Dissertation

Lower-Bounded Clustering -

Models, Complexity and (Parameterised) Approximation

Katrin Casel

January 2018

Dissertation

Lower-Bounded Clustering -
Models, Complexity and (Parameterised) Approximation

vorgelegt von: Katrin Casel

zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften (Dr. rer. nat.)
am Fachbereich IV der Universität Trier

Erster Gutachter:	Prof. Dr. Henning Fernau, Universität Trier
Zweiter Gutachter:	Prof. Dr. Ljiljana Brankovic, The University of Newcastle (Australia)

Januar 2018

Abstract

This thesis considers the general task of computing a partition of a set of given objects such that each set of the partition has a cardinality of at least a fixed number k. Among such kinds of partitions, which we call *k-clusters*, the objective is to find the k-cluster which minimises a certain cost derived from a given pairwise difference between objects which end up the same set. As a first step, this thesis introduces a general problem, denoted by $(\|\cdot\|, f)$-k-CLUSTER, which models the task to find a k-cluster of minimum cost given by an objective function computed with respect to specific choices for the cost functions f and $\|\cdot\|$. In particular this thesis considers three different choices for f and also three different choices for $\|\cdot\|$ which results in a total of nine different variants of the general problem $(\|\cdot\|, f)$-k-CLUSTER.

Especially with the idea to use the concept of parameterised approximation, we first investigate the role of the lower bound on the cluster cardinalities and find that k is not a suitable parameter, due to remaining NP-hardness even for the restriction to the constant 3. The reductions presented to show this hardness yield the even stronger result which states that polynomial time approximations with some constant performance ratio for any of the nine variants of $(\|\cdot\|, f)$-k-CLUSTER require a restriction to instances for which the pairwise distance on the objects satisfies the triangle inequality.

For this restriction to what we informally refer to as *metric* instances, constant-factor approximation algorithms for eight of the nine variants of $(\|\cdot\|, f)$-k-CLUSTER are presented. While two of these algorithms yield the provably best approximation ratio (assuming $\mathsf{P} \neq \mathsf{NP}$), others can only guarantee a performance which depends on the lower bound k.

With the positive effect of the triangle inequality and applications to facility location in mind, we discuss the further restriction to the setting where the given objects are points in the Euclidean metric space. Considering the effect of computational hardness caused by high dimensionality of the input for other related problems (*curse of dimensionality*) we check if this is also the source of intractability for $(\|\cdot\|, f)$-k-CLUSTER. Remaining NP-hardness for restriction to small constant dimensionality however disproves this theory.

We then use parameterisation to develop approximation algorithms for $(\|\cdot\|, f)$-k-CLUSTER without restriction to metric instances. In particular, we discuss structural parameters which reflect how much the given input differs from a metric. This idea results in parameterised approximation algorithms with parameters such as the number of *conflicts* (our name for pairs of objects for which the triangle inequality is violated) or the number of *conflict vertices* (objects involved in a conflict). The performance ratios of these parameterised approximations are in most cases identical to those of the approximations for metric instances. This shows that for most variants of $(\|\cdot\|, f)$-k-CLUSTER efficient and reasonable solutions are also possible for non-metric instances.

Zusammenfassung Deutsch

Die Arbeit beschäftigt sich mit dem abstrakten Clustering-Problem, für eine gegebene Menge von Objekten eine nach gewissen Qualitätsmaßen gemessene beste Partition zu bestimmen, sodass jede Teilmenge dieser eine gegebene feste Mindestkardinalität k besitzt. Als Qualitätsmaß werden insgesamt neun verschiedene konkrete Maßfunktionen diskutiert, die alle mit einer gegebenen paarweisen Distanz d auf den Objekten arbeiten. Für die neun Probleme, die sich daraus ergeben, werden Lösungsverfahren diskutiert, die hauptsächlich Methoden aus der parametrisierten und approximativen Algorithmik nutzen.

Konkret wird zunächst die Komplexität dieser Probleme in Bezug auf die Mindestkardinalität k als Parameter diskutiert. Es wird gezeigt, dass alle neun Problemvarianten bereits für die Einschränkung auf $k = 3$ NP-schwer sind, was nicht nur exakte polynomielle Lösbarkeit sondern auch effiziente parametrisierte Algorithmen für diese Parameterwahl sehr unrealistisch macht. Die Reduktionen, die für diese Komplexitätsschranken erstellt werden, zeigen außerdem, dass Approximierbarkeit nur dann möglich ist, wenn die gegebene Distanzfunktion d die Dreiecksungleichung erfüllt.

Mit Einschränkung auf Dreiecksungleichung werden für acht der neun Problemvarianten polynomielle Approximationsalgorithmen mit beweisbarer Güte vorgestellt. Zwei dieser Algorithmen garantieren eine bestmögliche Approximationsgüte (unter der Annahme $\mathsf{P} \neq \mathsf{NP}$), für die restlichen sechs lässt sich dagegen nur eine Güte beweisen, die von k abhängt.

Des weiteren wird diskutiert, ob eine Einschränkung auf Instanzen im Euklidischen Raum zu leichterer Lösbarkeit führen kann. Insbesondere im Hinblick auf den sog. *curse of dimensionality*, wird untersucht, ob sich Vektoren in niedrig dimensionalen Räumen effizient partitionieren lassen. Es stellt sich heraus, dass NP-Schwere für die meisten der neun Problemvarianten auch für Punkte im zwei- oder drei-dimensionalen Raum bestehen bleibt, sogar in Kombination mit einer Einschränkung auf konstante Werte für k.

Um Probleminstanzen zu betrachten, für die d die Dreiecksungleichung verletzt, muss, auch für approximative Lösungen mit beweisbarer Güte, mehr als polynomielle Laufzeit investiert werden. Mit einer Parametrisierung nach der Anzahl von *Konflikten* (Objektpaare, die die Dreiecksungleichung verletzen), lassen sich die zuvor für eingeschränkte Instanzen entwickelten polynomiellen Verfahren verallgemeinern. Konkret liefert dies Algorithmen, die beweisbare Approximationsgüten besitzen und deren Laufzeit polynomiell in der Eingabegröße und lediglich exponentiell in der Anzahl von Konflikten ist, sog. *fixed parameter tractability* für die Konfliktanzahl als Parameter.

Als weitere Möglichkeit mit Verletzung der Dreiecksungleichung umzugehen, wird eine Relaxierung dieser um einen festen Faktor α diskutiert. Auch für diese Sichtweise lassen sich die zuvor entwickelten Verfahren verallgemeinern. Dies führt zu rein polynomiellen Approximationsalgorithmen, deren Güte sich proportional zu α verschlechtert.

Contents

1 Introduction

Clustering problems arise in different areas in very diverse forms with the only common objective of finding a partition of a given set of objects into, by some measure, similar parts. A summary which gives an overview of the clustering concepts developed over the years can be found in [44]. Most models consider variants of the classical k-Means or k-Median problem in the sense that k is a fixed given integer which determines the number of clusters one searches for. In some applications however it is not necessary to compute a partition with exactly k parts, sometimes it is not even clear how to reasonably choose a number for k. We want to discuss a clustering model which does not fix the number of clusters but instead requires that each cluster contains at least k objects. This constraint can be seen as searching for a clustering into parts of a specified minimum significance. For general classification or compression tasks, one might consider small clusters as disposable outliers.

One concrete scenario for this type of partitioning is Load Balanced Facility Location [40], a variant of the facility location problem where one is only interested in building facilities which are profitable. In this scenario, a facility is not measured by the initial cost of building it but by its profitability once it is opened. Consequently, it is only reasonable to build a facility if there are enough (but maybe not too many) customers who use it but aside from this constraint we can build as many facilities as we want.

The considered cardinality constraint also models the basic principle of "hiding in a crowd" introduced by the concept of *k-anonymity* [56]. Anonymity for an individual record x representing a person (including or linking to potentially sensitive information) in this sense, is provided by the existence of at least $k-1$ other records which are indistinguishable from x. First clustering given records into sets each of minimum cardinality k followed by some distortion step which makes records in the same set indistinguishable is one possibility for algorithmic anonymisation. Depending on the type of distortion, this concept introduces formal problems such as r-Gather [4], k-Member Clustering [16] and Microaggregation [27].

In the context of community detection in (social) networks, clustering models are usually also applied without fixing the number of clusters. The objective there is to determine sets which are highly connected, see for example the abstract model of dense graph partition, as defined in [23] for such tasks. As clusters of a small cardinality do not offer the possibility of high connectivity, the objective of community detection appears closely related to our request for a minimum cardinality.

Collaborative filtering for recommender systems is also often based on clustering; the *Recommender Systems Handbook* [55] features a whole chapter on k-Means and related techniques. The purpose of a recommender system is to predict the interest of a set of given users for a set of given items based on ratings these users have given for a subset of the items in the past. Clustering techniques are used in this regard, to partition either the set of users into groups with similar interests, or the set of items into groups with similar properties. Especially for the approach which partitions the set of users, it seems that a clustering with lower bound on the cardinality of each set is also a reasonable model for this task.

This thesis considers the general task of computing a clustering of given objects into sets of minimum cardinality $k \in \mathbb{N}$, while minimising a certain cost derived from the given pairwise difference between objects which end up in the same set. We begin by introducing an abstract framework to model such types of problems. For this purpose, we define the generic problem $(\|\cdot\|, f)$-k-CLUSTER and specifically discuss nine variants of it, characterised via three different choices for the local cost function f and the global cost $\|\cdot\|$. Before we start with the formal definition of this family of problems, we introduce the formal notation used throughout the thesis.

1.1 Notation

Although the notation in this thesis is mostly standard, this section lists the commonly used definitions for clarity. Some further notations are only used in a specific section and therefore are introduced where they are needed.

When estimating running times of algorithms we use the $\mathcal{O}$-notation to suppress constants. For non-polynomial algorithms, we further use the $\mathcal{O}^*$-notation to also suppress polynomial factors. For integer n, B_n denotes the nth *Bell* number, which can be bounded by $B_n < \left(\frac{0.792n}{\log(n+1)}\right)^n$ [12]. The function name *log* is used to denote the logarithm with base 2.

1.1.1 Graph Theory Terminology

We usually use $G = (V, E)$ to denote an undirected graph given by a set of vertices V and a set of edges $E \subseteq V \times V$. For $u, v \in V$ we denote the edge connecting u to v by $\{u, v\}$; u, v and $\{u, v\}$ are also called *adjacent*. Our graphs are always loopless, so $\{u, u\} \notin E$ for all $u \in V$. The *degree* of a vertex $v \in V$ is the number of edges (and hence also vertices) adjacent to v, formally $|\{u \in V \colon \{u, v\} \in E\}|$.

For a clear distinction we always use the term *network* to refer to directed graphs and also use the term *arc* instead of directed edge and denote those by rounded parenthesis, i.e., an arc from u to v is denoted by (u, v).

For a given graph $G = (V, E)$ and any set $V' \subseteq V$, we use $G[V']$ to denote the graph *induced by* V', formally defined by the graph over vertex set V' and edge set $\{\{u, v\} \in E \colon u, v \in V'\}$. We call a set $V' \subseteq V$ an *independent set* in G, if $G[V']$ contains no edges. A set $V' \subseteq V$ is called a *vertex cover* for G, if $V \setminus V'$ is an independent set.

A *path (of length s)* in a graph $G = (V, E)$, is a sequence of pairwise distinct vertices $v_1, \ldots, v_{s+1} \in V$ such that $\{v_i, v_{i+1}\} \in E$ for all $i \in \{1, \ldots, s\}$. We call a graph *connected*, if any two vertices u and w in it can be *connected*, i.e., it exists a path $v_1, \ldots, v_{s+1}$ with $v_1 = u$ and $v_{s+1} = w$.

A graph $G = (V, E)$ is called a *forest*, if for all $u, w \in V$ which are connected in G, the path connecting u to w is unique. If G is a forest and also connected, G is also called a *tree*. A tree which only contains at most one vertex of degree more than 1 is called a *star*.

For a vertex set V, the *complete* graph on V which contains all edges $\{u, v\}$, $u, v \in V$ with $u \neq v$. As we fixed all our graphs to be loopless, we will sometimes use the Cartesian product $V \times V$ to denote the set of edges of the complete graph on V. A vertex set $V' \subseteq V$ for which $G[V']$ is the complete graph on V' is called a *clique*.

For more detailed information on graph theory, we refer to standard textbooks like [14, 25].

1.1.2 Approximation Terminology

An *optimisation problem* P is defined by a quadruple (I, S, m, goal) with I being the set of instances of P, S being a function which maps instances $x \in I$ to the set of feasible solutions for x, m being the objective function, mapping pairs (x, y) such that $x \in I$ and $y \in S(x)$ to a positive rational number and $\mathit{goal} \in \{\min, \max\}$. For every $x \in I$, we denote by $m^*(x)$ the *optimum value for P on x*, formally $m^*(x) := \mathit{goal}\{m(x, y) \colon y \in S(x)\}$. The class NPO contains all optimisation problems $P = (I, S, m, \mathit{goal})$ for which I is recognisable in polynomial time, there exists a polynomial q such that $\mathit{size}(y) \leq q(\mathit{size}(x))$ for each $x \in I$ and $y \in S(x)$ and such that for all y' with $\mathit{size}(y') \leq q(\mathit{size}(x))$ it is decidable in polynomial time whether $y' \in S(x)$, and m is computable in polynomial time.

An algorithm $\mathcal{A}$ is called an *r-approximation algorithm* for an optimisation problem P for some $r > 1$ if for every $x \in I$ with $S(x) \neq \emptyset$, $\mathcal{A}$ computes in time polynomial in the size of x a solution $y \in S(x)$ such that $r \geq \max\{\frac{m^*(x)}{m(x,y)}, \frac{m(x,y)}{m^*(x)}\}$. The class APX contains all problems from NPO for which there exists an r-approximation algorithm for some $r > 1$.

The probably most obvious way to connect classical complexity results to approximability are so-called *gap-reductions.* For a decision problem D and an minimisation problem $P = (I, S, m, \min)$, a pair of polynomially computable functions (f, c) is a gap-reduction with gap α if f maps instances of D to instances of P and c maps instances of D to a natural number such that for all instances x of D:

- $m^*(f(x)) \leq c(x)$ if x is a "yes"-instance of D, and
- $m^*(f(x)) \geq c(x)(1 + \alpha)$ if x is a "no"-instance of D.

It is not hard to see that if D is NP-hard, there exists no approximation algorithm with performance ratio $\alpha - \varepsilon$ for any $\varepsilon > 0$ for D, unless $\mathsf{P} = \mathsf{NP}$.

For two problems $P_1, P_2 \in \mathsf{NPO}$ with $P_j := (I_j, S_j, m_j, opt_j)$, $j \in \{1, 2\}$, an *L-reduction* from P_1 to P_2 is a quadruple (f, g, β, γ) such that

- f is a function from I_1 to I_2 which is computable in polynomial time and satisfies $S_2(f(x)) \neq \emptyset$ for all $x \in I_1$ such that $S_1(x) \neq \emptyset$.
- g is a function mapping for each $x \in I_1$, any pair (x, y) with $y \in S_2(f(x))$ to a solution in $S_1(x)$ in polynomial time.
- β is a constant such that $m_2^*(f(x)) \leq \beta \cdot m_1^*(x)$ for each $x \in I_1$.
- γ is a constant such that for each $x \in I_1$ and $y \in S_2(f(x))$ the following inequality holds: $|m_1^*(x) - m_1(x, g(x, y))| \leq \gamma \cdot |m_2^*(f(x)) - m_2(f(x), y)|$.

For minimisation problems ($goal$ = min), L-reduction preserves membership in APX. Since they further imply PTAS-reductions, L-reductions can be used to show hardness for APX. Since $\mathsf{APX} \neq \mathsf{PTAS}$, unless $\mathsf{P} = \mathsf{NP}$, APX-hardness of a problem is often interpreted as a strong indication that there exists no polynomial time approximation scheme. For more detailed information about approximation algorithms see [9].

1.1.3 Parameterised Complexity Terminology

A *parameterised problem* is a decision problem P with instances (x, k), where x is the actual input and $k \in \mathbb{N}$ is the *parameter*. Such a parameterised problem is called *fixed parameter tractable* if it can be solved with an algorithm which requires a running time in $\mathcal{O}(g(k) \cdot f(n))$, for a computable function g and polynomial f; we will use the term *fpt-time* to express running times of this type. The class of fixed parameter tractable parameterised problems is denoted by FPT.

Above the class FPT, parameterised problems are characterised in the W-hierarchy and above this, XP denotes the class of parameterised problems that are solvable in time $\mathcal{O}(n^{f(k)})$ (where n is the size of the instance); we will informally use *xp-time* to describe running times of this type. These complexity classes relate in the following way:

$$\mathsf{FPT} \subseteq \mathsf{W}[1] \subseteq \mathsf{W}[2] \cdots \subseteq \mathsf{W}[P] \subseteq \mathsf{XP}$$

The inclusions above are believed to be strict, most notably in this regard, the exponential time hypothesis implies $\mathsf{FPT} \neq \mathsf{W}[1]$ by [18]. Completeness for these complexity classes is defined with respect to fpt reductions. A (classical) many-one reduction R from a parameterised problem to another is an *fpt reduction*, if the parameter of the target problem is bounded in terms of the parameter of the source problem, i. e., there is a recursive function $h\colon \mathbb{N} \to \mathbb{N}$ such that $R(x,k) = (x',k')$ implies $k' \leq h(k)$.

If a parameterised problem is NP-hard for the parameter fixed to a constant, then it is not in FPT, unless $\mathsf{NP} = \mathsf{P}$. In such a case, it follows that the parameterised problem is hard for the complexity class called $\mathsf{para}\text{-}\mathsf{NP}$, which is defined as the class which contains all parameterised problems that can be solved by a non-deterministic algorithm with a running time in $\mathcal{O}(g(k) \cdot f(n))$, for a computable function g and polynomial f. Although XP and $\mathsf{para}\text{-}\mathsf{NP}$ are not comparable with respect to inclusion (which is why we did not include the class $\mathsf{para}\text{-}\mathsf{NP}$ in the inclusion chain above), it is not hard to see that problems which are $\mathsf{para}\text{-}\mathsf{NP}$-hard are not in XP, unless $\mathsf{P} = \mathsf{NP}$.

For more details about parameterised complexity see [22, 30, 38].

1.1.4 Parameterised Approximation Terminology

In most definitions (see for example [17, 20]), parameterised approximation is defined for, in a sense, very specific decision versions of optimisation problems. There the parameterised version of an optimisation problem given by (I, S, m, goal) is the decision problem P containing instances (x,k), where $x \in I$ and $k \in \mathbb{N}$ is the parameter which is interpreted as a bound on the optimum value, i.e., the answer to instance (x,k) is "yes" if and only if $m^*(x) \leq k$ for $\mathit{goal} = \min$ ($m^*(x) \geq k$ for $\mathit{goal} = \max$, resp.).

A *parameterised approximation algorithm with ratio* r for a parameterised approximation problem P is an algorithm which is guaranteed to compute on each input (x,k) which is a "yes"-instance, a solution $y \in S$ such that

$$m(x,y) \begin{cases} \leq r \cdot k & \text{if } \mathit{goal} = \min \\ \geq \frac{1}{r} \cdot k & \text{if } \mathit{goal} = \max \end{cases}$$

with a running time in $\mathcal{O}(g(k) \cdot f(n))$, for a computable function g and polynomial f. For an input (x,k) which is a "no"-instance, the behaviour of the parameterised approximation algorithm is not fixed; usually one just asks that if no solution is computed, the algorithm returns some sort of reject notice. This rejection of "no"-instances seems a little inconvenient considering that sometimes the algorithm will not give a solution for an instance (x,k) even if $S(x)$ is not empty. For minimisation problems this is only a technical issue as one can equivalently consider the parameter k to be implicitly given by the optimum value, see [20].

In this thesis however, we will only consider structural parameterisation which does not fit into this definition; Section 2.3.2 will very clearly discuss why the optimum value is not a reasonable choice of parameter. The definition of *fpt-approximation algorithm with parameter* κ discussed in [49], asks for an approximation algorithm, not for a decision but an optimisation problem, which runs in fpt-time, i.e., with running time in $\mathcal{O}(g(\kappa) \cdot f(n))$, for a computable function g and polynomial f. We prefer this view, as it captures any kind of parameterisation. Still, to avoid confusion with the term parameterised approximation algorithm as defined for standard parameterisation, we will always state our results without this notion but instead talk about asymptotic worst-case running times of approximation algorithms measured with respect to some parameter. Especially for negative results, there does not seem to exist a unified notion of parameterised approximation hardness, so for these kinds of results, we will not use hardness notions but link the existence of certain parameterised approximations to the equivalence of complexity classes.

1.2 General Abstract Model

Our goal here is to design a model which captures the task of partitioning a set of n given objects into sets of cardinality at least k in a very general way while offering close connections to other well-known combinatorial problems. We represent the objects as vertices of an undirected graph $G = (V, E)$. A feasible solution is any partition $P_1, \ldots, P_s$ of V such that $|P_i| \geq k$ for all $i \in \{1, \ldots, s\}$. In the following we will refer to such a partition as *k-cluster*. Recall that, in contrast to the classical clustering problems like s-Means or s-Median, the number of clusters s is not necessarily part of the input.

1.2.1 Distance

Of course, one does not search for just any k-cluster but for a partition which preferably only combines objects which are in some sense “close”. This similarity can be very hard to capture and the appropriate way to measure it highly depends on the clustering task and the structure of the input. We therefore consider an arbitrary distance function $d\colon V \times V \to \mathbb{Q}_+$ which for any two objects $u, v \in V$ represents the distortion which is caused by combining u and v. This general view allows to simultaneously study many different measures for dissimilarity.

In our model, the distance d is defined via a given edge-weight function $w_E\colon E \to \mathbb{Q}_+$. For two vertices $u, v \in V$ with $u \neq v$ we define $d(u, v) := w_E(\{u, v\})$ if $\{u, v\} \in E$, and if $\{u, v\} \notin E$, the distance $d(u, v)$ is defined by the shortest path, with respect to w_E, from u to v in G. For simplicity we always extend d to a function on the whole set $V \times V$ by defining $d(v, v) = 0$ for all $v \in V$. This definition of d captures the possibility of missing information about pairwise distances, as often encountered in practical scenarios.

