16] Irit Dinur. 'Mildly exponential reduction from gap 3SAT to polynomial-gap label-cover'. In: *Electron. Colloquium Comput. Complex.* (2016).

[Li16] Shi Li. 'Approximating capacitated $k$-median with $(1+\varepsilon)k$ open facilities'. In: *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*. 2016. DOI: 10.1137/1.9781611974331.ch56.

[SFL16] Piotr Skowron, Piotr Faliszewski and Jérôme Lang. 'Finding a Collective Set of Items: From Proportional Multirepresentation to Group Recommendation'. In: *Artif. Intell.* (2016). DOI: 10.1016/j.artint.2016.09.003.

[Ahm+17] Sara Ahmadian, Ashkan Norouzi-Fard, Ola Svensson and Justin Ward. 'Better Guarantees for k-Means and Euclidean k-Median by Primal-Dual Algorithms'. In: *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS*. 2017. DOI: 10.1109/FOCS.2017.15.

[Cha+17] Parinya Chalermsook, Marek Cygan, Guy Kortsarz, Bundit Laekhanukit, Pasin Manurangsi, Danupon Nanongkai and Luca Trevisan. 'From Gap-ETH to FPT-Inapproximability: Clique, Dominating Set, and More'. In: *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS*. 2017. DOI: 10.1109/FOCS.2017.74.

[Man17] Pasin Manurangsi. 'Inapproximability of Maximum Edge Biclique, Maximum Balanced Biclique and Minimum k-Cut from the Small Set Expansion Hypothesis'. In: *44th International Colloquium on Automata, Languages, and Programming, ICALP*. 2017. DOI: 10.4230/LIPIcs.ICALP.2017.79.

[MR17] Pasin Manurangsi and Prasad Raghavendra. 'A Birthday Repetition Theorem and Complexity of Approximating Dense CSPs'. In: *44th International Colloquium on Automata, Languages, and Programming, ICALP*. 2017. DOI: 10.4230/LIPIcs.ICALP.2017.78.

[GLL18] Anupam Gupta, Euiwoong Lee and Jason Li. 'An FPT Algorithm Beating 2-Approximation for *k*-Cut'. In: *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*. 2018. DOI: 10.1137/1.9781611975031.179.

[Ada+19] Marek Adamczyk, Jaroslaw Byrka, Jan Marcinkowski, Syed Mohammad Meesum and Michal Wlodarczyk. 'Constant-Factor FPT Approximation for Capacitated k-Median'. In: *27th Annual European Symposium on Algorithms, ESA*. 2019. DOI: 10.4230/LIPIcs.ESA.2019.1.

[Coh+19] Vincent Cohen-Addad, Anupam Gupta, Amit Kumar, Euiwoong Lee and Jason Li. 'Tight FPT Approximations for k-Median and k-Means'. In: *46th International Colloquium on Automata, Languages, and Programming, ICALP*. 2019. DOI: 10.4230/LIPIcs.ICALP.2019.42.

[CL19] Vincent Cohen-Addad and Jason Li. 'On the Fixed-Parameter Tractability of Capacitated Clustering'. In: *46th International Colloquium on Automata, Languages, and Programming, ICALP*. 2019. DOI: 10.4230/LIPIcs.ICALP.2019.41.

[DLM19] Szymon Dudycz, Mateusz Lewandowski and Jan Marcinkowski. 'Tight Approximation Ratio for Minimum Maximal Matching'. In: *Integer Programming and Combinatorial Optimization - 20th International Conference, IPCO*. 2019.

[Bar+20] Siddharth Barman, Omar Fawzi, Suprovat Ghoshal and Emirhan Gürpinar. 'Tight Approximation Bounds for Maximum Multi-coverage'. In: *Integer Programming and Combinatorial Optimization - 21st International Conference, IPCO*. 2020. DOI: 10.1007/978-3-030-45771-6_6.

[Bat20] Wouter Cames van Batenburg. 'Minimum maximal matchings in cubic graphs'. In: *CoRR* (2020). arXiv: 2008.01863.

[Byr+20] Jaroslaw Byrka, Szymon Dudycz, Pasin Manurangsi, Jan Marcinkowski and Michal Wlodarczyk. 'To Close Is Easier Than To Open: Dual Parameterization To k-Median'. In: *Approximation and Online Algorithms - 18th International Workshop, WAOA*. 2020. arXiv: 2011.08083.

[DMM20] Szymon Dudycz, Pasin Manurangsi and Jan Marcinkowski. 'Tight Inapproximability of Minimum Maximal Matching on Bipartite Graphs and Related Problems'. to appear. 2020.

[Dud+20] Szymon Dudycz, Pasin Manurangsi, Jan Marcinkowski and Krzysztof Sornat. 'Tight Approximation for Proportional Approval Voting'. In: *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI*. 2020. DOI: 10.24963/ijcai.2020/39.

[Man20] Pasin Manurangsi. 'Tight Running Time Lower Bounds for Strong Inapproximability of Maximum *k*-Coverage, Unique Set Cover and Related Problems (via *t*-Wise Agreement Testing Theorem)'. In: *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA*. 2020. DOI: 10.1137/1.9781611975994.5.

[BFF21] Siddharth Barman, Omar Fawzi and Paul Fermé. 'Tight Approximation Guarantees for Concave Coverage Problems'. In: *38th International Symposium on Theoretical Aspects of Computer Science, STACS*. 2021. DOI: 10.4230/LIPIcs.STACS.2021.9.