We will say that d satisfies the *triangle inequality* if $d(u, v) \leq d(u, w) + d(w, v)$ for all $u, v, w \in V$. Observe that our definition allows for distances d which do not satisfy this property, a simple example is the complete graph over $V = \{u, v, w\}$ with $w_E(\{u, v\}) = w_E(\{u, w\}) = 1$ and $w_E(\{v, w\}) = 3$. Violations of the triangle inequality are only possible for distances defined by an edge. Edges hence do not necessarily imply similarity but can reflect a difference greater than the shortest path between two objects and make it more unattractive to cluster them together; very different from the multiedges introduced in the hypergraph model for k-anonymous clustering from [61], where hyperedges reflect similar groups.

1.2.2 Objective Function

The overall cost of a partition $P_1, \ldots, P_s$ is always in some sense proportional to the dissimilarities within each set or *cluster* P. On an abstract level, the *global cost* induced by a partition $P_1, \ldots, P_s$ is calculated by first computing the *local cost* of each cluster and second by combining all this individual information. We discuss three different measures for the local cost caused by a cluster P:

Radius: $\mathrm{rad}(P) := \min\{\max\{d(x,y) \colon y \in P\} \colon x \in P\}$.

Diameter: $\mathrm{diam}(P) := \max\{\max\{d(x,y) \colon y \in P\} \colon x \in P\}$.

Average Distortion: $\mathrm{avg}(P) := |P|^{-1} \cdot \min\{\sum_{y \in P} d(x,y) \colon x \in P\}$.

In the following, d always denotes the distance induced *on the whole graph*; hence we consider for $u, v \in P$ with $\{u,v\} \notin E$ as distance $d(u,v)$ the shortest path from u to v in G even if this path contains vertices which are not in P. For the local cost functions average distortion or radius we will sometimes call a vertex $x \in P$ *a central vertex* for cluster P, if $\mathrm{avg}(P) = \frac{1}{|P|} \sum_{y \in P} d(x,y)$ or $\mathrm{rad}(P) = \max\{d(x,y) \colon y \in P\}$, respectively. Observe that central vertices with respect to average distortion and radius may be different; in the cluster $P = \{x, y, x_1, y_1, y_2\}$ with $w_E(\{y,x\}) = w_E(\{y,y_1\}) = w_E(\{y,y_2\}) = 1$ and $w_E(\{x,x_1\}) = 2$, the vertex x is the only central vertex with respect to radius and y is the only central vertex with respect to average distortion.

The overall cost of a k-cluster $P_1, \ldots, P_s$ is given by a combination of the local costs $f(P_1), \ldots, f(P_s)$ with $f \in \{\mathrm{rad}, \mathrm{diam}, \mathrm{avg}\}$. In order to model the most common problem versions we consider the following three possibilities:

Worst Local Cost: Maximum cost among all clusters, formally computed by $\max\{f(P_i) \colon 1 \leq i \leq s\}$, denoted by $\|\cdot\|_\infty$ and informally often referred to as ∞-*norm* or *infinity-norm.*

Worst Weighted Local Cost: Maximum cost among all clusters, weighted by their sizes computed by $\max\{|P_i| f(P_i) \colon 1 \leq i \leq s\}$, denoted by $\|\cdot\|_\infty^w$, informally often referred to as *weighted* ∞- or *infinity-norm.*

Accumulated Weighted Local Cost: The sum of the local costs of all clusters, weighted by their sizes, computed by $\sum_{i=1}^{s} |P_i| f(P_i)$, denoted by $\|\cdot\|_1^w$ and informally often referred to as *1-norm.*

(Structural properties discussed in Section 2.1 will explain why we do not consider unweighted 1-norm.)

1.2.3 Problem Family

Any choice of $f \in \{\text{rad}, \text{diam}, \text{avg}\}$ and $\|\cdot\| \in \{\|\cdot\|_1^w, \|\cdot\|_\infty^w, \|\cdot\|_\infty\}$ yields a different problem. For a fixed $k \in \mathbb{N}$, the general optimisation problem is formally given by the set I being pairs (G, k) of undirected graphs G with edge-weight function and integer k, $S(G, k)$ contains all k-clusters for G, m is the composition of $\|\cdot\|$ with f and *opt* is min. More informally, we want to think of the following class of problems:

$(\|\cdot\|, f)$-k-CLUSTER

Input: Graph $G = (V, E)$ with edge-weight function $w_E \colon E \to \mathbb{Q}_+$, $k \in \mathbb{N}$.

Output: A k-cluster $P_1, \ldots, P_s$ of V for some $s \in \mathbb{N}$, which minimises $\|(f(P_1), \ldots, f(P_s))\|$.

We will use the name $(\|\cdot\|, f)$-k-CLUSTER to also refer to the natural corresponding decision problem, i.e., given a graph G with edge-weights, an integer k and a bound $D \in \mathbb{Q}_+$, does there exist a k-cluster $P_1, \ldots, P_s$ of V for some $s \in \mathbb{N}$ such that $\|(f(P_1), \ldots, f(P_s))\| \leq D$.

Also, we denote the global cost of an optimal solution for $(\|\cdot\|, f)$-k-CLUSTER on G with distance d by $\text{opt}(G, d, \|\cdot\|, f, k)$. Sometimes we will discuss the restriction of a version of $(\|\cdot\|, f)$-k-CLUSTER to a fixed value for k. In this case we denote the problem by writing this fixed value instead of k, for example, for k fixed to 2 we write $(\|\cdot\|, f)$-2-CLUSTER.

Some of the variants of $(\|\cdot\|, f)$-k-CLUSTER are known under different names. $(\|\cdot\|_1^w, \text{diam})$-$k$-CLUSTER is equivalent to k-MEMBER CLUSTERING [16] and with d chosen as the Euclidean distance, $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER is the problem r-GATHER [4] (with $r = k$). The variant $(\|\cdot\|_1^w, \text{avg})$-$k$-CLUSTER models LOAD BALANCED FACILITY LOCATION [40] with unit demands and without facility costs. Further, again with d being the Euclidean distance, $(\|\cdot\|_1^w, \text{avg})$-$k$-CLUSTER is equivalent to MICROAGGREGATION [27].

Choosing between the cluster measures and norms allows adjustment for specific types of objects and different forms of output representation. The norm decides if the desired output has preferably uniformly structured clusters with or without uniform cardinalities (∞-norm) or builds clusters of object-specific irregular structure (1-norm). For cohesive clustering, the diameter measure is more suitable for the choice of f. Average distortion is best used when the output chooses one representative of each cluster and projects all other objects in this cluster to it; a scenario which for example occurs for facility location type problems. If the output does not project to one representative but considers clusters as circular areas, the radius measure is the most reasonable choice for f. Optimal k-clusters may differ for different choices of $\|\cdot\|$ and/or f. Still, we will see that there are also very useful similarities.

1.3 Content of this Thesis

This thesis considers the general task of computing a clustering of given objects into sets of minimum cardinality $k \in \mathbb{N}$ as formally defined by the problem family $(\|\cdot\|, f)$-k-CLUSTER in the previous section.

At first, we investigate the role of the bound k on the cardinality in Section 2. We will see that the cardinality constraint comes with properties which are different from clustering tasks which fix the number of clusters. First, we compare the nine problem variants of $(\|\cdot\|, f)$-k-CLUSTER with respect to structural differences. These considerations reveal some interesting differences for the possible choices of local and global cost. We then classify the complexity for $(\|\cdot\|, f)$-k-CLUSTER restricted to small values of k by identifying polynomial time solvable cases with connections to matching-type problems and deriving NP-hardness results for the remaining cases. These results will not just show that k is not a very helpful choice for parameterisation but also that the triangle inequality for the distance d plays a key role for efficient solvability of $(\|\cdot\|, f)$-k-CLUSTER, especially with respect to approximations.

In Section 3 we therefore first consider finding approximation techniques for the restriction to distances d which satisfy the triangle inequality. We there use a large variety of connections to other graph problems, including the positive results from Section 2, to develop approximation algorithms for this restriction of most variants of $(\|\cdot\|, f)$-k-CLUSTER[1].

The positive effect of the triangle inequality raises the natural question if the restriction to even more specific distances d can improve solvability further. The most natural and commonly discussed distance is probably the Euclidean distance and we will hence consider a restriction of $(\|\cdot\|, f)$-k-CLUSTER to Euclidean space in Section 4. As for such geometric problems, the dimension of the space is usually considered as the source of computational hardness (*curse of dimensionality*), we investigate if restriction of this dimension can yield improvements. For most variants of $(\|\cdot\|, f)$-k-CLUSTER it will however follow that NP-hardness remains even for a small constant dimension.

In Section 5 we consider distances which violate the triangle inequality in some specific way, that is, either only by a limited magnitude or only for a certain number of vertices. There we will show that many results from Section 3 can be generalised to α-relaxed triangle inequality and, with the concept of parameterised approximation, also partially translate with a parameterisation by *conflicts* (pairs of vertices which violate the triangle inequality).

In Section 6 we summarise the specific results achieved in the thesis and give further research directions and a list of open problems.

We implemented some of the (parameterised) approximation algorithms to test their behaviour in practice. Throughout the thesis, we will sometimes refer to these tests and mention the insights they brought to the project.

[1] Parts of Sections 2 and 3 were published in [1] and [2].

2 The Role of the Lower Bound k

Especially with the objective of parameterisation in mind, a first interesting question concerning our problem family $(\|\cdot\|, f)$-k-CLUSTER is the role of the minimum cardinality k. We will see that this cardinality constraint generates properties which differ greatly from those of classical clustering models. Further, it turns out that the seemingly natural restriction to distances d which satisfy the triangle inequality plays an important role for structural properties and especially for approximability.

As a first step, we will investigate if there are bounds for the maximum cardinality of a cluster in an optimal solution for the different variants of $(\|\cdot\|, f)$-k-CLUSTER. Especially the choice of local cost function will play an important role for these results. But also the restriction to instances for which the distance d satisfies the triangle inequality is relevant for these bounds. We then consider fixed values for k, 2 and 3, to be precise, to see if a restriction to these yields tractable problems. For k larger or equal to 3, we will see that all variants of $(\|\cdot\|, f)$-k-CLUSTER are NP-hard, even with restriction to instances for which the distance d satisfies the triangle inequality. For $(\|\cdot\|, f)$-2-CLUSTER we find that five of the nine variants are polynomial time solvable and a sixth one becomes polynomial time solvable when restricted to instances where d satisfies the triangle inequality.

2.1 Structural Properties of Optimal Partitions

The diverse behaviour for different choices of f and $\|\cdot\|$ is nicely displayed in the cluster cardinalities of optimal solutions. For the example $G = (V, E)$ with $V = \{c, v_1, v_2, \ldots, v_n\}$ and $E = \{\{v_i, c\}\colon 1 \leq i \leq n\}$ with $w_E(\{c, v_i\}) = 1$ for all i, we find that for radius and average distortion, the single cluster V is *the* optimal solution with $\|\cdot\|_\infty$ or $\|\cdot\|_1^w$. If $w_E(\{v_i, v_j\}) = D$ for some large value D, any k-cluster with more than one set is arbitrarily worse. For the diameter measure however we know that in general $\operatorname{diam}(S) \leq \operatorname{diam}(P)$ for all sets $S \subseteq P$, which immediately yields:

Corollary 1
From any given solution $\mathfrak{P}$ for an instance of $(\|\cdot\|, \operatorname{diam})$-$k$-CLUSTER it is possible to compute in polynomial time a solution $\mathfrak{P}'$ of the same (or smaller) global cost for which $|P| < 2k$ for all $P \in \mathfrak{P}'$, for all choices of $k \in \mathbb{N}$ and $\|\cdot\| \in \{\|\cdot\|_1^w, \|\cdot\|_\infty^w, \|\cdot\|_\infty\}$.

For radius we only have the weaker property that $\operatorname{rad}(S) \leq \operatorname{rad}(P)$ for all sets $S \subseteq P$ such that a central vertex for P with respect to radius is contained in S. Average distortion lacks such monotone behaviour entirely. Observe that a large cardinality of a cluster can somehow “smooth over” some larger distances, for example for three vertices u, v, w with $w_E(\{u, v\}) = 3$ and $w_E(\{u, w\}) = 1$, adding w to the cluster $\{u, v\}$ decreases the average distortion from $\frac{3}{2}$ to $\frac{4}{3}$. Examples like these show that, even with triangle inequality for d, we cannot in general restrict the maximum cluster cardinality for $(\|\cdot\|_\infty, \text{avg})$-$k$-CLUSTER, which is a bit undesirable, given that most applications also prefer to have some upper bound on the cardinality (not too many customers). In a realistic scenario, we encounter sets of cardinality $2k$ or larger in optimal solutions for $(\|\cdot\|_\infty, \text{avg})$-$k$-CLUSTER, if they contain an object (often called outlier) which has a large distance from all objects. Deleting such outliers before computing clusters is generally a reasonable pre-processing step, which makes large clusters in $(\|\cdot\|_\infty, \text{avg})$-$k$-CLUSTER unlikely.

In general, we would like the computation of global cost to somehow favour finer partitions in order to exploit the difference to clustering models which bound the number of sets. This is the reason why we do not consider the unweighted 1-norm, i.e., $\|(f(P_1), \ldots, f(P_s))\|_1 := \sum_{i=1}^{s} f(P_i)$. For the example $V = \{v_i^1, v_i^2 \colon 1 \leq i \leq n\}$ with $w_E(\{v_i^1, v_i^2\}) = 1$ for $i \in \{1, \ldots, n\}$ and $w_E(\{v_i^h, v_j^k\}) = n - 1$ for $i, j \in \{1, \ldots, n\}$ with $i \neq j$ and $h, k \in \{1, 2\}$, the best 2-cluster with respect to $\|\cdot\|_1$ with any choice for f is V itself, while the most reasonable 2-cluster for most applications one can think of for this graph is obviously $\{\{v_i^1, v_i^2\} \colon 1 \leq i \leq n\}$. This makes $\|\cdot\|_1$ very unattractive for our clustering purposes. Observe that triangle inequality does not improve this behaviour, since the distance d for this example satisfies it.

Triangle inequality however has the strong advantage that we can restrict (for most variants of $(\|\cdot\|, f)$-k-CLUSTER without loss of generality) the set of solutions to only contain clusters of a maximum cardinality of $2k - 1$.

Theorem 2

For any $k \in \mathbb{N}$ and any graph G with edge-weights for which the induced distance d satisfies the triangle inequality, it is possible to compute in polynomial time from any given k-cluster $\mathfrak{P}$ for G, a k-cluster $\mathfrak{P}'$ for which $|P| < 2k$ for all $P \in \mathfrak{P}'$ and such that:

- *$\mathfrak{P}'$ has the same global cost as $\mathfrak{P}$ with respect to $\|\cdot\|_\infty^w$ and* rad *or* avg.
- *$\mathfrak{P}'$ has at most twice the global cost of $\mathfrak{P}$ with respect to $\|\cdot\|_1^w$ and* rad *or* avg, *and also with respect to $\|\cdot\|_\infty$ and* rad.

Proof. Consider a k-cluster $\mathfrak{P}$ containing a cluster P of cardinality $s = tk + r$ for some $t \geq 2$ and $k > r \geq 0$ with some central vertex $c \in P$ with respect to the considered local cost $f \in \{\text{rad}, \text{avg}\}$. Construct successively for $i \in \{1, \ldots, t-1\}$ the sets V_i containing k vertices from $P_i \setminus \{c\}$, where $P_i := P \setminus (V_1 \cup \cdots \cup V_{i-1})$, including $v_i := \text{argmin}\{d(p, c) \colon p \in P_i \setminus \{c\}\}$. We consider the increase of global cost for replacing P by $V_1, \ldots, V_{t-1}, P_{t-1}$ in $\mathfrak{P}$:

For the local cost radius, we see that $\operatorname{rad}(P_i) \leq \operatorname{rad}(P)$ for all i and hence especially for $i = t-1$. The radius of the sets V_i can be bounded by:

$$\operatorname{rad}(V_i) \leq \max\{d(v_i, p) \colon p \in V_i\} \leq d(v_i, c) + \max\{d(c, p) \colon p \in V_i\} \leq 2 \cdot \operatorname{rad}(P)\,.$$

The global cost for $(\|\cdot\|_1^w, \operatorname{rad})$-$k$-CLUSTER and $(\|\cdot\|_\infty, \operatorname{rad})$-$k$-CLUSTER only increases by a factor of at most 2. For the weighted ∞-norm, these inequalities yield:

$$|V_i| \cdot \operatorname{rad}(V_i) = k \cdot \operatorname{rad}(V_i) \leq 2k \cdot \operatorname{rad}(P) \leq |P| \cdot \operatorname{rad}(P)\,.$$

The global cost for $(\|\cdot\|_\infty^w, \operatorname{rad})$-$k$-CLUSTER consequently does not increase.

For the local cost average distortion, the weighted average for each P_i with $i \in \{1, \dots, t-1\}$ is bounded by:

$$|P_i| \cdot \operatorname{avg}(P_i) \leq \sum_{p \in P_i} d(c, p) \leq |P| \cdot \operatorname{avg}(P)\,.$$

The local cost for V_i with $i \in \{1, \dots, t-1\}$ is bounded by:

$$|V_i| \cdot \operatorname{avg}(V_i) \leq \sum_{p \in V_i} d(v_i, p) \leq k \cdot d(v_i, c) + \sum_{p \in V_i} d(c, p)\,.$$

By the choice of the vertices v_i we can bound $k \cdot d(v_i, c) \leq \sum_{p \in P_i} d(c, p)$ and conclude that:

$$|V_i| \cdot \operatorname{avg}(V_i) \leq \sum_{p \in P_i} d(c, p) + \sum_{p \in V_i} d(c, p) = \sum_{p \in P_{i-1}} d(c, p) \leq |P| \cdot \operatorname{avg}(P)\,.$$

The global cost with respect to the weighted ∞-norm $\|\cdot\|_\infty^w$ consequently does not increase by replacing P by $V_1, \dots, V_{t-1}, P_{t-1}$. For $(\|\cdot\|_1^w, \operatorname{avg})$-$k$-CLUSTER the partition $V_1, \dots, V_{t-1}, P_{t-1}$ adds each distance $d(c, p)$ with $p \in P$ at most twice compared to partitioning into P, which also means that the global cost is at most doubled. □

We will look at the particular case of $k = 2$ in the next section and therefore also show:

Proposition 3

For any instance of $(\|\cdot\|_1^w, \operatorname{avg})$-2-CLUSTER *for which the induced distance* d *satisfies the triangle inequality, it is possible to compute in polynomial time from any optimal solution* $\mathfrak{P}$*, an optimal solution* $\mathfrak{P}'$ *for which* $|P| \in \{2, 3\}$ *for all* $P \in \mathfrak{P}'$*.*

Proof. For a cluster $P = \{x_1, x_2 \dots, x_r\}$ with $r > 3$, let x_r be a central vertex with respect to average distortion. A further partitioning of P into $\{x_{2i}, x_{2i+1}\}$

for $i \in \{1, \ldots, z-1\}$ with $z = \lfloor \frac{r}{2} \rfloor$ and $\{x_1, x_{2z}, x_r\}$ does not increase the global cost for $(\|\cdot\|_1^w, \text{avg})$-2-CLUSTER, since:

$$\begin{aligned}
|P| \cdot \text{avg}(P) &= \sum_{i=1}^{r} d(x_i, x_r) \\
&= d(x_{2z}, x_r) + d(x_r, x_1) + \sum_{i=1}^{z-1} d(x_{2i}, x_r) + d(x_{2i+1}, x_r) \\
&\geq |\{x_1, x_{2z}, x_r\}| \cdot \text{avg}(\{x_1, x_{2z}, x_r\}) + \sum_{i=1}^{z-1} d(x_{2i}, x_{2i+1}) \\
&= |\{x_1, x_{2z}, x_r\}| \cdot \text{avg}(\{x_1, x_{2z}, x_r\}) + \sum_{i=1}^{z-1} 2 \cdot \text{avg}(\{x_{2i}, x_{2i+1}\}) \\
&= \|\ \text{avg}(\{x_1, x_{2z}, x_r\}), \text{avg}(\{x_2, x_3\}), \ldots, \text{avg}(\{x_{2z-2}, x_{2z-1}\})\ \|_1^w
\end{aligned}$$

□

2.2 Connections to Matching Problems

The graph representation we chose to define $(\|\cdot\|, f)$-k-CLUSTER reveals relations to other well studied graph problems, in case of $k = 2$ not to classical clustering but to matching problems. A *matching* in an undirected graph $G = (V, E)$ is a subset M of E such that each vertex in V is adjacent to at most one edge in M. A matching M is *perfect* if each vertex is adjacent to exactly one edge in M.

Some variants of $(\|\cdot\|, f)$-k-CLUSTER can be reduced to the problem of finding a *minimum-weight edge cover* in a graph $G = (V, E)$ with edge-weights w_E, a subset M' of E of minimum weight (i.e., minimising $\sum_{e \in M'} w_E(e)$) for which each vertex in V is adjacent to at least one edge in M'. A minimum-weight edge cover can be reduced to the problem of finding a minimum-weight perfect matching (a simple reduction is described, e.g., in the first volume of Schrijver's monograph [[58], Section 19.3]). As a consequence, a minimum-weight edge cover can be found in $O(n^3)$ time by the results of Edmonds and Johnson [31].

Theorem 4
$(\|\cdot\|_1^w, \text{avg})$-2-CLUSTER *can be solved in* $\mathcal{O}(n^3)$ *time.*

Proof. $(\|\cdot\|_1^w, \text{avg})$-2-CLUSTER searches for a 2-cluster $P_1, \ldots, P_s$ minimising:

$$\sum_{i=1}^{s} \min \left\{ \sum_{y \in P_i} d(x, y) \colon x \in P_i \right\}.$$

In other words, for any graph $G = (V, E)$, the global cost is the weight of the cheapest edge set $E' \subseteq V \times V$ for which the graph $G' := (V, E')$ has s connected components $P_1, \dots, P_s$ with at least 2 vertices such that the induced subgraph of each P_i is a star graph. This property is equivalent to E' being a minimum-weight edge cover for the complete graph on V with edge-weights equal to the distance d; observe that the graph (V, E') is a forest without isolates and without paths of length 3 for every minimum-weight edge cover E' which means that its connected components are star graphs. □

Proposition 5

$(\|\cdot\|_\infty, \mathrm{rad})$-2-CLUSTER *can be solved in* $\mathcal{O}(n^3)$ *time.*

Proof. For a graph $G = (V, E)$, first check all vertices in V and find the smallest value $c > 0$ such that each vertex v has distance at most c from at least one other vertex. This c is obviously a general lower bound on the global cost, since each vertex needs at least one 'partner'.

For $k = 2$, this c is also the optimal value. To see this, let $\bar{E}$ be any minimum edge cover for the graph $G' := (V, E')$ with edge-set E' defined by $\{\{u, v\} \colon 0 < d(u, v) \leq c\}$. Such a cover exists, as there are no isolated vertices in G' by the choice of c. Let $C_1, \dots, C_s$ be the connected components of the graph induced by the edges in $\bar{E}$. Each such component C_i is a star graph by the minimality of the edge cover and contains at least two vertices, hence the partition $\{V[C_i] \colon 1 \leq i \leq s\}$ is a 2-cluster for G with radius at most c for each cluster. An optimal solution for $(\|\cdot\|_\infty, \mathrm{rad})$-2-CLUSTER can hence be obtained by computing a minimum edge cover for G'. □

With respect to diameter, this edge cover strategy is not applicable for clusters of cardinality larger than two. Even for $k = 2$ there are cases for which clusters of cardinality 3 are required in every optimal solution. It seems difficult to compute the diameter of a cluster by summing up certain edge-weights. We therefore consider the following problem:

SIMPLEX MATCHING

Input: Hypergraph $H = (V, F)$ with $F \subseteq (V^2 \cup V^3)$ and cost function $c \colon F \to \mathbb{Q}$ satisfying:

1. $\{\{u, v\}, \{v, w\}, \{u, w\}\} \subseteq F$ for all $\{u, v, w\} \in F$. (*subset condition*)
2. $c(\{u, v\}) + c(\{v, w\}) + c(\{u, w\}) \leq 2c(\{u, v, w\})$ for all $\{u, v, w\} \in F$. (*simplex condition*)

Output: A perfect matching of H (that is a set $S \subseteq F$ such that every vertex in V appears in exactly one hyperedge of S) of minimal cost.

This problem which can be seen as a generalisation of matching seems much more involved but it is still solvable in $\mathcal{O}(n^3m^2 \log n)$, see [7] (this kind of generalised matching can also be used for anonymisation by deletion, see [13]). It turns out that $(\|\cdot\|_1^w, \mathrm{diam})$-2-cluster can be solved with the help of the polynomial time algorithm for Simplex Matching. In particular, this idea yields the following result.

Proposition 6
$(\|\cdot\|_1^w, \mathrm{diam})$-2-cluster *can be solved in* $\mathcal{O}(n^9 \log n)$ *time.*

Proof. We model our problem as a particular instance of Simplex Matching. Let $G = (V, E)$ be an input graph for $(\|\cdot\|_1^w, \mathrm{diam})$-2-cluster. The corresponding input for Simplex Matching is the hypergraph $H = (V, V^2 \cup V^3)$ which obviously satisfies the subset condition. By Corollary 1, there exists an optimal solution for $(\|\cdot\|_1^w, \mathrm{diam})$-2-cluster among the perfect matchings for H. According to the original problem, the cost function c for any $u, v, w \in V$ is defined as:

- $c(\{u, v\}) := 2d(u, v)$ and
- $c(\{u, v, w\}) := 3 \cdot \max\{d(u, v), d(v, w), d(u, w)\}$

and hence satisfies the simplex condition. Since this complete hypergraph has $\mathcal{O}(n^3)$ hyperedges, the overall running time is in $\mathcal{O}(n^9 \log n)$. □

Diameter combined with the ∞-norms could be solved using Proposition 6 by fixing some maximum diameter D and multiplying all hyperedge costs which exceed D with a large value C, say $C = n \cdot \max\{d(u, v) \colon u, v \in V\}$. This does not violate the simplex condition for the cost function and there exists a solution for $(\|\cdot\|_\infty, \mathrm{diam})$-2-cluster of value D for the input graph if and only if the hypergraph with adjusted costs has a Simplex Matching solution of value less than C.

To improve upon the running time from Proposition 6 for the ∞-norms, we will use following problem from [66].[2]

Simplex Cover

Input: Hypergraph $H = (V, F)$ with $F \subseteq (V^2 \cup V^3)$ satisfying the subset condition, i.e., $\{\{u, v\}, \{v, w\}, \{u, w\}\} \subseteq F$ for all $\{u, v, w\} \in F$.

Output: A perfect matching of H.

[2] This covering problem is sometimes also called Unweighted Simplex Matching and is equivalent to $\{K_2, K_3\}$-packing, an old, well studied generalisation of the classical matching problem [21].

Proposition 7
$(\|\cdot\|_\infty, \text{diam})$- *and* $(\|\cdot\|_\infty^w, \text{diam})$-2-CLUSTER *and can be solved in* $\mathcal{O}(n^6 \log n)$ *time. On instances for which* d *satisfies the triangle inequality,* $(\|\cdot\|_\infty^w, \text{avg})$-2-CLUSTER *can also be solved in* $\mathcal{O}(n^6 \log n)$ *time.*

Proof. We will reduce solving each of the 2-CLUSTER problem variants to solving an instance of SIMPLEX COVER. Let $G = (V, E)$ be the input graph for the clustering problem. By Corollary 1 and Theorem 2 we can find optimal solutions for each considered problem variant among the set of perfect matchings for the hypergraph $H = (V, F)$ with $F = V^2 \cup V^3$. For a fixed value D, we build a subset $F' \subseteq F$ by removing from F all $e \in F$ depending on the problem variant by the following rule:

- Remove e if $\text{diam}(e) > D$ for $(\|\cdot\|_\infty, \text{diam})$-2-CLUSTER.
- Remove e if $|e|{\cdot}\text{diam}(e) > D$ for $(\|\cdot\|_\infty^w, \text{diam})$-2-CLUSTER.
- Remove e if $|e|{\cdot}\text{avg}(e) > D$ for $(\|\cdot\|_\infty^w, \text{avg})$-2-CLUSTER.

We claim that in all three cases, this deletion yields a subset of $V^2 \cup V^3$ that satisfies the subset condition:

- $\{u, v, w\} \in F'$ for $(\|\cdot\|_\infty, \text{diam})$-2-CLUSTER yields $\text{diam}(\{u, v, w\}) \leq D$, hence $\text{diam}(\{u, v\}), \text{diam}(\{u, w\}), \text{diam}(\{v, w\}) \leq \text{diam}(\{u, v, w\}) \leq D$, so $\{\{u, v\}, \{v, w\}, \{u, w\}\} \subseteq F'$.
- $\{u, v, w\} \in F'$ for $(\|\cdot\|_\infty^w, \text{diam})$-2-CLUSTER yields $3{\cdot}\text{diam}(\{u, v, w\}) \leq D$, hence $2{\cdot}\text{diam}(\{u, v\}) \leq D$, $2{\cdot}\text{diam}(\{u, w\}) \leq D$ and $2{\cdot}\text{diam}(\{v, w\}) \leq D$, so $\{\{u, v\}, \{v, w\}, \{u, w\}\} \subseteq F'$.
- $\{u, v, w\} \in F'$ for $(\|\cdot\|_\infty^w, \text{avg})$-2-CLUSTER yields $3{\cdot}\text{avg}(\{u, v, w\}) \leq D$. Let u be central for $\{u, v, w\}$, so $d(u, v) + d(u, w) = 3 \cdot \text{avg}(\{u, v, w\})$. It follows that $2 \cdot \text{avg}(\{u, v\}) = d(u, v) \leq D$, $2 \cdot \text{avg}(\{u, w\}) = d(u, w) \leq D$. For the edge $\{v, w\}$ we require that d satisfies the triangle inequality, in which case $2 \cdot \text{avg}(\{v, w\}) = d(v, w) \leq d(u, v) + d(u, w) \leq D$, so $\{\{u, v\}, \{v, w\}, \{u, w\}\} \subseteq F'$.

In all three cases, any subset of F' which exactly covers V, i.e., a simplex cover for $H' := (V, F')$, yields a feasible 2-cluster with global cost at most D. The augmenting path strategy from [59] solves SIMPLEX COVER in time $\mathcal{O}(m^2)$, where m is the number of hyperedges of the input graph, here at most $\mathcal{O}(n^3)$. Possible values for D are the $\mathcal{O}(n^2)$ possible different distances $d(u, v)$ for all $u, v \in V$, which, including a binary search among all possible values for D, yields an overall running time in $\mathcal{O}(n^6 \log n)$ to solve each of the 2-CLUSTER variants. □

Remark 1*:* We would like to point out that SIMPLEX MATCHING is also an interesting way to solve a sort of geometric version of $(\|\cdot\|_1^w, \text{avg})$-2-CLUSTER, originally introduced as MICROAGGREGATION in [27], which considers clustering a set of vectors in $\mathbb{R}^d$ and measures local cost for a cluster $\{x_1, \ldots, x_t\}$ by $\sum_{i=1}^t ||x_i - x||_2^2$ where x is the centroid $\frac{1}{t}(x_1 + \cdots + x_t)$ and $\|\cdot\|_2^2$ is the squared Euclidean norm. With the hypergraph $(V, V^2 \cup V^3)$ with $V = \{v_1, \ldots, v_n\}$ representing $\{x_1, \ldots, x_n\}$ and the cost function c defined by: $c(\{v_i, v_j, v_k\}) := \sum_{h \in \{i,j,k\}} ||x_h - \frac{1}{3}(x_i + x_j + x_k)||_2^2$ for all $1 \leq i < j < k \leq n$ and $c(\{v_i, v_j\}) := \frac{1}{2}||x_i - x_j||_2^2$ for all $1 \leq i < j \leq n$, the simplex condition holds, since:

$$2 \cdot c(\{v_i, v_j, v_k\}) = \tfrac{4}{3}(c(\{v_i, v_j\}) + c(\{v_j, v_k\}) + c(\{v_i, v_k\}))\,.$$

This construction gives a polynomial time algorithm to solve 2-MICROAGGREGATION which improves on the 2-approximation from [28].

Observe that a similar construction for $(\|\cdot\|_1^w, \text{rad})$-2-CLUSTER does not work, since the cluster cardinality is not bounded by three. Also, even if d satisfies the triangle inequality, the corresponding cost function c would not satisfy the simplex condition, since for the small example of three vertices u, v, w with $d(u, v) = d(u, w) = 1$ and $d(v, w) = 2$, the cost with respect to radius would give $1 = c(\{u, v, w\}) < \frac{1}{2}(c(\{u, v\}) + c(\{u, w\}) + c(\{v, w\})) = 2$. Similar problems arise for the other so far unresolved variants of $(\|\cdot\|, f)$-2-CLUSTER.

At last, we would like to point out that the running times presented in this section all assume the worst-case in which there are $\mathcal{O}(n^2)$ pairs of vertices with small distance to each other; a property that might be avoided for certain specific clustering tasks. We further believe that an augmenting path strategy which is specifically tailored to the above problems can also yield significant improvement on the worst-case running time.

2.3 Computational Lower Bounds

As our attempts to find polynomial algorithms to solve versions of $(\|\cdot\|, f)$-2-CLUSTER seem to have reached an end, we move on to a search for lower bounds. We first check the case $k = 3$ and then settle the remaining open questions regarding the restriction to $k = 2$.

2.3.1 The Case $k \geq 3$

The problem variant $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER with the specific choice of d being the Euclidean distance was discussed in [4] under the name r-GATHER (where r takes the role of k) and was there shown to be NP-complete for $k \geq 7$. In [8] this result was strengthened by a reduction from EXACT-t-COVER to $k \geq 3$, however for a type of problem where the cluster center exists as an input vertex but is assigned to a different cluster (i.e., with the radius of a cluster

P calculated by: $\min\{\max\{d(x,y)\colon y \in P\}\colon x \in V\}$) which does not comply with our definition. We establish different reductions which show NP-hardness for all variants of $(\|\cdot\|, f)$-k-CLUSTER with $k \geq 3$. We also reduce from the problem EXACT-t-COVER, formally given by:

EXACT-t-COVER

Input: A universe $X = \{x_1, \ldots, x_n\}$ and a collection $C = \{S_1, \ldots, S_r\}$ of subsets of X, such that each S_i, $i \in \{1, \ldots, r\}$, has cardinality t.

Question: Does there exist a subset $C' \subseteq C$ (*exact cover*) that is a partition of X?

EXACT-t-COVER is known to be NP-hard for all $t \geq 3$ by [33].

Theorem 8
The problem $(\|\cdot\|, \mathrm{rad})$-$k$-CLUSTER is NP-hard for each $k \geq 3$ and all choices of $\|\cdot\| \in \{\|\cdot\|_\infty, \|\cdot\|_\infty^w, \|\cdot\|_1^w\}$, even with the restriction to distances d which satisfy the triangle inequality.

Proof. We reduce from EXACT-k-COVER. Let $S_1, \ldots, S_r$ be subsets of the universe $\{x_1, \ldots, x_n\}$, with $|S_i| = k$, an instance of EXACT-k-COVER and let $\ell := r - \frac{n}{k}$ (exactly the number of sets not included in an exact cover). We construct a graph $G = (V, E)$ for $(\|\cdot\|, \mathrm{rad})$-$k$-CLUSTER with a vertex set V built from the following three types of vertices:

- $u_1, \ldots, u_n$ representing $x_1, \ldots, x_n$,
- $w_1, \ldots, w_r$ representing $S_1, \ldots, S_r$ and
- y_i^j for $i \in \{1, \ldots, \ell\}$ and $j \in \{1, \ldots, k-1\}$, vertices which will be clustered with the w-vertices corresponding to sets which are not in the exact cover.

The set E contains the following edges, all of weight 1:

- $\{u_i, w_j\}$ for all $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, r\}$ with $x_i \in S_j$,
- $\{y_i^1, w_j\}$ for each $i \in \{1, \ldots, \ell\}$ and $j \in \{1, \ldots, r\}$ and
- $\{y_i^1, y_i^h\}$ for each $i \in \{1, \ldots, \ell\}$ and $h \in \{2, \ldots, k-1\}$.

We claim that there exists a k-cluster for G which only contains clusters of radius 1 if and only if there exists an exact cover for $S_1, \ldots, S_r$.

Let $\mathfrak{P}$ be a k-cluster for G which only contains clusters of radius 1, and let d be the distance on $V \times V$ induced by the edges of G. For each $i \in \{1, \ldots, \ell\}$, let P_i denote the cluster in $\mathfrak{P}$ containing y_i^2, as $k \geq 3$, a vertex y_i^j with index $j = 2$ is always included in G. Since y_i^1 is the only vertex at distance 1 from y_i^2,

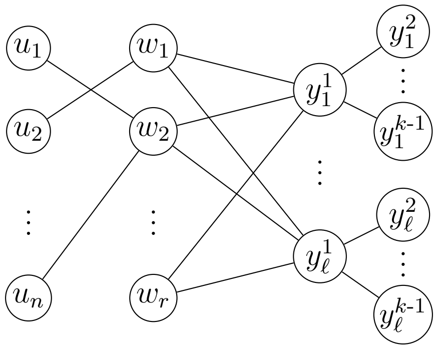


Figure 1: Illustration of the reduction for Theorem 8 .

it follows that y_i^1 is included as the unique central vertex in P_i which means that $P_i \subseteq \{v \in V \colon d(v, y_i^1) \leq 1\}$. As $\{v \in V \colon d(v, y_i^1) = 1\} = \{y_i^1, \dots, y_i^{k-1}\} \cup \{w_1, \dots, w_r\}$ and $|P_i| \geq k$, it follows that at least ℓ of the vertices $w_1, \dots, w_r$ are included in the clusters $P_1, \dots, P_\ell$, none of which contain a vertex from $\{u_1, \dots, u_n\}$. Since $d(u_i, u_j) \geq 2$ for all $i \neq j$, a cluster in $\mathfrak{P}$ which contains two vertices from $\{u_1, \dots, u_n\}$ has to contain at least one of the vertices w_z as central vertex. Such a cluster then has to be a subset of $\{w_z\} \cup \{u_i \colon x_i \in S_z\}$. There are only $\frac{n}{k}$ vertices from $\{w_1, \dots, w_r\}$ which lie in such a cluster, so $\mathfrak{P}$ has to contain exactly the clusters $\{w_z\} \cup \{u_i \colon x_i \in S_z\}$ for all $w_z \notin P_1 \cup \dots \cup P_\ell$ in order to include all vertices u_i in a cluster of radius 1. This means that the sets S_z with $\{w_z\} \cup \{u_i \colon x_i \in S_z\} \in \mathfrak{P}$ build an exact cover for $\{x_1, \dots, x_n\}$. It also follows that all clusters in a k-cluster of maximum radius 1 contain at most $k + 1$ vertices.

Conversely, for any exact cover $S \subseteq \{S_1, \dots, S_r\}$ the union of the sets $\{w_z\} \cup \{u_i \colon x_i \in S_z\}$ for all z with $S_z \in S$ and $\{y_i^1, \dots, y_i^{k-1}\} \cup \{w_{j_i}\}$ for all $i \in \{1, \dots, \ell\}$ where $\{S_1, \dots, S_r\} \setminus S = \{S_{j_1}, \dots, S_{j_\ell}\}$ yields a k-cluster of radius 1 for G.

If $\operatorname{rad}(P) > 1$ for some cluster P in a k-cluster $\mathfrak{P}$ for G, it follows that $\operatorname{rad}(P) \geq 2$; observe that since G only has edges of weight 1, all shortest paths have integer length. This means that the global cost of $\mathfrak{P}$ with respect to radius and $\|\cdot\|^{w}_{\infty}$ is at least $2k$, so strictly larger than the global cost of a k-cluster of maximum radius 1 for this norm, which is $k + 1$ by the above stated property of k-cluster of maximum radius 1 for G. Also, the global cost of $\mathfrak{P}$ with respect to $\|\cdot\|^{w}_{1}$ is at least $kr + \frac{n}{k} + k$ (at least k vertices produce a cost of 2), while a k-cluster of maximum radius 1 with respect to this norm yields a global cost of $kr + \frac{n}{k}$ (each vertex produces a cost of 1). In summary, there exists an exact cover for $S_1, \dots, S_r$ if and only if there exists a solution for $(\|\cdot\|, \operatorname{rad})$-$k$-CLUSTER of global cost 1, $k + 1$ and $kr + \frac{n}{k}$ for norm $\|\cdot\|_{\infty}, \|\cdot\|^{w}_{\infty}$ and $\|\cdot\|^{w}_{1}$, respectively. □

In the above proof of Theorem 8 a "yes"-instance of Exact-k-Cover is equivalent to the existence of a k-cluster with maximum radius 1. Since all distances in the constructed instance for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-cluster are integral, a k-cluster with maximum radius more than 1, contains a cluster of radius at least 2. This simple observation shows a gap of 2 for the maximum radius between "yes"- and "no"-instance for Exact-k-Cover, which implies:

Corollary 9
There is no $(2-\varepsilon)$-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-cluster in polynomial time for any $k \geq 3$ and any $\varepsilon > 0$, unless $\mathsf{P} = \mathsf{NP}$, even if d satisfies the triangle inequality.

If we alter the reduction used for Theorem 8 for $k \geq 4$ to reduce to Exact-$(k-1)$-Cover, we can conclude that for a "yes"-instance for the covering problem all clusters in a k-cluster of maximum radius 1 for the corresponding graph G contain exactly k vertices. This yields a gap of 2 also for the maximum weighted radius, which implies:

Corollary 10
There is no $(2-\varepsilon)$-approximation for $(\|\cdot\|^w_\infty, \mathrm{rad})$-$k$-cluster in polynomial time for any $k \geq 4$ and any $\varepsilon > 0$, unless $\mathsf{P} = \mathsf{NP}$, even if d satisfies the triangle inequality.

For diameter, we need a different construction, since for this measure, the vertices $u_1, \ldots, u_n$ have to also be at distance 1 to enable some of them to be in the same cluster. With such distances, we need a different structure which makes sure that a solution of diameter 1 does not build clusters only containing vertices from $u_1, \ldots, u_n$.

Theorem 11
The problem $(\|\cdot\|, \mathrm{diam})$-$k$-cluster is NP-hard for each $k \geq 3$ and all choices for $\|\cdot\| \in \{\|\cdot\|_\infty, \|\cdot\|^w_\infty, \|\cdot\|^w_1\}$, even with the restriction to distances d which satisfy the triangle inequality.

Proof. We reduce from Exact-t-Cover with $t = (k-1)^2$. Let $S_1, \ldots, S_r$ be subsets of $\{x_1, \ldots, x_n\}$, with $|S_i| = t$, an instance of Exact-t-Cover and let $\ell := r - \frac{n}{t}$. We construct a graph G for $(\|\cdot\|, \mathrm{diam})$-$k$-cluster with the following three types of vertices:

- $u_1, \ldots, u_n$ representing $x_1, \ldots, x_n$,
- $w_i^1, \ldots, w_i^{k-1}$ representing S_i for $i \in \{1, \ldots, r\}$ and
- $v_1, \ldots, v_\ell$ which model the selection of the ℓ sets not in the cover.

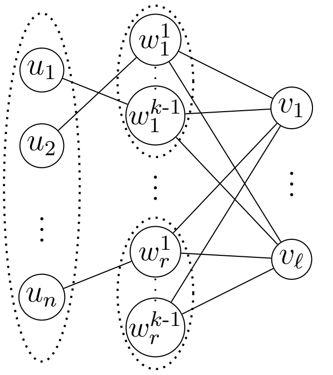


Figure 2: Illustration of the reduction for Theorem 11 . Dotted ellipses surround cliques.

The graph G contains the following edges, all of of weight 1:

- edges such that the set $\{u_1, \ldots, u_n\}$ is a clique,
- edges such that the set $\{w_i^1, \ldots, w_i^{k-1}\}$ is a clique for each $i \in \{1, \ldots, r\}$,
- each v_h, $h \in \{1, \ldots, \ell\}$, is connected to all w_i^z with $i \in \{1, \ldots, r\}$ and $z \in \{1, \ldots, k-1\}$ and
- to model the sets, edges connect u_j to one of the vertices $w_i^1, \ldots, w_i^{k-1}$ if $x_j \in S_i$, more precisely, for every set S_i pick and fix an arbitrary partition $S_i = S_i^1 \cup \cdots \cup S_i^{k-1}$ into disjoint subsets of cardinality $k-1$ and connect u_j with w_i^z if $u_j \in S_i^z$.

We claim that there exists an exact cover for $S_1, \ldots, S_r$ if and only if a there exists a k-cluster of maximum diameter 1 for G.

Let $\mathfrak{P}$ be a k-cluster for G which only contains clusters of diameter 1, and let d be the distance on $V \times V$ induced by the edges of G. Since $d(w_i^y, w_j^z) = 2$ for $i \neq j$ and any $y, z \in \{1, \ldots, k-1\}$ and $d(v_q, v_p) = 2$ for $q \neq p$, each v_h can only be in a cluster of cardinality at least k and diameter 1, if v_h is contained in the cluster $N_i^h := \{v_h, w_i^1, \ldots, w_i^{k-1}\}$ for some $i \in \{1, \ldots, r\}$. The only possibilities for a cluster of cardinality at least k and diameter 1 which contains a vertex w_i^z is either exactly the cluster $C_i^z := \{w_i^z\} \cup \{u_j : x_j \in S_i^z\}$ or the cluster N_i^h for some $h \in \{1, \ldots, \ell\}$. As $|N_i^h| = |C_i^z| = k$ and $N_i^h \cap C_i^z = \{w_i^z\}$ for all $i \in \{1, \ldots, r\}$, $h \in \{1, \ldots, \ell\}$ and $z \in \{1, \ldots, k-1\}$, it follows that for each $i \in \{1, \ldots, r\}$ either $N_i^h \in \mathfrak{P}$ for some $h \in \{1, \ldots, \ell\}$ or $C_i^z \in \mathfrak{P}$ for all $z \in \{1, \ldots, k-1\}$. As there are exactly $\ell = r - \frac{n}{t}$ vertices v_h, which have to be included in some cluster N_i^h, $\mathfrak{P}$ contains exactly $\frac{n}{t}$ clusters $C_i^1, \ldots, C_i^{k-1}$ which is possible if and only if $\{S_i : C_i^1 \in \mathfrak{P}\}$ is an exact cover; observe that all sets in $\mathfrak{P}$ are disjoint, so the $(k-1)\frac{n}{t}$ sets of type C_i^z in $\mathfrak{P}$ contain exactly $(k-1)(k-1)\frac{n}{t} = n$ vertices from $\{u_1, \ldots, u_n\}$.

Conversely, for every exact cover $S \subseteq \{S_1, \ldots, S_r\}$, the union of the set $\{C_i^1, \ldots, C_i^{k-1} \colon S_i \in S\}$ and $\{N_{j_h}^h \colon 1 \leq h \leq \ell\}$ where $\{S_1, \ldots, S_r\} \setminus S =: \{S_{j_1}, \ldots, S_{j_h}\}$ is a k-cluster of diameter 1 for G.

Specific to the norm, it follows that there exists a k-cluster of global cost 1 for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER if and only if $S_1, \ldots, S_r$ is a "yes"-instance for Exact-t-Cover. Further, each cluster that has the possibility of being of diameter 1 contains exactly k vertices, so $S_1, \ldots, S_r$ is a "yes"-instance for Exact-t-Cover if and only if there exists a solution of global cost k for $(\|\cdot\|_\infty^w, \text{diam})$-$k$-CLUSTER. At last, a solution for diameter with weighted 1-norm of global cost $n + r(k-1) + \ell$ is possible if and only if each cluster has diameter 1, hence if and only if $S_1, \ldots, S_r$ is a "yes"-instance for Exact-t-Cover. □

The reduction shown in the above proof of Theorem 11 is also a gap-reduction with a gap of 2 for the maximum diameter between "yes"- and "no"-instance for EXACT-t-COVER. The maximum cardinality of a cluster in an optimal solution in case of a "yes"-instance for EXACT-t-COVER is k, so the reduction also gives a gap of 2 for the maximum weighted diameter and hence implies:

Corollary 12

There is no $(2-\varepsilon)$-approximation in polynomial time for $(\|\cdot\|_\infty, \text{diam})$- or $(\|\cdot\|_\infty^w, \text{diam})$-$k$-CLUSTER for any $k \geq 3$ and any $\varepsilon > 0$, unless $\mathsf{P} = \mathsf{NP}$, even if d satisfies the triangle inequality.

The construction in the proof of Theorem 8 almost also shows the same hardness result for average distortion. The only problem is that an optimal solution requires clusters of cardinality $k+1$ which means that with respect to $\|\cdot\|_\infty^w$, we have a global cost of k, which is also achieved by a cluster of cardinality k in which 1 vertex has distance 2 from the central vertex. We will therefore use a third reduction for average distortion which represents each set by $k-1$ vertices as in the construction for diameter and combines this with the idea to use stars with $k-1$ vertices to disable $r - \frac{n}{t}$ sets from being used to "cover" $u_1, \ldots, u_n$, as used for radius.

Theorem 13

The problem $(\|\cdot\|, \text{avg})$-k-CLUSTER is NP-hard for each $k \geq 3$ and all choices for $\|\cdot\| \in \{\|\cdot\|_\infty, \|\cdot\|_\infty^w, \|\cdot\|_1^w\}$, even with the restriction to distances d which satisfy the triangle inequality.

Proof. We reduce from EXACT-t-COVER with $t = (k-1)^2$. Let $S_1, \ldots, S_r$ be subsets of $\{x_1, \ldots, x_n\}$, with $|S_i| = t$, an instance of EXACT-t-COVER. We construct a graph G for $(\|\cdot\|, \text{avg})$-k-CLUSTER with the following vertices:

- $u_1, \ldots, u_n$ representing $x_1, \ldots, x_n$,

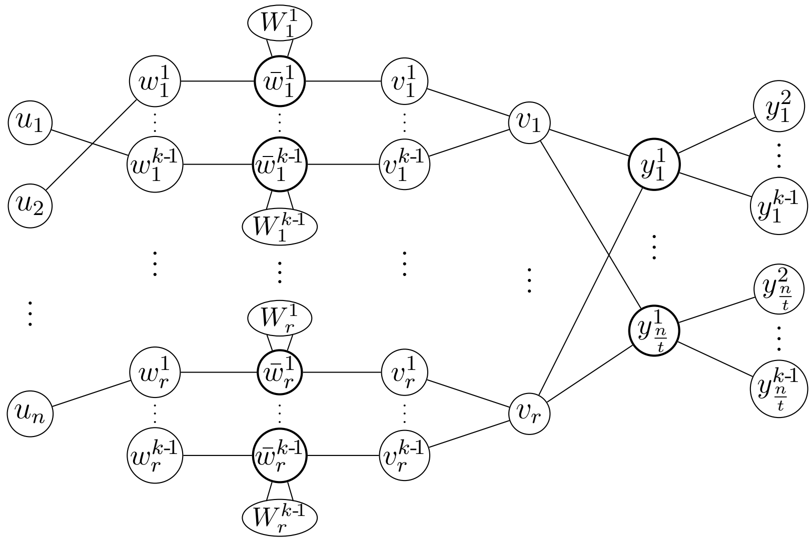


Figure 3: Illustration of the reduction for Theorem 13 . Thick vertices have to be central in a k-cluster of maximum radius 1.

- $w_i^1, \ldots, w_i^{k-1}$ representing S_i for $i \in \{1, \ldots, r\}$,
- $\bar{w}_i^z$ for all $i \in \{1, \ldots, r\}$ and $z \in \{1, \ldots, k-1\}$,
- a set of $k-2$ vertices W_i^z for all $i \in \{1, \ldots, r\}$ and $z \in \{1, \ldots, k-1\}$,
- $v_i, v_i^1, \ldots, v_i^{k-1}$ for all $i \in \{1, \ldots, r\}$ and
- y_i^j for $i \in \{1, \ldots, \frac{n}{t}\}$ and $j \in \{1, \ldots, k-1\}$.

The graph G contains the following edges, all of weight 1:

- like for diameter, pick and fix for every set S_i an arbitrary partition $S_i = S_i^1 \cup \cdots \cup S_i^{k-1}$ into disjoint subsets of cardinality $k-1$ and connect u_j with w_i^z if $u_j \in S_i^z$,
- $\{w, \bar{w}_i^z\}$ for all $w \in W_i^z$, $i \in \{1, \ldots, r\}$ and $z \in \{1, \ldots, k-1\}$ (the graph induced by the vertices $W_i^z \cup \{\bar{w}_i^z\}$ is a star graph with center $\bar{w}_i^z$),
- $\{v_i, v_i^h\}$ for all $i \in \{1, \ldots, r\}$ and $h \in \{1, \ldots, k-1\}$,
- $\{w_i^z, \bar{w}_i^z\}$ and $\{\bar{w}_i^z, v_i^z\}$ for all $i \in \{1, \ldots, r\}$ and $z \in \{1, \ldots, r\}$,
- $\{v_i, y_j^1\}$ for all $i \in \{1, \ldots, r\}$ and $j \in \{1, \ldots, \frac{n}{t}\}$,
- $\{y_i^1, y_i^h\}$ for each $i \in \{1, \ldots, \frac{n}{t}\}$ and $h \in \{2, \ldots, k-1\}$.

We claim that there exists an exact cover for $S_1, \dots, S_r$ if and only if a there exists a k-cluster for G such that each cluster has average distortion $\frac{k-1}{k}$.

Let $\mathfrak{P}$ be a k-cluster for G such that each cluster has average distortion $\frac{k-1}{k}$, and let d be the distance on $V \times V$ induced by the edges of G. First of all, observe that any cluster of cardinality at least k has average distortion $\frac{k-1}{k}$ if and only if it hast radius 1 and cardinality k. Similar to the proof of Theorem 8, denote for each $i \in \{1, \dots, \frac{n}{t}\}$ by P_i the cluster in $\mathfrak{P}$ which contains y_i^2. With the property of P_i having radius 1 and cardinality k for each $i \in \{1, \dots, \frac{n}{t}\}$, it follows that exactly $\frac{n}{t}$ of the vertices $v_1, \dots, v_r$ are included in some cluster P_i, which otherwise only contains the vertices $y_i^1, \dots, y_i^{k-1}$. A similar argument applies for a cluster P which contains a vertex from W_i^r, as these vertices also only have one vertex $(\bar{w}_i^z)$ at distance 1, which then has to be central for P; this cluster then always contains the whole set W_i^r. So, denote by P_i^z the cluster containing the set W_i^r and $\bar{w}_i^z$. For each $i \in \{1, \dots, r\}$ and $r \in \{1, \dots, k-1\}$, the set P_i^z contains either v_i^z or w_i^z, as these are the only other vertices at distance 1 from $\bar{w}_i^z$. For each of the exactly $r - \frac{n}{t}$ vertices v_i which are not contained in any of the clusters $P_1, \dots, P_{\frac{n}{t}}$, the only option for a cluster of cardinality k and radius 1 is the cluster $V_i := \{v_i, v_i^1, \dots, v_i^{k-1}\}$; observe that a vertex v_i^h with $h \in \{1, \dots, k-1\}$ cannot be central for a cluster of cardinality $k \geq 3$ as the only vertices at distance 1 from v_i^h are v_i and $\bar{w}_i^z$ and the latter one already has to be the central vertex for P_i^z. Also, the only vertices at distance 1 from v_i which are not in some cluster P_j are $v_i^1, \dots, v_i^{k-1}$. Hence there are exactly $r - \frac{n}{t}$ indices i in $\{1, \dots, r\}$ such that V_i is a cluster in $\mathfrak{P}$. For all $i \in \{1, \dots, r\}$ for which V_i is a cluster in $\mathfrak{P}$, the cluster P_i^z contains w_i^z for all $z \in \{1, \dots, k-1\}$ since v_i^z is not available as kth vertex in P_i^z. Again similar to the proof of Theorem 8, there are exactly enough vertices w_i^z not included in a set of the form P_i^z in $\mathfrak{P}$ to build clusters of radius 1 for the vertices $\{u_1, \dots, u_n\}$ if and only if the sets S_i with indices $i \in \{1, \dots, r\}$ for which V_i is not a cluster in $\mathfrak{P}$ are an exact cover.

Conversely, for every exact cover $S \subseteq \{S_1, \dots, S_r\}$, a k-cluster of average distortion $\frac{k-1}{k}$ for G can be built with the following sets:

- $\{w_i^z\} \cup \{u_j \colon x_j \in S_i^z\}$ and $P_i^z \cup \{v_i^z\}$ for all i with $S_i \in S$, $z \in \{1, \dots, k-1\}$,
- V_i and $P_i^z \cup \{w_i^z\}$ for all i with $S_i \notin S$, $z \in \{1, \dots, k-1\}$ and
- $\{v_{j_i}, y_i^1, \dots, y_i^{k-1}\}$ for all $i \in \{1, \dots, \frac{n}{t}\}$ with $S = \{S_{j_1}, \dots, S_{j_{\frac{n}{t}}}\}$.

So, there exists an exact cover for $S_1, \dots, S_r$ if and only if there exists a solution for $(\|\cdot\|_\infty, \text{avg})$-$k$-CLUSTER of global cost $\frac{k-1}{k}$. Further, for any k-cluster for G, a global cost of $k-1$ with respect to average distortion and $\|\cdot\|_\infty^w$ is only possible if each cluster has radius 1 and cardinality k; a cluster with $k' > k$ vertices gives a global cost of at least $k' - 1 > k - 1$ and a cluster of radius larger than 1 contains at least one vertex at distance 2 from the central vertex which gives a global cost of at least k. So, there exists an exact cover for

$S_1, \ldots, S_r$ if and only if there exists a solution for $(\|\cdot\|^w_\infty, \text{avg})$-$k$-CLUSTER of global cost $k-1$.

At last, for any k-cluster for G, a global cost of $n+r(k-1)k$ with respect to average distortion and $\|\cdot\|^1_\infty$ is only possible if each vertex contributes exactly the minimum cost of $\frac{k-1}{k}$ to the global cost. Hence there exists an exact cover for $S_1, \ldots, S_r$ if and only if there exists a solution for $(\|\cdot\|^w_1, \text{avg})$-$k$-CLUSTER of global cost $n + r(k-1)k$. □

In the above reduction used to prove Theorem 13, a "yes"-instance for EXACT-t-COVER corresponds to a graph for which there exists a k-cluster with a maximum weighted average distortion of $k-1$. Integrality again implies that a "no"-instance for EXACT-t-COVER corresponds to graph for which the maximum weighted average distortion of any k-cluster is at least k. This gives the following result.

Corollary 14

There is no $(\frac{k}{k-1} - \varepsilon)$-approximation for $(\|\cdot\|^w_\infty, \text{avg})$-$k$-CLUSTER in polynomial time for any $k \geq 3$ and any $\varepsilon > 0$, unless $\mathsf{P} = \mathsf{NP}$, even if d satisfies the triangle inequality.

2.3.2 Harsh Consequences for Approximation & Parameterisation

The previous section shows that $(\|\cdot\|, f)$-k-CLUSTER remains NP-hard even for k restricted to 3 for all choices of local cost $f \in \{\text{rad}, \text{diam}, \text{avg}\}$ and global cost $\|\cdot\| \in \{\|\cdot\|^w_\infty, \|\cdot\|_\infty\}$, so the bound on the cardinality is obviously not a parameter which helps with tractability (at least not without a combination with further parameters). A closer look at reductions used for these hardness results have further consequences, not just for parameterisation but also for approximabilty, which we summarise here.

When considering parameterisation for optimisation problems, the first parameter that comes to mind is usually the optimum value. For our definition of $(\|\cdot\|, f)$-k-CLUSTER, this choice is a bit odd, as we allow rational weights, which means that these and hence also the optimum value, can be scaled pretty much arbitrarily without changing the general nature of the problem, i.e., the optimum solution remains unchanged.

The most reasonable option to repair this is a restriction to positive integer weights. For the 1-norm, this restriction yields an optimum value which is too large to be an interesting parameter, for all choices of f; observe that even for $k = 2$, the global cost is at least $\frac{n}{2}$, as at least half of the vertices then cause a cost of at least 1. For the (weighted) infinity norm, the reductions in the previous section only use integer distances and yield optimum values which are either constant or in $\mathcal{O}(k)$. Since the NP-hardness in all cases already holds for $k = 3$, these results also disqualify parameterisation by optimum value for the infinity norms, and formally imply:

Corollary 15

$(\|\cdot\|, f)$-k-CLUSTER parameterised by optimum value (and k) is para-NP*-hard for all choices of $f \in \{\text{rad}, \text{diam}, \text{avg}\}$ and $\|\cdot\| \in \{\|\cdot\|_{\infty}^{w}, \|\cdot\|_{\infty}\}$, even if d satisfies the triangle inequality.*

Another idea related to the restriction to integer distances is considering the number of different values in the input as parameter, as suggested by the concept of parameterisation by the "number of numbers", see [35]. The constructions used in proofs of the previous section only contain edges of weight 1. Even taking this concept further, and considering all pairwise distances between vertices does not yield an angle for parameterisation. Fixing the distances which are not defined by edges in the proofs of the previous section to all be equal to 2, the NP-hardness reductions still work and only use two (or three, if the distance 0 for a vertex to itself is counted) different distance values, which yields:

Corollary 16

$(\|\cdot\|, f)$-k-CLUSTER parameterised by the number of different distance values (and k) is para-NP*-hard, for all choices of $f \in \{\text{rad}, \text{diam}, \text{avg}\}$ and $\|\cdot\| \in \{\|\cdot\|_{1}^{w}, \|\cdot\|_{\infty}^{w}, \|\cdot\|_{\infty}\}$, even if d satisfies the triangle inequality.*

The results from the previous section also yield negative results for approximability. The following result gives a very clear explanation why triangle inequality for d is an important requirement to develop polynomial time approximation for $(\|\cdot\|, f)$-k-CLUSTER. If we consider instances of $(\|\cdot\|, f)$-k-CLUSTER for which the induced distance d can violate the triangle inequality, additional edges of a large weight w in the constructions for Theorem 8 and Theorem 11 can be used to amplify the gap between a "yes"- and a "no"-instance of EXACT-t-COVER strictly monotonically with w which gives:

Proposition 17

If d violates the triangle inequality, there is no polynomial time constant-factor approximation for $(\|\cdot\|, f)$-k-CLUSTER, for all choices of $f \in \{\text{rad}, \text{diam}, \text{avg}\}$ and $\|\cdot\| \in \{\|\cdot\|_{1}^{w}, \|\cdot\|_{\infty}^{w}, \|\cdot\|_{\infty}\}$, unless P = NP.

Proof. Let $G = (V, E)$ be the graph constructed in the proof of Theorem 8 for a given instance I of EXACT-k-COVER, so there exists a k-cluster of maximum radius 1 for G if and only if I is a "yes"-instance. Further, every k-cluster of maximum radius 1 for G only contains sets of maximum cardinality $k+1$. If I is a "no"-instance, any k-cluster for G contains at least one set S of radius larger than 1, so for every choice of $v \in S$ there exists at least one vertex $v' \in S \setminus \{v\}$ such that $\{v, v'\} \notin E$. If we now turn the graph G into a complete graph with additional edges of weight w, it follows that the radius of such a cluster S is w. This also means that the average distortion for such a cluster S is larger than $\frac{w}{n}$, while the minimum average distortion of a k-cluster for G is $\frac{k}{k+1}$ in case I is a "yes"-instance. For every norm, the global cost grows strictly monotonically with the local cost. This means that the gap between I being a "yes" or "no"-instance for the optimum value of a k-cluster for G with respect to radius or average distortion with any norm grows strictly monotonically with w. As this is true for every value of w, a constant-factor approximation in polynomial time for $(\|\cdot\|, f)$-k-CLUSTER with $f \in \{\text{rad,avg}\}$ with any norm would solve EXACT-k-COVER which however is NP-hard for any $k \geq 3$.

For diameter, we use the same idea and turn the graph G constructed in the proof of Theorem 11 into a complete graph by adding edges of weight w. Similarly, it follows that there exists a k-cluster of maximum diameter 1 if the corresponding instance I of EXACT-t-COVER is a "yes"-instance, while the maximum diameter is w if I is a "no"-instance. So, a constant-factor approximation in polynomial time for $(\|\cdot\|, \text{diam})$-k-CLUSTER with any norm would solve the NP-hard problem EXACT-t-COVER. □

This result demonstrates pretty clearly why we will dedicate a lot of attention to the properties of d with respect to triangle inequality. Already here we want to mention that, while it discards any constant-factor approximation for a general scenario, the constructions used to show this negative result all require both an unbounded number of bad edges and an unbounded increase of edge-weight. For instances which do not carry this property, we will discuss possibilities to design approximation procedures later in Section 5.

First however, we want finish the investigation of the general complexity of $(\|\cdot\|, f)$-k-CLUSTER with respect to k for which there are still some open cases with $k = 2$.

2.3.3 Hard Cases for $k = 2$

Section 2.2 only provided polynomial time solvability for roughly half of the variants of $(\|\cdot\|, f)$-2-CLUSTER. We will now complete the complexity picture for $k = 2$ by showing that the remaining variants are already NP-hard even when restricted to the smallest reasonable cardinality.

We start with $(\|\cdot\|_1^w, \text{rad})$-2-CLUSTER and reduce from the restriction of the vertex cover problem to cubic graphs, formally defined by:

CUBIC VERTEX COVER

Input: Graph $G = (V, E)$ such that all vertices $v \in V$ have degree 3.

Output: A set $C \subseteq V$ (*vertex cover*) of minimum cardinality such that $e \cap C \neq \emptyset$ for all $e \in E$.

Cubic Vertex Cover seen as optimisation problem is APX-hard by [5][3]. The reduction we now give from Cubic Vertex Cover to $(\|\cdot\|_1^w, \mathrm{rad})$-2-cluster will not only show NP-hardness for the associated decision problem but also prove APX-hardness for the optimisation problem.

Theorem 18

$(\|\cdot\|_1^w, \mathrm{rad})$-2-cluster *interpreted as decision problem is* NP*-hard, and interpreted as optimisation problem is* APX*-hard, even with the restriction to distances d which satisfy the triangle inequality.*

Proof. For the sake of simplicity, we first describe a polynomial reduction from the decision problem Cubic Vertex Cover and later argue how this construction can be interpreted as an L-reduction. Let $G = (V, E)$ with $V = \{v_1, \dots, v_n\}$ and $m := |E|$ be the input graph for Cubic Vertex Cover. We construct a graph $G' = (V', E')$ for $(\|\cdot\|_1^w, \mathrm{rad})$-2-cluster defined by the vertex set $V' := \{v_i^1, v_i^2 \colon 1 \leq i \leq n\} \cup \{v_e \colon e \in E\}$ and edge set $E' := \{\{v_i^1, v_i^2\} \colon 1 \leq i \leq n\} \cup \{\{v_i^1, v_e\} \colon v_i \in e\}$ with weights $w_E(\{v_i^1, v_i^2\}) = 1$ and $w_E(\{v_i^1, v_e\}) = 2$. We claim that G has a vertex cover of cardinality ℓ if and only if there exists a solution for $(\|\cdot\|_1^w, \mathrm{rad})$-2-cluster with global cost $2n + 2\ell + 2m$.

For any vertex cover C of G, we construct a 2-cluster for G' by first building clusters $\{v_i^1, v_i^2\}$ for all $i \in \{1, \dots, n\}$. We then pick (arbitrarily, if there is a choice) for every edge $e = \{u, w\} \in E$ an index $i \in \{1, \dots, n\}$ such that $v_i \in C$ and $v_i \in \{u, w\}$ and add the vertex v_e to the cluster $\{v_i^1, v_i^2\}$. As C is a vertex cover for G, we can assign each vertex v_e in such a way and arrive at a 2-cluster $\mathfrak{P}$ for G' which contains only the following two types of clusters:

- $\{v_i^1, v_i^2\} \in \mathfrak{P}$ for all $i \in \{1, \dots, n\}$ with $v_i \notin C$,
- for all $i \in \{1, \dots, n\}$ with $v_i \in C$, $\mathfrak{P}$ contains a cluster P_i with $\{v_i^1, v_i^2\} \subseteq P_i$ and $P_i \backslash \{v_i^1, v_i^2\} \subseteq \{v_e \colon \exists 1 \leq j \leq n \colon e = \{v_i, v_j\}\}$. With v_i^1 considered as central vertex, P_i has radius at most 2, as all vertices v_e with $e = \{v_i, v_j\}$ for some $j \in \{1, \dots, n\}$ have distance 2 from v_i^1.

Considering, w.l.o.g., a vertex numbering such that $C = \{v_1, \dots, v_\ell\}$, the global cost of $\mathfrak{P}$ with respect to radius and weighted 1-norm is hence at most:

$$\sum_{i=\ell+1}^{n} 2 \cdot \mathrm{rad}(\{v_i^1, v_i^2\}) + \sum_{i=1}^{\ell} |P_i| \cdot \mathrm{rad}(P_i) \ \leq\ 2(n - \ell) + 2 \cdot \sum_{i=1}^{\ell} |P_i| \,.$$

As the union of all the clusters P_i with $i \in \{1, \dots, \ell\}$ contains exactly all vertices v_e, $e \in E$ and all vertices v_i^1, v_i^2 with $i \in \{1, \dots, \ell\}$, it follows that

[3] The much older paper [54] is often cited for this result, but only gives APX-hardness for 4-regular graphs.

$\sum_{i=1}^{\ell} |P_i| = |E| + 2\ell$. The global cost of $\mathfrak{P}$ as solution for $(\|\cdot\|_1^w, \mathrm{rad})$-2-CLUSTER is hence at most $2(n-\ell) + 2(m+2\ell) = 2n + 2\ell + 2m$.

Conversely, let $\mathfrak{P}$ be a 2-cluster for G' such that the global cost with respect to radius and 1-norm is at most $2n + 2\ell + 2m$. We define for this solution a cost function c on V' by $c(v) := \mathrm{rad}(P)$ for all $v \in P$ and $P \in \mathfrak{P}$. The global cost of $\mathfrak{P}$ with respect to $(\|\cdot\|_1^w, \mathrm{rad})$ can hence be computed by $\sum_{v \in V'} c(v)$. Observe first that from the structure of the graph G' it immediately follows that $c(v) \geq 1$ and $c(v) \in \mathbb{N}$ for all $v \in V'$. We consider the possible costs $c(v)$ for all types of vertices:

- For all $i \in \{1, \dots, n\}$ and $h \in \{1, 2\}$, $c(v_i^h) = 1$ if and only if $\{v_i^1, v_i^2\} \in \mathfrak{P}$.
- For all $e \in E$, we know that $c(v_e) \geq 2$.
- For any $e = \{v_i, v_j\} \in E$, $c(v_e) = 2$ is only possible if $\{v_i^1, v_j^1\} \cap P \neq \emptyset$ for the set $P \in \mathfrak{P}$ with $v_e \in P$.
- For any $e = \{v_i, v_j\} \in E$, $c(v_e) = 3$ is only possible if $\{v_i^2, v_j^2\} \cap P \neq \emptyset$ for the set $P \in \mathfrak{P}$ with $v_e \in P$.

Assume that $C := \{v_i \colon c(v_i^1) \geq 2\}$ is not a vertex cover of size ℓ for G. If $|C| > \ell$, we see that, since $c(v_i^1) \geq 2$ if and only if $c(v_i^2) \geq 2$, the global cost of $\mathfrak{P}$ exceeds the assumed value, as:

$$\sum_{v \in V'} c(v) \geq \sum_{i=1}^{n} (c(v_i^1) + c(v_i^2)) + 2m \geq 2 \cdot 2|C| + 2(n - |C|) + 2m > 2n + 2\ell + 2m\,.$$

If there is some edge $e = \{v_i, v_j\}$ which is not covered by C, the sets $\{v_i^1, v_i^2\}$ and $\{v_j^1, v_j^2\}$ are both in $\mathfrak{P}$ by the definition of C, hence $c(v_e) \geq 4$. So let $\bar{E} \subseteq E$ be the set of edges which are not covered by C. It follows that:

$$\begin{aligned} 2n+2\ell+2m &\geq \sum_{v_i \in C} (c(v_i^1)+c(v_i^2)) + \sum_{v_i \notin C} (c(v_i^1)+c(v_i^2)) + \sum_{e \in \bar{E}} c(v_e) + \sum_{e \in E \setminus \bar{E}} c(v_e) \\ &\geq 4 \cdot |C| + 2(n - |C|) + 4 \cdot |\bar{E}| + 2(m - |\bar{E}|) \\ &= 2n + 2m + 2(|C| + |\bar{E}|) \end{aligned}$$

This means that $|C| \leq \ell - |\bar{E}|$, so if C is not already a vertex cover for G, we can greedily chose for each edge in $\bar{E}$ an arbitrary adjacent vertex to cover it and arrive at a vertex cover for G of cardinality at most ℓ.

At last, with CUBIC VERTEX COVER and $(\|\cdot\|_1^w, \mathrm{rad})$-2-CLUSTER seen as optimisation problems given by $(I_{vc}, S_{vc}, m_{vc}, min)$ and (I_2, S_2, m_2, min), respectively, the above reduction can be seen as an L-reduction (f, g, β, γ) as follows:

- f is given by the above polynomial reduction which creates $G' \in I_2$ from an input $G \in I_{vc}$.
- g maps an input $G \in I_{vc}$ and a 2-cluster $\mathfrak{P}$ for $f(G)$ to the set C as described in the proof above; observe that the cost $c(v)$ can be computed from G and $\mathfrak{P}$ in polynomial time.
- Since $m = 3n/2$ and $\ell \geq n/2$ for a cubic graph, we have $2n + 2\ell + 2m \leq 12k$. With $\beta := 15$ this means $m_2^*(f(G)) \leq \beta m_{vc}^*(G)$.
- By the definition of g, it follows that $m_{vc}(G, g(G, \mathfrak{P})) \leq \frac{1}{2}(m_2(f(G), \mathfrak{P}) - 2n - 2m)$ for each graph $G \in I_{vc}$ with n vertices and m edges and each 2-cluster $\mathfrak{P}$ for $f(G)$. Given any vertex cover of size ℓ for G, the reduction above showed that there exists a 2-cluster of global cost at most $2(n + m + \ell)$ for G', which in particular means that $m_2^*(f(G))$ can be bounded by $2(n+m+m_{vc}^*(G))$. With $E_P(x, g(x, y)) := m_P(x, g(x, y)) - m_P^*(x)$ denoting the *error* for a solution $y \in S(x)$ to an instance $x \in I$ of a minimisation problem $P = (I, S, m, \min)$ and $\gamma = \frac{1}{2}$, it follows that:

$$\begin{aligned} E_{vc}(G, g(G, \mathfrak{P})) &= m_{vc}(G, g(G, \mathfrak{P})) - m_{vc}^*(G) \\ &\leq \tfrac{1}{2}(m_2(f(G), \mathfrak{P}) - 2n - 2m) - \tfrac{1}{2}(m_2^*(f(G)) - 2n - 2m) \\ &= \tfrac{1}{2}(m_2(f(G), \mathfrak{P}) - m_2^*(f(G))) \\ &= \gamma E_2(f(G), \mathfrak{P})\,. \end{aligned}$$

The L-reduction (f, g, β, γ) hence translates the APX-hardness from Cubic Vertex Cover to $(\|\cdot\|_1^w, \mathrm{rad})$-2-cluster. □

The reduction above cannot be adapted for the cases of $(\|\cdot\|, f)$-2-cluster which were not shown to be polynomial time solvable so far. We therefore consider the following variation of Satisfiability for the remaining cases:

(3, 3)-Satisfiability (or (3, 3)-SAT)

Input: Boolean formula F in conjunctive normal form such that each clause contains at most 3 literals and each variable occurs both positively and negatively in F and overall at most 3 times.

Question: Does there exist a satisfying assignment for F?

This restricted version of the classical satisfiability problem remains NP-hard by [62].

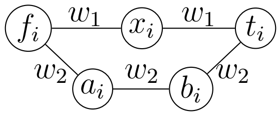


Figure 4: Illustration of the gadget introduced for variable v_i in Theorem 19.

Theorem 19
$(\|\cdot\|_\infty^w, \text{avg})$-, $(\|\cdot\|_\infty, \text{avg})$- *and* $(\|\cdot\|_\infty^w, \text{rad})$-2-CLUSTER *are* NP*-hard, for the latter two even with the restriction to distances d which satisfy the triangle inequality.*

Proof. Let $v_1, \ldots, v_n$ be the variables and $c_1, \ldots, c_m$ be the clauses of a $(3,3)$-SAT formula F. We construct a graph $G = (V, E)$ by introducing five vertices t_i, f_i, x_i, a_i, b_i for each v_i and edges $\{x_i, f_i\}, \{x_i, t_i\}$ of weight w_1 and $\{a_i, f_i\}, \{b_i, t_i\}, \{a_i, b_i\}$ of weight w_2 as in Figure 4. Also, for each clause c_j, introduce a vertex y_j and edges of weight w_2 from y_j to each literal in c_j, i.e., to f_i if $\bar{v}_i$ is a literal in c_j and to t_i if v_i is a literal in c_j. We will assign values for w_1 and w_2 differently for each problem variant such that a 2-cluster for G has global cost (and hence maximum (weighted) cost of each cluster) at most 1 if and only if the following *assignment properties* hold:

- Each x_i has to be in a cluster of cardinality 2 with either t_i or f_i (this reflects the assignment for v_i to be the vertex not clustered with x_i).
- Each y_j is in a cluster with 1 adjacent vertex, so t_i (or f_i) for some i with v_i ($\bar{v}_i$) being a literal in c_j (this literal satisfies the clause).
- For all $i \in \{1, \ldots, n\}$, the vertices a_i and b_i lie in the same cluster which otherwise can only possibly contain either t_i or f_i (in case we do not need the variable v_i to satisfy any clause).

Assuming $w_1 \geq w_2$, the induced distance d on G satisfies:

- $d(x_i, v) \geq w_1 + w_2$ for all $v \in V \setminus \{x_i, t_i, f_i\}$,
- $d(t_i, y_j) \geq 3w_2$ for all i, j such that v_i is no literal in c_j,
- $d(f_i, y_j) \geq 3w_2$ for all i, j such that $\bar{v}_i$ is no literal in c_j,
- $d(y_j, v) \geq 2w_2$ for all $v \in V \setminus \{t_i, f_i \colon 1 \leq i \leq n\}$.

These distances imply that a 2-cluster which does not satisfy the assignment properties contains at least one cluster of either a cardinality at least 3 and radius at least w_1 (some vertex x_i not properly clustered), or a radius of at least $2w_2$ (some vertex y_j not in a cluster with adjacent vertex). We now consider each problem variant and define respective weights w_1 and w_2.

For $(\|\cdot\|_\infty^w, \mathrm{rad})$-2-CLUSTER we choose $w_1 = \frac{1}{2}$ and $w_2 = \frac{1}{3}$. With these weights, a cluster P in a 2-cluster for G of weighted radius at most 1 can have radius w_1 only if it has cardinality 2 and otherwise has radius w_2 and cardinality at most 3. As $2w_2 > w_1$, a solution for $(\|\cdot\|_\infty^w, \mathrm{rad})$-2-CLUSTER of global cost at most 1 fulfils the assignment properties.

For $(\|\cdot\|_\infty, \mathrm{avg})$-2-CLUSTER we choose $w_1 = 2$ and $w_2 = \frac{3}{2}$. With these weights, all pairs of distinct vertices in G have a distance at least $\frac{3}{2}$ and hence the average distortion of every cluster is at least $\frac{3}{2}(|P|-1)/|P|$, which means that the maximum cardinality of a cluster of average distortion 1 is 3. A cluster of cardinality 3 has average distortion at most 1 only if it has radius $\frac{3}{2} = w_2$. A cluster of cardinality 2 has average distortion at most 1 only if it has radius at most $2 = w_1$. As again $2w_2 > w_1$, this means that a solution for $(\|\cdot\|_\infty, \mathrm{avg})$-2-CLUSTER of global cost at most 1 fulfils the assignment properties.

For $(\|\cdot\|_\infty^w, \mathrm{avg})$-2-CLUSTER we choose $w_1 = 1$ and $w_2 = \frac{1}{2}$ but also have to add some more edges; observe that so far, the induced distance d satisfies the triangle inequality, so by Proposition 7, a 2-cluster could be computed in polynomial time, hence our construction cannot be complete. With the current definition we have $2w_2 = w_1$ which yields that clusters of the form $\{y_i, y_j\}$ or $\{a_i, y_j\}$ could also have a weighted average distortion of 1 as there could be a shortest path from y_j to y_i or a_i via two edges of weight $\frac{1}{2}$. If we add edges $\{y_i, y_j\}$ for all $i \neq j$ and $\{a_i, y_j\}, \{b_i, y_j\}$ for all $i \in \{1, \dots, n\}, j \in \{1, \dots, m\}$ each of weight 2, these types of clusters have a weighted average distortion of 2. Other clusters in a 2-cluster which yield a violation of the assignment property have either cardinality at least 3 and radius at least w_1, which yields a weighted average distortion of at least $\frac{3}{2}$, or radius at least $\min\{w_1 + w_2, 3w_2\} = \frac{3}{2}$, so weighted average distortion at least $\frac{3}{2}$. A solution for $(\|\cdot\|_\infty^w, \mathrm{avg})$-2-CLUSTER of global cost at most 1 fulfils the assignment properties.

Finally, there exists a 2-cluster with assignment properties for G (for any choice of the weights w_1, w_2) if and only if the formula F is satisfiable:

Given a 2-cluster $\mathfrak{P}$ for G with assignment property, the vertices of G corresponding to the clauses are clustered with their satisfying literal and for each variable v_i either $\{t_i, x_i\} \in \mathfrak{P}$ or $\{f_i, x_i\} \in \mathfrak{P}$, so the assignment $v_i = \mathit{true}$ if and only if $\{f_i, x_i\} \in \mathfrak{P}$ is a satisfying assignment for F.

Conversely, given a satisfying assignment ϕ for F, build a partition from the union of the sets $\{\{x_i, t_i\}, \{f_i\} \colon \phi(v_i) = \mathit{false}\}$, $\{\{x_i, f_i\}, \{t_i\} \colon \phi(v_i) = \mathit{true}\}$ and $\{\{a_i, b_i\} \colon 1 \leq i \leq n\}$, and put for each $j \in \{1, \dots, m\}$ the vertex y_j into the cluster which contains the assignment of the literal (an arbitrary literal if there is a choice) which satisfies c_j, i.e., if v_i ($\bar{v}_i$) is a literal in c_j and $\phi(v_i) = \mathit{true}$ ($\phi(v_i) = \mathit{false}$) put y_j in the cluster containing t_i (f_i). As F is an instance of (3,3)-SAT, at most 2 clause vertices are assigned to the same cluster. If there is some i such that $\{t_i\}$ or $\{f_i\}$ remains a cluster of cardinality 1, merge this cluster with $\{a_i, b_i\}$. The resulting partition is a 2-cluster with assignment properties for G. □

$k=2$	rad	diam	avg
$\lVert\cdot\rVert_\infty$	in P *(Edge Cover)* (Proposition 5)	in P *(Simplex Cover)* (Proposition 7)	NP-complete (Theorem 19)
$\lVert\cdot\rVert_\infty^w$	NP-complete (Theorem 19)	in P *(Simplex Cover)* (Proposition 7)	NP-complete (Theorem 19)
$\lVert\cdot\rVert_1^w$	APX-hard (Theorem 18)	in P *(Simplex Matching)* (Proposition 6)	in P *(Weighted Edge Cover)* (Theorem 4)

Table 1: Summary of the complexity of all problem variants for $k = 2$.

With this result, we have completed our investigation of the complexity of $(\lVert\cdot\rVert, f)$-k-CLUSTER with respect to k. At last, we want to give an overview of the diverse behaviour of the different problem variants for $k = 2$ in Table 1. Observe that already for the restriction to $k = 2$, the generally NP-hard variant $(\lVert\cdot\rVert_\infty^w, \text{avg})$-$k$-CLUSTER becomes polynomial time solvable for the restriction to distances d which satisfy the triangle inequality. The next section will discuss approximation strategies for this restriction for general values for k.

3 General Metric Instances

We will now discuss polynomial time approximations for $(\|\cdot\|, f)$-k-CLUSTER but only consider the case where d satisfies the triangle inequality. Despite the chosen title of this section, this does not necessarily make d a metric in the classical definition of this word, as we allow the existence of $u \neq v$ with $d(u, v) = 0$ (violation of the so-called *identity of indiscernibles* property of metrics); recall that by the formal definition of $(\|\cdot\|, f)$-k-CLUSTER, the distance d derived from edge-weights is just non-negative, symmetric and reflexive. There exist different names for distances with these properties, so for the lack of a fixed unified notion we chose to still use the simple term *metric* to describe that distance d for $(\|\cdot\|, f)$-k-CLUSTER satisfies triangle inequality. In the formal results, we always clearly state that we just require triangle inequality for d (and not identity of indiscernibles). This restriction is not just reasonable in many scenarios but in some sense necessary to achieve any kind of approximation as Proposition 17 indicates.

This work is not the first attempt to find polynomial approximations for $(\|\cdot\|, f)$-k-CLUSTER. Known approximation results for clustering with size constraints include a 9-approximation from [8] for LOAD BALANCED FACILITY LOCATION without facility cost, which is related to $(\|\cdot\|_1^w, \text{avg})$-$k$-CLUSTER here, but with the additional constraint that each customer should be assigned to the nearest open facility. The techniques used for this result highly rely on the additional constraint, which unfortunately means that they can not be applied here. Other approximations for this problem instead relax the constraint that each cluster needs to contain at least k vertices; [40] for example presents a $2k$-approximation which constructs clusters of cardinality at least $k/3$. We will see that for our problem such an approximation factor can be achieved without relaxing the cardinality constraints. In general, our results however do not extend to LOAD BALANCED FACILITY LOCATION, since the addition of facility costs yields a very different type of problem; we especially lose the upper bound of $2k - 1$ on the cardinality of clusters in an optimal solution from Theorem 2. Some other related results will however prove to be quite helpful.

The main idea of the approximations developed in this section is to exploit the structural connections between the different problem variants to especially translate results from one variant to another and to find related problems with the help of the given graph formulation. This approach will yield polynomial time approximations for eight of the nine problem variants of $(\|\cdot\|, f)$-k-CLUSTER.

3.1 Greedy Approximation

The problem that we call $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER is discussed under the name r-GATHER in [4], where r takes the role of k. The greedy concept for the

2-approximation presented there can be altered and also used to compute a 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER which gives the following result; recall that we use $\mathrm{opt}(G, d, \|\cdot\|, f, k)$ to denote the global cost of an optimal solution for $(\|\cdot\|, f)$-k-CLUSTER on G with distance d.

Theorem 20

$(\|\cdot\|_\infty, \mathrm{rad})$- *and* $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER *are 2-approximable in polynomial time for all $k \geq 2$, if d satisfies the triangle inequality.*

Proof. Let $G = (V, E)$ be the input graph with induced distance d. By a binary search among all values in $\{d(v, v') : v, v' \in V\}$, we search for the smallest value D such that the procedure described below to build a k-cluster of maximum radius D is successful.

For a fixed D, we first build a partition of V in the following way: Beginning with $i = 1$ and $V_1 := V$ we iteratively, until $V_i = \emptyset$, choose arbitrarily some $z_i \in V_i$ and build clusters $P(z_i) := \{v \in V_i : d(z_i, v) \leq D\}$ and set $V_{i+1} = V_i \setminus P(z_i)$. This yields a partition of V into a finite number of clusters $P(z_i)$. Let q be the number of clusters created by this strategy. If each cluster $P(z_i)$ contains at least k vertices, we have found a k-cluster of maximum radius D.

Some of the clusters $P(z_i)$ however might have a cardinality of less than k. In this case, we try to reassign some vertices to adjust the cardinalities. Observe that by the strategy used to build the clusters, possible vertices outside $P(z_i)$ at distance at most D from z_i can only be in clusters $P(z_j)$ with $j < i$. Hence, we define the sets $S(i, j) := \{v \in P(z_j) \setminus \{z_j\} : d(v, z_i) \leq D\}$ for all $1 \leq j < i$ to collect all vertices which can be moved from cluster $P(z_j)$ to cluster $P(z_i)$ without increasing the radius of $P(z_i)$ to be more than D. If $\sum_{j=1}^{i-1} |S(i, j)| < k - |P(z_i)|$ for some $i \in \{1, \ldots, q\}$ there are not enough vertices to move to cluster $P(z_i)$, so we delete this clustering attempt and try again for a larger value for D. Otherwise, we try to move some vertices in $S(i, j)$ from $P(z_j)$ to $P(z_i)$, $1 \leq j < i \leq q$, in order to arrive at a partition which is a k-cluster. Moving some vertices from $S(i, j)$ into $P(z_i)$ to increase the cardinality of $P(z_i)$ might mean that the cardinality of $P(z_j)$ decreases below k and hence requires moving some vertices from $S(j, \ell)$ into $P(z_j)$ for some $\ell < j$. This kind of ripple effect is the reason why we solve this problem of moving vertices in $S(i, j)$ to create a k-cluster by modelling it as a max-flow problem with the following network:

- The network has a source s and target t.

- For each $i \in \{1, \ldots, q\}$ we create a vertex z_i' representing $P(z_i)$ in the network. If $|P(z_i)| > k$ we add an arc from s to z_i' of capacity $|P(z_i)| - k$. If $|P(z_i)| < k$ we add an arc from z_i' to t of capacity $k - |P(z_i)|$ to t.

- For each $v \in \bigcup_{i=1}^{q-1} \bigcup_{j=1}^{i-1} S(i, j)$, create a vertex v' in the network and arcs of capacity 1 from v' to z'_i for all i with $v \in S(i,j)$ for some j and also from z'_j to v' for all j with $v \in S(i,j)$ for some i.

There exists a maximum flow of $\sum_{i=2}^{q} \max\{0, k - |P(z_i)|\}$ from s to t in this network if and only if we can find a reassignment of the vertices in the sets $S(i,j)$ to turn $P(z_1), \ldots, P(z_q)$ into a k-cluster: Moving a vertex $v \in S(i,j)$ from $P(z_i)$ to $P(z_j)$ corresponds to a flow of 1 in the network from z'_i to v' and then to z'_j. If $|P(z_i)| > k$, at most $|P(z_i)| - k$ vertices are allowed to be moved out of $P(z_i)$ which corresponds to the capacity of the arc from s to z'_i. If $|P(z_i)| < k$, exactly $k - |P(z_i)|$ vertices have to be moved into $P(z_i)$ without replacement, saturating the capacities of the arc from z'_i to t. Max-Flow can be solved in time $O(m \cdot n)$ [45, 52] on a directed graph with n vertices and m edges. If we find a flow of size $\sum_{i=2}^{q} \max\{0, k - |P(z_i)|\}$, we can build a k-cluster for V with maximum radius D, otherwise we abort and try a larger value for D.

We claim that the procedure described above is successful for $D = 2r^*$ with $r^* = \operatorname{opt}(G, d, \|\cdot\|_\infty, \operatorname{rad}, k)$. The vertices z_i chosen while computing a solution for $D = 2r^*$ belong to different clusters in an optimal solution, since vertices in the same cluster have a distance of at most $2r^*$ (observe that this is false if d violates the triangle inequality). Since at most one vertex from each optimal cluster was chosen to be some z_j, there are enough vertices at distance at most $2r^*$ from each such vertex to distribute them among the sets $P(z_j)$ such that each has a cardinality of at least k. A similar reasoning proves that the greedy procedure is successful for $D = d^*$ with $d^* = \operatorname{opt}(G, d, \|\cdot\|_\infty, \operatorname{diam}, k)$. In case of diameter, the vertices z_j can not belong to the same cluster in the optimal solution as soon as their distance is larger than d^*. Maximum radius of D together with triangle inequality shows that the k-cluster computed for $D = d^*$ is a 2-approximation for $(\|\cdot\|_\infty, \operatorname{diam})$-$k$-cluster. □

Remark 2*:* A natural greedy procedure for $(\|\cdot\|_\infty, \operatorname{avg})$-$k$-cluster could build up the sets $P(z_i)$ by successively adding $\operatorname{argmin}\{d(v, z_i) \colon v \in V_i \setminus P(z_i)\}$ until $\operatorname{avg}(P(z_i))$ exceeds D, but moving vertices from $S(i,j)$ to $P(z_i)$ could unfortunately increase the average distortion of $P(z_j)$.

3.2 Constraint Forest Problems

Some variants of the very broad class of constraint forest problems introduced in [39] also has a close relation to clustering with lower bounds. The function f on subsets of vertices used to design constraints in the integer programming formulation used for constraint forest problems can be defined to model our minimum cardinality requirement for a partition. This very general framework introduced in [39] includes a large collection of different specific problems. We will however mostly use the following specific problem from the class of constraint forest optimisation problems:

Lower Capacitated Tree Partitioning

Input: Graph $G = (V, E)$, edge-weights $w_E \colon E \to \mathbb{Q}_+$, capacity $k \in \mathbb{N}$.

Output: A set $E' \subseteq E$ minimising $\sum_{e \in E'} w_E(e)$ such that each $v \in V$ occurs in at least one $e \in E'$ and each component in the graph induced by E' is a tree with at least k vertices.

While [39] also gives an approximation procedure for the whole class of constraint forest problems, this specific problem Lower Capacitated Tree Partitioning can be 2-approximated in polynomial time with an application of the much simpler greedy approximation presented in [43].

Proposition 21

$(\|\cdot\|_1^w, \text{avg})$-$k$-cluster *is $2k$-approximable in polynomial time for all $k \geq 2$, if d satisfies the triangle inequality.*

Proof. Let $G = (V, E)$ be an instance of $(\|\cdot\|_1^w, \text{avg})$-$k$-cluster with induced distances d. We consider solving Lower Capacitated Tree Partitioning with capacity k on $G' = (V, V \times V)$ with edge-weights computed via d. Any solution $P_1, \ldots, P_s$ for $(\|\cdot\|_1^w, \text{avg})$-$k$-cluster of global cost L on $G = (V, E)$ can be interpreted as a solution of cost L for Lower Capacitated Tree Partitioning on G'; a spanning forest for G' with connected components $P_1, \ldots, P_s$ and cost $\|(\text{avg}(P_1), \ldots, \text{avg}(P_s))\|_1^w$ is given by the edge set:

$$\bigcup_{i=1}^{s} \{\{c_i, v_i\} \colon v_i \in P_i\} \quad \text{with} \quad c_i = \operatorname{argmin}\{\sum_{v \in P_i} d(v, c) \colon c \in P_i\}\,.$$

Conversely, any minimal solution $\bar{E} \subset V \times V$ for Lower Capacitated Tree Partitioning for G' of cost L can be interpreted as a solution for $(\|\cdot\|_1^w, \text{avg})$-$k$-cluster with global cost at most $k \cdot L$. Let $\mathfrak{C}$ be the set of connected components of the graph induced by $\bar{E}$. Any component $C \in \mathfrak{C}$ which contains a path with more than $2k-1$ vertices can be split into two connected components, each of cardinality at least k by deleting an edge (i.e., reducing the cost of the partition). Further, we can assume that for all components $C \in \mathfrak{C}$ there is at least one $c \in C$ such that $C \setminus \{c\}$ is a forest of trees each of maximum cardinality k. This can be seen as follows: Start at an arbitrary vertex as candidate for c and let $T_1^c, \ldots, T_{s_c}^c$ be the connected components of $C \setminus \{c\}$. If there is an index i such that $|T_i^c| > k$, consider the neighbour of c in T_i^c as new candidate and iterate this procedure. Observe that the index i for which $|T_i^c| > k$ has to be unique; otherwise, T_i^c and $C \setminus V[T_i^c]$ are both trees of cardinality at least k and the tree partitioning can be altered by deleting the edge which connects c with T_i^c. (In case that the tree partitioning is not chosen locally minimal in this sense, the above described procedure to find a suitable vertex c is constructive and can be used to alter the partitioning accordingly in polynomial time.) By the same argument, it follows that a suitable vertex c will be reached after at most k iterations.

With the simple observation that the distance of a vertex in T_i^c to c only depends on the cost of the edges of the tree partition in the subtree T_i^c, the triangle inequality implies that $d(v, c) \leq c(E[T_i^c])$ for all $v \in T_i^c$. Considering c as central vertex for the cluster $V[C]$, this gives the following relation between the partition $\mathfrak{P}$ given by $\{V[C] \colon C \in \mathfrak{C}\}$ and the cost L of the tree partition $\bar{E}$:

$$\begin{aligned}
\operatorname{avg}(\mathfrak{P}) &= \sum_{C \in \mathfrak{C}} |C| \cdot \operatorname{avg}(C) \\
&\leq \sum_{C \in \mathfrak{C}} \sum_{v \in C} d(v, c) \\
&\leq \sum_{C \in \mathfrak{C}} \sum_{i=1}^{s_c} \sum_{v \in T_i} d(v, c) \\
&\leq \sum_{C \in \mathfrak{C}} \sum_{i=1}^{s_c} \sum_{v \in T_i^c} |T_i^c| \cdot c(E[T_i^c]) \\
&\leq \sum_{C \in \mathfrak{C}} \sum_{i=1}^{s_c} \sum_{v \in T_i^c} k \cdot c(E[T_i^c]) \\
&\leq \sum_{C \in \mathfrak{C}} k \cdot E[C] \\
&= k \cdot L\,.
\end{aligned}$$

Since Lower Capacitated Tree Partitioning can be 2-approximated, this yields a $2k$-approximation for $(\|\cdot\|_1^w, \operatorname{avg})$-$k$-cluster. □

Remark 3*:* Theorem 4 showed that $(\|\cdot\|_1^w, \operatorname{avg})$-2-cluster can be solved in polynomial time which also translates to solving Lower Capacitated Tree Partitioning with capacity $k = 2$; finding a tree partitioning with capacity 2 is hence equivalent to the computation of a weighted edge cover.

Essential for the result above is excluding paths of length $2k$ in all components C, but this property does not prevent C from containing arbitrarily many vertices. For $(\|\cdot\|_1^w, \operatorname{diam})$- or $(\|\cdot\|_1^w, \operatorname{rad})$-$k$-cluster we need an upper bound on the cardinality to prove an approximation ratio. We therefore consider Lower Capacitated Path Partitioning, the restriction of Lower Capacitated Tree Partitioning to paths as connected components. On weighted graphs for which the weights obey the triangle inequality, [39] provides a 4-approximation for Lower Capacitated Path Partitioning.

Proposition 22

$(\|\cdot\|_1^w, \mathrm{diam})$-$k$-CLUSTER is $8(k-1)$-approximable in polynomial time for all $k \geq 2$, if d satisfies the triangle inequality.

Proof. Consider for any input $G = (V, E)$ with induced distances d for the problem $(\|\cdot\|_1^w, \mathrm{diam})$-$k$-CLUSTER, the complete graph $G' = (V, V \times V)$ with d as input for path partitioning. Let $P_1, \ldots, P_s$ be an optimal solution for $(\|\cdot\|_1^w, \mathrm{diam})$-$k$-CLUSTER with $|P_i| \leq 2k-1$ (transformed with Corollary 1). For each $i \in \{1, \ldots, s\}$, a cheapest spanning path for P_i has a cost of at most $(|P_i|-1) \cdot \mathrm{diam}(P_i)$. Building a cheapest spanning path for each set P_i hence gives a solution of LOWER CAPACITATED PATH PARTITIONING on G' of cost at most

$$\begin{aligned}\sum_{i=1}^{s}(|P_i|-1)\cdot\mathrm{diam}(P_i) &= \sum_{i=1}^{s}\frac{|P_i|-1}{|P_i|}\cdot|P_i|\cdot\mathrm{diam}(P_i))\\ &\leq \tfrac{2k-2}{2k-1}\cdot\sum_{i=1}^{s}|P_i|\cdot\mathrm{diam}(P_i))\\ &= \tfrac{2k-2}{2k-1}\cdot\mathrm{opt}(G,d,\|\cdot\|_1^w,\mathrm{diam},k)\,.\end{aligned}$$

This especially implies that the cost T^* of an optimal path partitioning for G' is at most $\frac{2k-2}{2k-1} \cdot \mathrm{opt}(G, d, \|\cdot\|_1^w, \mathrm{diam}, k)$.

Let $\tilde{E} \subseteq V \times V$ be a solution for LOWER CAPACITATED PATH PARTITIONING for G' of cost T. Let $P'_1, \ldots, P'_s$ be the vertex sets corresponding to the connected components of the graph induced by $\tilde{E}$. The partition $P'_1, \ldots, P'_s$ yields a solution for $(\|\cdot\|_1^w, \mathrm{diam})$-$k$-CLUSTER of global cost at most $(2k-1)T$; observe that any set P'_i contains at most $2k-1$ vertices as a path containing more than $2k-1$ vertices can be split into 2 paths by deleting an edge from $\tilde{E}$. Considering $\bar{E}$ to be a 4-approximation for LOWER CAPACITATED PATH PARTITIONING on G' computed with [43], the partition $P'_1, \ldots, P'_s$ gives a solution for $(\|\cdot\|_1^w, \mathrm{diam})$-$k$-CLUSTER of global cost at most:

$$\begin{aligned}(2k-1)T &\leq (2k-1)4T^*\\ &\leq (2k-1)4\cdot\tfrac{2k-2}{2k-1}\cdot\mathrm{opt}(G,d,\|\cdot\|_1^w,\mathrm{diam},k)\\ &\leq 8(k-1)\cdot\mathrm{opt}(G,d,\|\cdot\|_1^w,\mathrm{diam},k)\,.\end{aligned}$$

□

Remark 4: We believe that it is possible to improve the above result to yield a $6k$-approximation by also starting with the 2-approximation for LOWER CAPACITATED TREE PARTITIONING. The basic idea for this approach is to split up components C of large cardinality in the tree partitioning at the point c chosen as central vertex in the proof of the approximation ratio for Proposition 21 such that each edge in the tree partitioning only occurs in a cluster of at most $6k$ vertices. As a proper algorithmic description of this split appears to be rather complicated, we prefer here the cleaner result via path partitioning presented above.

3.3 Consequences for Other Problem Variants

One advantage of the unified model for $(\|\cdot\|, f)$-k-CLUSTER is that approximations for one variant also yield approximations for another. In case d satisfies the triangle inequality, the different local cost functions relate in the following way:

$$\operatorname{avg}(P_i) \leq \operatorname{rad}(P_i) \leq \operatorname{diam}(P_i) \leq 2 \cdot \operatorname{rad}(P_i) \tag{1}$$

With this relation, Proposition 22 immediately yields:

Corollary 23
$(\|\cdot\|_1^w, \operatorname{rad})$-$k$-CLUSTER is $16(k-1)$-approximable in polynomial time for all $k \geq 2$, if d satisfies the triangle inequality.

By definition, the two ∞-norms also relate optimal values in the following way for every choice of $f \in \{\operatorname{rad}, \operatorname{diam}, \operatorname{avg}\}$:

$$\operatorname{opt}(G, d, f, \|\cdot\|_\infty^w, k) \geq k \cdot \operatorname{opt}(G, d, f, \|\cdot\|_\infty, k) \tag{2}$$

This equation is helpful to derive approximations for the weighted ∞-norm:

Proposition 24
$(\|\cdot\|_\infty^w, \operatorname{diam})$- and $(\|\cdot\|_\infty^w, \operatorname{rad})$-$k$-CLUSTER are 4-approximable in polynomial time for all $k \geq 2$, if d satisfies the triangle inequality.

Proof. Let for a given graph G with induced distances d the sets $P_1, \ldots, P_s$ be the 2-approximation for $(\|\cdot\|_\infty, \operatorname{diam})$-$k$-CLUSTER from Theorem 20. By Corollary 1, we can assume that $|P_i| \leq 2k - 1$. This yields:

$$\begin{aligned} &\max\{|P_i| \cdot \operatorname{diam}(P_i) \colon 1 \leq i \leq s\} \\ &\leq (2k-1) \cdot \max\{\operatorname{diam}(P_i) \colon 1 \leq i \leq s\} \\ &\leq 2(2k-1) \cdot \operatorname{opt}(G, d, \operatorname{diam}, \|\cdot\|_\infty, k) \end{aligned}$$

By Equation (2) this implies

$$\max\{|P_i| \cdot \operatorname{diam}(P_i) \colon 1 \leq i \leq s\} \leq (4k-2) \cdot \tfrac{1}{k} \cdot \operatorname{opt}(G, d, \operatorname{diam}, \|\cdot\|_\infty^w, k)$$

which makes $P_1, \ldots, P_s$ a 4-approximation for $(\|\cdot\|_\infty^w, \operatorname{diam})$-$k$-CLUSTER.

A similar reasoning can be used with a 2-approximation for $(\|\cdot\|_\infty, \operatorname{rad})$-$k$-CLUSTER in order to compute a 4-approximation for $(\|\cdot\|_\infty^w, \operatorname{rad})$-$k$-CLUSTER. If a cluster P in the approximate solution for $(\|\cdot\|_\infty, \operatorname{rad})$-$k$-CLUSTER contains more than $2k-1$ vertices, we remove exactly k vertices from it (keeping at least one of its central vertices with respect to radius) and build a new cluster $\bar{P}$ with them. By triangle inequality this cluster has a radius of at most $2 \cdot \mathrm{rad}(P)$. We repeat this splitting until all clusters have at most $2k-1$ vertices. Let $P_1', \ldots, P_s'$ be the clusters created from the approximation $P_1, \ldots, P_s$ for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER by removing vertices and let $\bar{P}_1, \ldots, \bar{P}_r$ be all newly created clusters of cardinality k. Since at least one central vertex of P_i remains in P_i', we know that $\mathrm{rad}(P_i') \leq \mathrm{rad}(P_i)$. This partition yields a solution for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER of size:

$$\begin{aligned}
&\max\{\max\{|P_i'| \cdot \mathrm{rad}(P_i) \colon 1 \leq i \leq s\}, \max\{|\bar{P}_j| \cdot \mathrm{rad}(\bar{P}_j) \colon 1 \leq j \leq r\}\} \\
\leq\ &\max\{\max\{(2k-1) \cdot \mathrm{rad}(P_i') \colon 1 \leq i \leq s\}, \max\{k \cdot \mathrm{rad}(\bar{P}_j) \colon 1 \leq j \leq r\}\} \\
\leq\ &\max\{\max\{(2k-1) \cdot \mathrm{rad}(P_i) \colon 1 \leq i \leq s\}, \max\{k \cdot (2 \cdot \mathrm{rad}(P_i)) \colon 1 \leq i \leq s\}\} \\
\leq\ &2k \cdot \max\{\mathrm{rad}(P_i) \colon 1 \leq i \leq s\} \\
\leq\ &4k \cdot \mathrm{opt}(G, d, \mathrm{rad}, \|\cdot\|_\infty, k)
\end{aligned}$$

By Equation (2), this means that $P_1', \ldots, P_s', \bar{P}_1, \ldots, \bar{P}_r$ is a 4-approximation for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER. □

For $(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER we do not have a result for $(\|\cdot\|_\infty, \mathrm{avg})$-$k$-CLUSTER to transfer. Interestingly, a variant with different local and global cost can be used instead:

Proposition 25

$(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER is $(4k-2)$-approximable in polynomial time for all $k \geq 2$, if d satisfies the triangle inequality.

Proof. We first show $\mathrm{opt}(G, d, \mathrm{avg}, \|\cdot\|_\infty^w, k) \geq \mathrm{opt}(G, d, \mathrm{diam}, \|\cdot\|_\infty, k)$. Consider any set P in an optimal solution for $(\|\cdot\|_\infty, \mathrm{avg})$-$k$-CLUSTER. Triangle inequality and $k \geq 2$ yields:

$$\begin{aligned}
|P| \cdot \mathrm{avg}(P) =\ &\min\left\{\sum_{p \in P} d(c,p) \colon c \in P\right\} \\
\geq\ &\min\{\max\{d(u,c) + d(v,c) \colon u, v \in P, u \neq v\} \colon c \in P\} \\
\geq\ &\max\{d(u,v) \colon u, v \in P\} = \mathrm{diam}(P)\,.
\end{aligned}$$

The approximation procedure from Theorem 20 with a simple additional step of splitting up clusters of cardinality more than $2k-1$ by Corollary 1 produces a 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER for which each set contains at most $2k-1$ vertices. The global cost of this partition with respect to weighted ∞-norm and average distortion is at most $2(2k-1) \cdot \mathrm{opt}(G, d, \mathrm{diam}, \|\cdot\|_\infty, k)$, and hence yields a $(4k-2)$-approximation for $(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER. □

	rad	diam	avg
$\lVert\cdot\rVert_\infty$	**2** (Theorem 20)	**2** (Theorem 20)	?
$\lVert\cdot\rVert_\infty^w$	4 (Proposition 24)	4 (Proposition 24)	$4k-2$ (Proposition 25)
$\lVert\cdot\rVert_1^w$	$16(k-1)$ (Corollary 23)	$8(k-1)$ (Proposition 22)	$2k$ (Proposition 21)

Table 2: Summary of the approximation ratios for all problem variants, bold values are optimal assuming $\mathsf{P} \neq \mathsf{NP}$.

This concludes the collection of results we found considering polynomial time approximations for $(\lVert\cdot\rVert, f)$-k-CLUSTER restricted to instances for which d satisfies the triangle inequality. The approximation ratios of these results are summarised in Table 2. The only provably optimal ratios, assuming $\mathsf{P} \neq \mathsf{NP}$ are the 2-approximations for $(\lVert\cdot\rVert_\infty, \text{rad})$- and $(\lVert\cdot\rVert_\infty, \text{diam})$-$k$-CLUSTER.

We did not succeed in finding a technique which yields a constant-factor approximation for $(\lVert\cdot\rVert_\infty, \text{avg})$-$k$-CLUSTER but believe the approximation for $(\lVert\cdot\rVert_\infty, \text{rad})$-$k$-CLUSTER to be a good heuristic. Average distortion with its lack of monotonicity is the most challenging local cost, but we did also not find any lower bound which would suggest that approximations for $(\lVert\cdot\rVert_\infty, \text{avg})$-$k$-CLUSTER are unlikely. The question whether there exists an approximation with provable performance ratio, constant or at least in $\mathcal{O}(k)$, for this problem variant hence remains open.

3.4 A Particular Algorithm for $k = 4$

As a last idea for this section on general metric instances, we want to present an approximation which exploits the unified model by combining the solutions for $k = 2$ derived in Section 2.2 for two different problem variants to compute an approximate solution for the restriction to $k = 4$. Although this restriction is very specific, this result can be seen as a first step towards better approximation ratios for the weighted 1-norm.

Explicitly, we will combine the SIMPLEX MATCHING approach for the problem variant $(\lVert\cdot\rVert_1^w, \text{diam})$-2-CLUSTER and the EDGE COVER approach for $(\lVert\cdot\rVert_1^w, \text{avg})$-2-CLUSTER. For this result, we need the following connection between $(\lVert\cdot\rVert_1^w, \text{diam})$-4-CLUSTER and $(\lVert\cdot\rVert_1^w, \text{avg})$-2-CLUSTER.

Lemma 26
Let $P_1, \dots, P_s$ with $|P_i| \le 3$ for all $i \in \{1, \dots, s\}$ be an optimal solution for $(\|\cdot\|_1^w, \mathrm{diam})$-2-CLUSTER on a graph G with distance d. Let $G' = (P, P \times P)$ be the graph with $P := \{p_1, \dots, p_s\}$ and edge-weights w defined by $w_{i,j} := w(\{p_i, p_j\}) := \min\{d(u,v)\colon u \in P_i, v \in P_j\}$, then:

$$\mathrm{opt}(G, d, \mathrm{diam}, \|\cdot\|_1^w, 4) \ge 3 \cdot \mathrm{opt}(G', w, \mathrm{avg}, \|\cdot\|_1^w, 2)\,.$$

Proof. Let $S_1, \dots, S_r$ be an optimal solution for $(\|\cdot\|_1^w, \mathrm{diam})$-4-CLUSTER on G and define $c(v) := \mathrm{diam}(S_i)$ for all $v \in S_i$, $i \in \{1, \dots, r\}$. This yields:

$$D^* := \mathrm{opt}(G, d, \mathrm{diam}, \|\cdot\|_1^w, 4) = \sum_{v \in V} c(v)\,.$$

Let $\tilde{G} = (P, \tilde{E})$ be the restriction of G' (edge-weights inherited) to the edges:

$$\tilde{E} := \bigcup_{k=1}^{r} \{\{p_i, p_j\}\colon (i \ne j) \wedge (P_i \cap S_k \ne \emptyset) \wedge (P_j \cap S_k) \ne \emptyset\}\,.$$

Observe that $|P_i| \le 3$ for all $i \in \{1, \dots, s\}$ implies that the minimum degree in $\tilde{G}$ is 1; each $v \in P_i$ lies in some set S_j, $j \in \{1, \dots, r\}$, with $|S_j| \ge 4$, so there exists a vertex $v' \in S_j$ with $v' \in P_{i'}$ and $i' \ne i$ which yields $\{p_i, p_{i'}\} \in \tilde{E}$. By the definition of $\tilde{G}$, we know that, for any $v \in P_i$

$$c(v) \ge \min\{w_{i,j}\colon 1 \le j \le s, \{p_i, p_j\} \in \tilde{E}\}\,. \tag{3}$$

Let $C \subseteq \tilde{E}$ be a minimum-weight edge cover for $\tilde{G}$. We claim that $3 \cdot w(C) \le D^*$ and consider three cases for edges C based on the cardinality of the neighbourhoods of vertices p_i in C, formally defined by $N_C(i) := \{r\colon \{p_i, p_r\} \in C\}$. First observe that if $|N_C(i)| > 1$, minimality of C yields:

$$w_{i,j} \le \min\{w_{h,j}\colon 1 \le h \le s, \{p_h, p_j\} \in \tilde{E}\} \text{ for all } j \in N_C(i)\,. \tag{4}$$

Case 1: $|N_C(i)| = |N_C(j)| = 1$ for some $j \in \{1, \dots, s\}$ with $\{p_i, p_j\} \in C$. As $\{p_i, p_j\} \in \tilde{E}$, there exists some $k \in \{1, \dots, r\}$ such that $P_i \cap S_k \ne \emptyset$ and $P_j \cap S_k \ne \emptyset$, so let $u_1^i, u_1^j \in S_k$ be two vertices with $u_1^i \in P_i$ and $u_1^j \in P_j$. By definition of the functions w and c, it follows that:

$$c(u_1^i) = c(u_1^j) = \mathrm{diam}(S_k) \ge d(u_1^i, u_1^j) \ge w_{i,j}\,.$$

By minimality of C, we know that $w_{i,j} \le w_{i,z_i} + w_{j,z_j}$ for any choice of $z_i, z_j \in \{1, \dots, s\}$ with $\{p_i, p_{z_i}\}, \{p_j, p_{z_j}\} \in \tilde{E}$, so especially for z_h such that $w_{h,z_h} = \min\{w_{h,x}\colon 1 \le x \le s, \{p_h, p_x\} \in \tilde{E}\}$, $h \in \{i, j\}$. By Equation (3) this means that for any two vertices $v_h \in P_h$, $h \in \{i, j\}$:

$$w_{i,j} \le w_{i,z_i} + w_{j,z_j} \le c(v_i) + c(v_j)\,.$$

As $|P_h| \geq 2$, let $v_h \in P_h \setminus \{u_1^h\}$ for $h \in \{1,2\}$, which gives:

$$c(P_i \cup P_j) := \sum_{v \in P_i \cup P_j} c(v) \geq c(u_1^i) + c(u_1^j) + c(v_i) + c(v_j) \geq 3 \cdot w_{i,j}\,.$$

Case 2: If $|N_C(i)| = 2$ let $N_C(i) = \{j,k\}$. Equation (4) yields $w_{h,z} \geq w_{h,i}$ for $h \in \{j,k\}$ and all $z \in \{1,\dots,s\}$ with $\{p_h,p_z\} \in \tilde{E}$. Equation (3) hence yields $c(v) \geq w_{i,h}$ for all $v \in P_h$, $h \in \{j,k\}$. Let the edge $\{p_i,p_h\}$ be in $\tilde{E}$ because of u_h^i, u_h^1 for $h \in \{j,k\}$, i.e., $u_h^i \in P_i$ and $u_h^1 \in P_h$ and there exist $y_h \in \{1,\dots,r\}$ such that $u_h^i, u_h^1 \in S_{y_h}$. By this definition, it follows that $c(u_h^i) \geq \operatorname{diam}(S_{y_h}) \geq d(u_h^i, u_h^1) = w_{i,h}$. If $u_j^i \neq u_k^i$, it follows that

$$c(P_i \cup P_j \cup P_k) \geq c(u_j^i) + c(u_k^i) + |P_j| \cdot w_{ij} + |P_k| \cdot w_{i,k} \geq 3 \cdot (w_{i,j} + w_{i,k})\,.$$

If $u_j^i = u_k^i$, it follows that $y_j = y_k =: y$ and hence $u_j^1, u_k^1 \in S_y$, which means that the edge $\{p_j,p_k\}$ is in $\tilde{E}$ with weight at most $d(u_j^1,u_k^1) \leq \operatorname{diam}(S_y) = c(u_j^i)$. Minimality of C yields $w_{j,k} + \min\{w_{i,x} \colon 1 \leq x \leq s\} \geq w_{i,j} + w_{i,k}$, hence Equation (3) yields $w_{j,k} + c(|P_i \setminus \{u_j^i\}|) \geq w_{i,j} + w_{i,k}$, which overall gives:

$$c(P_i) \geq c(u_j^i) + c(|P_i \setminus \{u_j^i\}|) \geq w_{j,k} + (w_{i,j} + w_{i,k} - w_{j,k}) \geq w_{i,j} + w_{i,k}\,,$$

which overall gives $c(P_i \cup P_j \cup P_k) \geq 3 \cdot (w_{i,j} + w_{i,k})$.

Case 3: If $|N_C(i)| \geq 3$, let $N_C(i) = \{i_1,\dots,i_t\}$. Equations (4) and (3) yield:

$$c(v_{i_j}) \geq w_{i,i_j} \text{ for all } v_{i_j} \in P_{i_j},\ j \in \{1,\dots,t\}\,. \tag{5}$$

Let for each $j \in \{1,\dots,t\}$, $u_j \in P_i$ and $v_j \in P_{i_j}$ be the vertices defining the edge $\{p_i,p_{i_j}\}$, i.e., there exists $x_j \in \{1,\dots,r\}$ such that $u_j, v_j \in S_{x_j}$. By this definition, it follows that:

$$c(u_j) = \operatorname{diam}(S_{x_j}) \geq d(u_j,v_j) \geq w_{i,i_j} \text{ for all } j \in \{1,\dots,t\}\,. \tag{6}$$

If $u_j = u_{j'}$ for some $j \neq j'$, it follows that $x_j = x_{j'}$ and consequently the edge $\{p_{i_j}, p_{i_{j'}}\}$ is in $\bar{E}$ and has a cost of at most $d(v_j,v_{j'})$. Minimality of C implies that $d(v_j,v_{j'}) \geq w_{i,i_j} + w_{i,i_{j'}}$. On the other hand, we have $c(v) = \operatorname{diam}(S_{x_j}) \geq d(v_j,v_{j'})$, for all $v \in S_{x_j}$, hence especially for $v \in \{v_j,v_{j'}\}$. With Equation (5), this gives:

$$\begin{aligned} c(P_{i_j} \cup P_{i_{j'}}) &\geq c(v_j) + c(v_{j'}) + c(P_{i_j} \setminus \{v_j\}) + c(P_{i_{j'}} \setminus \{v_{j'}\}) \\ &\geq 2(w_{i,i_j} + w_{i,i_{j'}}) + w_{i,i_j} + w_{i,i_{j'}} \\ &= 3(w_{i,i_j} + w_{i,i_{j'}})\,. \end{aligned} \tag{7}$$

Let M be a maximum matching for the graph $H = (\{1,\dots,t\}, \hat{E})$ with $\hat{E} = \{\{j,j'\} \colon u_j = u_{j'}\}$. By the definition of the edges, maximality of M

yields that for the unmatched indices $N := \{j\colon \{j,j'\} \notin M \,\forall\, 1 \le j' \le t\}$, we have $|\{u_j\colon j \in N\}| = |N|$. With Equations (4),(6) and (7) this yields:

$$\begin{aligned}
c(P_i) + \sum_{j=1}^{t} c(P_{i_j}) &\ge \sum_{\{j,j'\}\in M} c(P_{i_j} \cup P_{i_{j'}}) + c(P_i) + \sum_{j\in N} c(P_{i_j}) \\
&\ge \sum_{\{j,j'\}\in M} 3(w_{i,i_j} + w_{i,i_{j'}}) + \sum_{j\in N} c(P_{i_j} \cup \{u_j\}) \\
&\ge \sum_{\{j,j'\}\in M} 3(w_{i,i_j} + w_{i,i_{j'}}) + \sum_{j\in N} w_{i,i_j} |P_{i_j} \cup \{u_j\}| \\
&\ge 3 \cdot \sum_{j=1}^{t} w_{i,i_j}\,.
\end{aligned}$$

Let $C_1, \dots, C_x$ be the connected components (stars) of the graph induced by the edges in C, and let p_{i_j} be the center of C_{i_j} for each $j \in \{1, \dots, x\}$, then:

$$D^* = \sum_{v\in V} c(v) = \sum_{j=1}^{r} c(P_j) = \sum_{t=1}^{x} c(\bigcup_{p_j \in C_t} P_j) \ge \sum_{t=1}^{x} 3 \cdot \sum_{j \in N_C(i_t)} w_{i_t,j} = 3 \cdot w(C)\,.$$

At last, since $\tilde{G}$ is a restriction of G', $w(C)$ is at least the cost of a minimum-weight edge cover for G' and by the proof of Theorem 4 any minimal edge cover for G' yields a solution for $(\|\cdot\|_1^w, \text{avg})$-2-CLUSTER. □

With the help of this Lemma, we can show that:

Theorem 27
The problem $(\|\cdot\|_1^w, \text{diam})$-4-CLUSTER *can be approximated in polynomial time within a factor of* $\frac{35}{6}$*, if* d *satisfies the triangle inequality.*

Proof. Let $G = (V, E)$ be the input graph with induced distances d. First, compute an optimal solution $P_1, \dots, P_s$ for $(\|\cdot\|_1^w, \text{diam})$-2-CLUSTER with Proposition 6. This solution satisfies $|P_i| \le 3$ for all $i \in \{1, \dots, s\}$. Let D^* be the global cost of $P_1, \dots, P_s$. It follows that

$$D^* \le \text{opt}(G, d, \text{diam}, \|\cdot\|_1^w, 4)\,, \tag{8}$$

simply because any 4-cluster is also a 2-cluster.

Then, consider the complete graph $G' = (P, P \times P)$ with vertices $P = \{p_1, \dots, p_s\}$ (p_i represents the set P_i) and edge-weights w defined by $w_{i,j} := w(\{p_i, p_j\}) := \min\{d(u,v)\colon u \in P_i, v \in P_j\}$. Compute an optimal solution $S_1, \dots, S_r$ for $(\|\cdot\|_1^w, \text{avg})$-2-CLUSTER on G' with Theorem 4 such that with $|S_i| \le 3$ for all $i \in \{1, \dots, s\}$ by Proposition 3 . Lemma 26 then yields:

$$D^* \ge 3 \cdot \text{opt}(G', w, \text{avg}, \|\cdot\|_1^w, 2) = 3 \cdot \sum_{j=1}^{s} |S_j| \cdot \text{avg}(S_j)\,. \tag{9}$$

We interpret this partition $S_1, \dots, S_r$ as a partition $\mathfrak{S} = \{S'_1, \dots, S'_r\}$ on the graph G, i.e., $S'_j := \bigcup_{p_i \in S_j} P_i$ for all $j \in \{1, \dots, r\}$. As $|P_i|, |S_j| \geq 2$ for all $i \in \{1, \dots, r\}$ and $j \in \{1, \dots, s\}$ it follows that $|S'_j| \geq 4$ for all $j \in \{1, \dots, s\}$, so $S'_1, \dots, S'_r$ is a 4-cluster for G.

If $|S'_q| \geq 8$ then S_q contains three vertices, so let $i, j, k \in \{1, \dots, s\}$ be the indices such that $S_q = \{p_i, p_j, p_k\}$ with central vertex p_i and $|P_j| = 3$, we replace the cluster S'_q in $\mathfrak{S}$ by the two clusters $P' := P_j \cup \{u_i\}$ and $P'' := S'_q \setminus P'$ with $u_i \in P_i$ such that

$$w_{i,j} = \min\{d(u_i, v) \colon v \in P_j\}\,.$$

These new clusters satisfy:

$$|P'| \cdot \operatorname{diam}(P') \leq 4 \cdot (\operatorname{diam}(P_j) + w_{i,j}) < 2 \cdot |P_j| \cdot \operatorname{diam}(P_j) + 4 \cdot w_{i,j}$$

and

$$\begin{aligned} |P''| \cdot \operatorname{diam}(P'') &\leq 5 \cdot (\operatorname{diam}(P_i) + \operatorname{diam}(P_k) + w_{i,k}) \\ &\leq \tfrac{5}{2} \cdot |P_i| \cdot \operatorname{diam}(P_i) + \tfrac{5}{2} \cdot |P_k| \cdot \operatorname{diam}(P_k) + 5 \cdot w_{i,k}\,. \end{aligned}$$

Consider any set $R \in \mathfrak{S}$ which is not the result of splitting up a cluster S'_q.

- If $R = P_i \cup P_j$, we know that $\operatorname{diam}(R) \leq \operatorname{diam}(P_i) + \operatorname{diam}(P_j) + w_{i,j}$ and $|R| \leq 6$, hence:

$$|R| \cdot \operatorname{diam}(R) \leq 3 \cdot |P_i| \cdot \operatorname{diam}(P_i) + 3 \cdot |P_j| \cdot \operatorname{diam}(P_j) + 6 \cdot w_{i,j}\,. \tag{10}$$

- If $R = P_i \cup P_j \cup P_k$, with p_i as central vertex for $S_q = \{p_i, p_j, p_k\}$; we know that $|R| \leq 7$ (as P_j and P_k have cardinality 2) and

$$\operatorname{diam}(R) \leq \operatorname{diam}(P_i) + \operatorname{diam}(P_j) + \operatorname{diam}(P_k) + w_{i,j} + w_{i,k}\,,$$

hence $|R| \cdot \operatorname{diam}(R)$ is bounded by:

$$\begin{aligned} |R| \cdot \operatorname{diam}(R) &\leq 7 \cdot (\operatorname{diam}(P_i) + \operatorname{diam}(P_j) + \operatorname{diam}(P_k) + w_{i,j} + w_{i,k}) \\ &\leq \tfrac{7}{2} \sum_{h \in \{i,j,k\}} |P_h| \cdot \operatorname{diam}(P_h) \;+\; 7(w_{i,j} + w_{i,k})\,. \end{aligned} \tag{11}$$

Equations (9), (10) and (11) yield:

$$\begin{aligned} \sum_{R \in \mathfrak{S}} |R| \cdot \operatorname{diam}(R) &\leq \tfrac{7}{2} \cdot \sum_{i=1}^{r} |P_i| \cdot \operatorname{diam}(P_i) + 6 \cdot \sum_{R \subseteq P_i \cup P_j} w_{i,j} + 7 \cdot \sum_{R = P_i \cup P_j \cup P_k} (w_{i,j} + w_{i,k}) \\ &\leq \tfrac{7}{2} \, \| (\operatorname{diam}((P_1), \dots, \operatorname{diam}(P_s)) \|_1^w + 7 \cdot \sum_{i=1}^{q} |S_i| \cdot \operatorname{avg}(S_i) \\ &\leq \tfrac{7}{2} D^* + \tfrac{7}{3} D^* = \tfrac{35}{6} D^*\,. \end{aligned}$$

□

Remark 5*:* With Equation 1, the above algorithm yields a $\frac{35}{3}$-approximation for $(\|\cdot\|_1^w, \mathrm{rad})$-4-CLUSTER. Equation 1 translates the above result to yield a $\frac{35}{3}$-approximation for $(\|\cdot\|_1^w, \mathrm{rad})$-4-CLUSTER.

Since the approximation ratios from Theorem 27 are significantly better than the path-partitioning approximation from Proposition 22 (factor 24 and 48, respectively), it would be interesting to nest this construction further and extend it for larger values of k. In our experimental results, we tested the performance of multiple nested applications of edge cover and matching while in each step always picking the strategy which gave a better global cost with respect to diameter and weighted 1-norm. Although the results looked promising, proving an approximation ratio for this strategy appears to be a very difficult task.

4 Geometric Instances

In our abstract formulation of $(\|\cdot\|, f)$-k-CLUSTER one might be surprised that objects are plainly represented by a vertex without additional object-specific information. In many applications for this type of clustering, objects are usually vectors of attribute values, for example a personal record, which contains values for several attributes. Records representing information about δ different attributes such as *age* and *salary* etc. are usually represented by δ-dimensional vectors. For many problems, this dimensionality δ is seen as the main factor for computational hardness, a property often referred to as the *curse of dimensionality*, a term first introduced by Richard Bellman in [11]. This link between dimension and tractability occurs for many types of problems, for k-anonymity, for example, this effect is discussed in [3]. For an overview of the problems which arise for the task of clustering high dimensional data in general, see [60].

A more optimistic way to look at this connection between dimension and complexity are strategies which exploit the additional structure given by low dimensionality. The study of problems for which instances are in some sense embedded in the plane (2-dimensional Euclidean space) is a huge research area and the list of examples for which restriction to the plane improves computability seems endless. Most significant and relevant to our problems here is probably the improvement of running time of the 2-approximation algorithm we used for tree partitioning from [43], from $\mathcal{O}(n^2)$ for general instances to $\mathcal{O}(n\log(n))$ for instances for which vertices correspond to points in the plane.

So far, our model captures this complexity of dimension with the structure of the distance d for which we have already noticed that requiring triangle inequality has a huge impact on structural properties and especially on approximability. In this section we want to study if an even further restriction to d being the Euclidean distance allows for a helpful parameterisation by the dimension. We proceed by first clearly defining these types of instances for $(\|\cdot\|, f)$-k-CLUSTER and the notion *dimension*. We will then see that this dimension is not the main factor of computational hardness for most versions of our family of problems.

4.1 Definition of EUCLIDEAN $(\|\cdot\|, f)$-CLUSTER

In order to properly discuss the term *dimension* for our family of clustering problems, we will define a geometric version of $(\|\cdot\|, f)$-k-CLUSTER which we will denote by EUCLIDEAN $(\|\cdot\|, f)$-k-CLUSTER, keeping the three choices for each f and $\|\cdot\|$. Instances of EUCLIDEAN $(\|\cdot\|, f)$-k-CLUSTER are given by vectors $\mathbb{R}^\delta$ which we will often simply call *points*, each point representing a vertex in the original graph definition. The term *k-cluster* translates for a finite set of points $P \subseteq \mathbb{R}^\delta$ analogously to a partition $P_1, \dots, P_s$ of P such that $|P_i| \geq k$ for all $i \in \{1, \dots, s\}$.

The distance between two vertices u and v is given by the *Euclidean distance* between the points representing u and v. Formally, for two vertices u, v given by coordinates $(u_1, \ldots, u_\delta)$ and $(v_1, \ldots, v_\delta)$, respectively, where $\delta \in \mathbb{N}$ and $u_i, v_i \in \mathbb{R}$ for each $i \in \{1, \ldots, \delta\}$, we define:

$$d_\delta(u, v) := \left(\sum_{i=1}^{\delta} (u_i - v_i)^2 \right)^{\frac{1}{2}} .$$

Whenever we speak about *distance* between points or vertices in the following, we refer to this distance with δ fixed according to the space $\mathbb{R}^\delta$ containing the representing points in question. The formal decision problem which we will discuss in this section is defined as follows:

Euclidean $(\|\cdot\|, f)$-k-cluster

Input: $P \subset \mathbb{R}^\delta$ finite, $k \in \mathbb{N}$, $D \in \mathbb{R}$.

Question: Is there a k-cluster $P_1, \ldots, P_s$ of P for some $s \in \mathbb{N}$, such that $\| (f(P_1), \ldots, f(P_s)) \| \leq D$, where the pairwise distances to compute this objective function is computed by d_δ.

4.2 Reduction Idea

In this section, we will show that Euclidean $(\|\cdot\|, f)$-k-cluster for local cost $f \in \{\text{diam}, \text{rad}\}$ and all choices for $\|\cdot\|$ turns out to be NP-complete already for very low dimensions, i.e., constant δ. In particular, we will show that Euclidean $(\|\cdot\|, f)$-k-cluster remains NP-hard even when restricted to instances in the 3-dimensional Euclidean space ($\delta = 3$) for both $f \in \{\text{diam}, \text{rad}\}$, all norms and k fixed to a small constant. For weighted 1-norm we show this hardness for the even stronger restriction to points in the plane ($\delta = 2$), again for both $f \in \{\text{diam}, \text{rad}\}$ and k fixed to a small constant. A discussion about $f = \text{avg}$ can be found at the end of this section.

This hardness even for low fixed dimension is not too surprising as the related problems k-Means, k-Median and k-Center have all been proven NP-hard even when restricted to instances in the plane (see [48] for k-Means and [50] for k-Median and k-Center). The problem closest to any of our versions of Euclidean $(\|\cdot\|, f)$-k-cluster in this context is the version of the k-Median problem as introduced by *Papadimitriou*, with the difference to the previously mentioned problems being that the "medians" have to be chosen among the given set of points. This version of k-Median also remains NP-hard when restricted to planar instances by a construction given in [53]. We will in fact in the following use constructions with a similar conceptual idea of reducing from a version of Exact-3-Cover. Recall the definition of this problem from Section 2.3.1, with t fixed to 3, which is NP-hard by [33].

Exact-3-Cover

Input: A universe $X = \{x_1, \ldots, x_n\}$ and a collection $C = \{S_1, \ldots, S_r\}$ of subsets of X, such that each S_i, $i \in \{1, \ldots, r\}$, has cardinality 3.

Question: Does there exist a subset $C' \subseteq C$ that is a partition of X?

The main difference is that our constructions use larger distances to give nice concrete coordinates to cleanly show how the reductions can be computed in polynomial time. Further we make changes to the original set of points according to each choices of global and local cost, mostly duplicating and slightly moving some points, but the overall idea remains the same as in the reduction used for k-Median and can intuitively be described as follows:

For an instance $C = \{S_1, \ldots, S_r\}$ over a universe X of Exact-3-Cover, we introduce for each set S_i a group of points which can be partitioned with small cost and according to the required minimum cardinality in exactly two different ways, which correspond to the decision whether or not S_i is chosen to be part of the cover. For each element in the universe X, there exists a corresponding set in such a, say feasible, partition of the points introduced for S_i. The group of points for the set S_i is, very informally speaking, arranged to lie on a horizontal line and starts (seen from small to larger x-coordinate) with some initial points, followed by the points which correspond to x_1, followed by the points for x_2 and so forth. The groups for $S_1, \ldots, S_r$ are arranged as parallel lines in this order which yields a horizontal separation of the space in the areas between the points for S_i and S_{i+1} for $i \in \{1, \ldots, r-1\}$. Points which mark whether an element x_j is contained in a set S_i are placed in the areas between S_{i-1} and S_i (and S_i and S_{i+1}), more precisely, we introduce a point $x_{i,j}$ if $x_j \in S_i$ and a point $y_{i,j}$ otherwise (also $x'_{i,j}$ or $y'_{i,j}$ between S_i and S_{i+1}). These points are arranged in such a way that $x_{i,j}$ (or $x'_{i,j}$) can only be included in a cluster of small cost and required cardinality, if the points for S_i are partitioned to reflect that S_i is chosen not to be in the cover, which may be confusing at first. In a sense, the reduction will use the exactness of the solution for Exact-3-Cover, and not the covering property to link k-clusters of small cost to the existence of an exact cover.

Denote by $(x_{i,j}/y_{i,j})$ the point $x_{i,j}$ or $y_{i,j}$, depending on which one exists according to the containment of x_j in S_i. The construction contains other points (named $q_{i,j}$ and $q'_{i-1,j}$) between S_{i-1} and S_i to enable a feasible clustering for $(x_{i,j}/y_{i,j})$ or $(x'_{i-1,j}/y'_{i-1,j})$ but not both, which means that one of these has to be in a cluster with its corresponding set in S_i and S_{i-1}, respectively. This way, with r horizontal point sets corresponding to $S_1, \ldots, S_r$ and $2(r-1)$ points of the type $(x_{i,j}/y_{i,j})$ in the areas between them from which only $r-1$ can be clustered with $q_{i,j}$ and $q'_{i-1,j}$, $r-1$ remain to be clustered with the points for their corresponding set S_i. Hence, for each j at most one set S_i can be chosen to have its points arranged in the way which signals containment in the cover.

In the following we will use the names already introduced in the sketch above and also use the expression $(x_{i,j}/y_{i,j})$ to denote $x_{i,j}$ or $y_{i,j}$, depending on which of these two points exists for indices i and j.

4.3 Diameter

Theorem 28

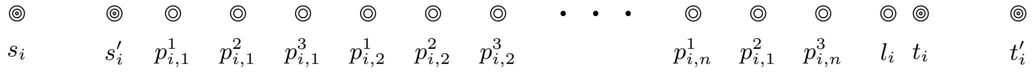
Euclidean $(\|\cdot\|_1^w, \mathrm{diam})$-$k$-cluster *with* $k = 6$ *and* $\delta = 2$ *is* NP*-hard.*

Proof. We reduce from Exact-3-Cover, with an instance given by a collection of sets $S_1, \ldots, S_r$ of cardinality 3 over the universe $\{x_1, \ldots, x_n\}$. We will define the instance for Euclidean $(\|\cdot\|_1^w, \mathrm{diam})$-$k$-cluster by giving points in the plane, each point corresponds to a vertex and the distance between two vertices is the Euclidean distance between the points. Whenever we speak about distance in the following, we refer to the Euclidean distance.

Let $\lambda \geq 0$ be a constant; we will later fix this value accordingly but the description of the reduction is more readable with the use of the substitute λ. We create the following points at y-coordinate $(i-1)6.64$ for every $i \in \{1, \ldots, r\}$ and $j \in \{1, \ldots, n\}$ (the row labelled "Count" denotes the number of points we create of each respective coordinates, for simplicity, we abuse notation and use the same name for all copies as we always refer to the sets):

Name	s_i	s'_i	$p^1_{i,j}$	$p^2_{i,j}$	$p^3_{i,j}$	l_i	t_i	t'_i
x-Coord	0	1.9	$\lambda{+}3j{-}1$	$\lambda{+}3j$	$\lambda{+}3j{+}1$	$2{+}\lambda{+}3n$	$2.1{+}\lambda{+}3n$	$4{+}\lambda{+}3n$
Count	3	3	2	2	2	2	3	3

This gives the following row of points for each set S_i:



For all (i,j) with $i \in \{1, \ldots, r\}$ and $j \in \{1, \ldots, n\}$ such that $x_j \in S_i$, create the following points at x-coordinate $1 + \lambda + 3j$ (except $x_{r,j}$, $q_{r,j}$ and $x'_{1,j}$, $q'_{1,j}$):

Name	$x_{i,j}$	$x'_{i,j}$	$q_{i,j}$	$q'_{i,j}$
y-Coord	$(i{-}1)6.64{+}1.73$	$(i{-}1)6.64{-}1.73$	$(i{-}1)6.64{+}3.73$	$(i{-}1)6.64{-}3.73$
Count	1	1	2	3

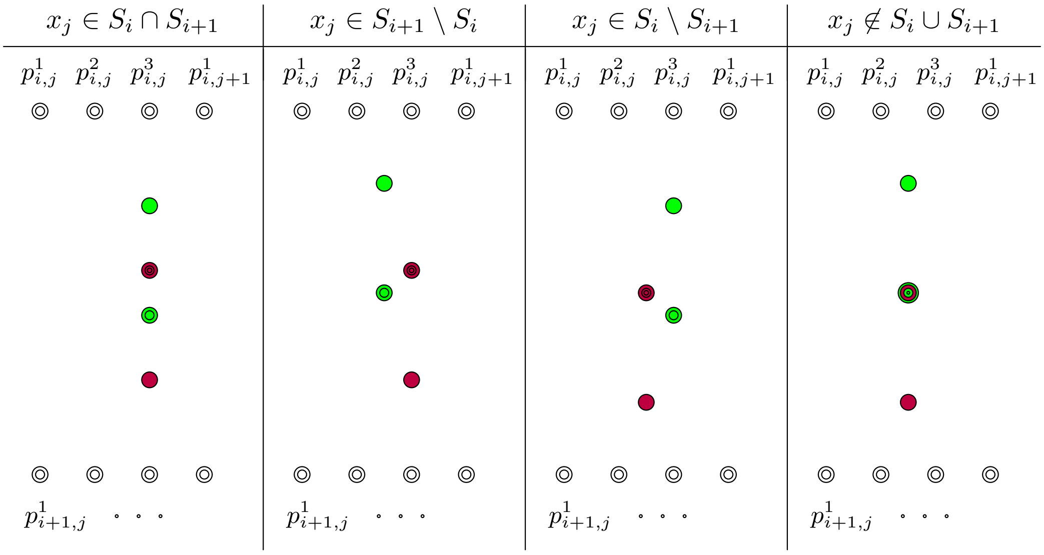
This arrangement places $x_{i,j}$ at a distance of approximately 2 (a little bit less than 2) from $p^2_{i,j}$ and $p^1_{i,j+1}$ (and, obviously, at distance exactly 2 from $q_{i,j}$). Symmetrically for $x'_{i,j}$ with $p^2_{i+1,j}$ and $p^1_{i+1,j+1}$.

For all (i,j) with $i \in \{1, \ldots, r\}$ and $j \in \{1, \ldots, n\}$ such that $x_j \notin S_i$, create the following points (except for $y_{r,j}$, $q_{r,j}$ and $y'_{1,j}$, $q'_{1,j}$) at x-coordinate $0.5 + \lambda + 3j$:

Name	$y_{i,j}$	$y'_{i,j}$	$q_{i,j}$	$q'_{i,j}$
y-Coord	$(i-1)6.64+1.32$	$(i-1)6.64-1.32$	$(i-1)6.64+3.32$	$(i-1)6.64-3.32$
Count	1	1	2	3

This arrangement places $y_{i,j}$ at a distance of approximately 2 (again, a bit less than 2) from $p^1_{i,j}$ and $p^1_{i,j+1}$. Symmetrically for $y'_{i,j}$ with $p^1_{i+1,j}$ and $p^1_{i+1,j+1}$.

Points $x_{i,j}, y_{i,j}, q_{i,j}$ (green) and $x'_{i+1,j}, y'_{i+1,j}, q'_{i+1,j}$ (purple) lie between the rows of points for S_i and S_{i+1} in one of the following ways:


$x_j \in S_i \cap S_{i+1}$
$x_j \in S_{i+1} \setminus S_i$
$x_j \in S_i \setminus S_{i+1}$
$x_j \notin S_i \cup S_{i+1}$
$p^1_{i,j}$ $p^2_{i,j}$ $p^3_{i,j}$ $p^1_{i,j+1}$
$p^1_{i+1,j}$ . . .

Denote by P the set of all points introduced by the above construction. With P as vertices with Euclidean distance as instance of $(\|\cdot\|_1^w, \mathrm{diam})$-$k$-CLUSTER with $k = 6$, we claim that there exists a 6-cluster of global cost $\mathcal{C}$ if and only if $S_1, \ldots, S_r$ is a "yes" instance for EXACT-3-COVER, where the cost $\mathcal{C}$ is given by:

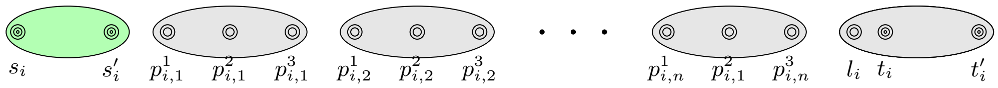
$$\mathcal{C} := 30rn + 27.4r - 14n + 8(r - \tfrac{n}{3})\lambda\,.$$

First, observe the following properties of partitions for the points created by the reduction:

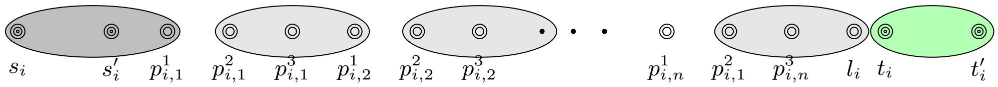
Observation 1*:* The minimum diameter of a set in a 6-cluster for P which contains a point $p \notin \{s_i, s'_i, t_i, t'_i \colon 1 \leq i \leq r\}$ is 2.

For each $i \in \{1 \ldots, r\}$ we will specifically name two options to partition the points in $P_i := \{p^h_{i,j} \colon 1 \leq j \leq n, 1 \leq h \leq 3\} \cup \{s_i, s'_i, t_i, t'_i\}$, as these will be the best options in an optimal solution and distinguishing between them is the relation to the exact cover solution:

- The sets $\{s_i, s'_i\}$ (diameter 1.9), $\{l_i, t_i, t'_i\}$ (diameter 2) and $\{p^1_{i,j}, p^2_{i,j}, p^3_{i,j}\}$ (diameter 2) for $j \in \{1, \ldots, n\}$. In the following we will call this partitioning *cluster-scheme 1*:

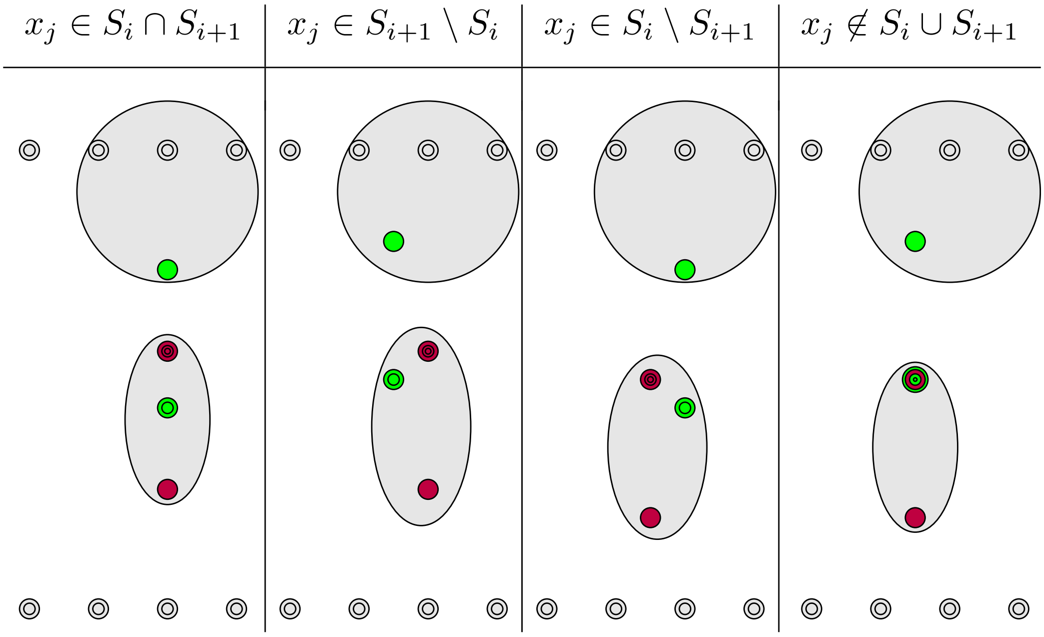


- The sets $\{s_i, s_i', p_{i,1}^1\}$ (diameter $2+\lambda$), $\{t_i, t_i'\}$ (diameter 1.9), $\{p_{i,n}^2, p_{i,n}^3, l_i\}$ (diameter 2) and $\{p_{i,j}^2, p_{i,j}^3, p_{i,j+1}^1\}$ (diameter 2) for $j \in \{1, \ldots, n-1\}$. This partition will in the following be referred to as *cluster-scheme 2*. Compared to cluster-scheme 1, this partition increases the global cost by 8λ and is, in a sense, shifted, see illustration below:



Other partitions for the points created for S_i can only be extended to a 6-cluster which contains at least one cluster of diameter more than 2.64 and, as we will see in the following when λ is fixed, yield a solution of global cost larger than C.

Using cluster-scheme 2 for the points in P_i (or symmetrically P_{i+1}), allows assigning all points that lie between the points in P_i and P_{i+1} to a cluster of diameter 2. Observe that $q_{i,j}, q_{i+1,j}'$ with *one* of the points $x_{i,j}, y_{i,j}, x_{i+1,j}', y_{i+1}'$ builds a cluster of diameter 2 in all cases:

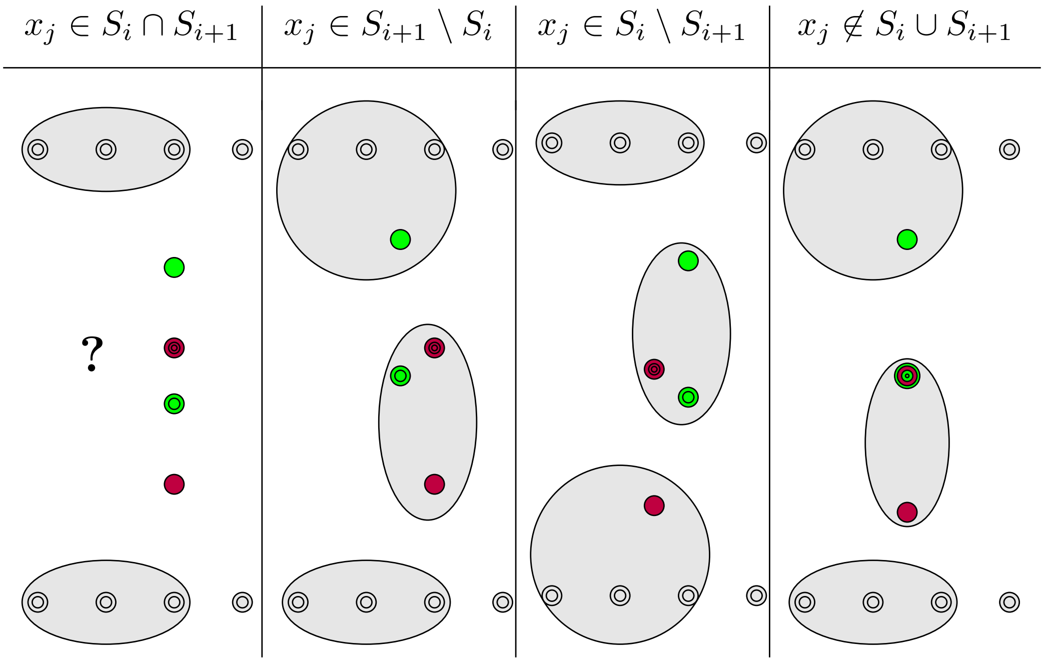

$x_j \in S_i \cap S_{i+1}$

$x_j \in S_{i+1} \setminus S_i$

$x_j \in S_i \setminus S_{i+1}$

$x_j \notin S_i \cup S_{i+1}$

If cluster-scheme 1 is used for both P_i and P_{i+1}, the case $x_j \in S_i \cap S_{i+1}$ is problematic. Cases $x_j \in S_{i+1} \setminus S_i$ and $x_j \in S_i \setminus S_{i+1}$ only leave one option to build clusters of cardinality at least 6 and diameter 2:


$x_j \in S_i \cap S_{i+1}$
$x_j \in S_{i+1} \setminus S_i$
$x_j \in S_i \setminus S_{i+1}$
$x_j \notin S_i \cup S_{i+1}$
?

With a distance of 3.18 between $x_{i,j}$ and $x'_{i+1,j}$, the cheapest clustering option (w.r.t. diameter and 1-norm) in the first case is to either build a cluster containing $x_{i,j}$ and $p^1_{i,j}, p^2_{i,j}, p^3_{i,j}$ or, symmetrically, a cluster containing $x'_{i+1,j}$ and $p^1_{i+1,j}, p^2_{i+1,j}, p^3_{i+1,j}$, which both give a cluster of diameter larger than 2.64.

Regardless of the structure of the sets S_i and the corresponding points $x_{i,j}, y_{i,j}$ and $q_{i,j}$, the following partition of P always gives a feasible 6-cluster:

Sets	Indices	Diameter	Cardinality
$\{s_i, s'_i, p^1_{i,1}\}$	$1 \leq i < r$	$2+\lambda$	8
$\{p^2_{i,j}, p^3_{i,j}, p^1_{i,j+1}, (x_{i,j} \setminus y_{i,j})\}$	$1 \leq i < r,\ \ 1 \leq j < n$	2	7
$\{p^2_{i,n}, p^3_{i,n}, l_i, (x_{i,n} \setminus y_{i,n})\}$	$1 \leq i < r$	2	7
$\{t_i, t'_i\}$	$1 \leq i < r$	1.9	6
$\{q_{i-1,j}, q'_{i,j}, (x'_{i,j} \setminus y'_{i,j})\}$	$1 < i < r,\ \ 1 \leq j \leq n$	2	6
$\{s_r, s'_r\}$		1.9	6
$\{p^1_{r,j}, p^2_{r,j}, p^3_{r,j}\}$	$1 \leq j \leq n$	2	6
$\{t_r, t'_r, l_r\}$		2	8

See Figure 5 for an illustration of this partition, which we will call *basic partition* in the following. As a solution for $(\|\cdot\|_1^w, \mathrm{diam})$-$k$-CLUSTER, the basic partition has a global cost of:

$$\mathcal{C}' := 2 \cdot |P| - 0.6r + 8(r-1)\lambda = 30rn + 27.4r - 14n + 8(r-1)\lambda\,.$$

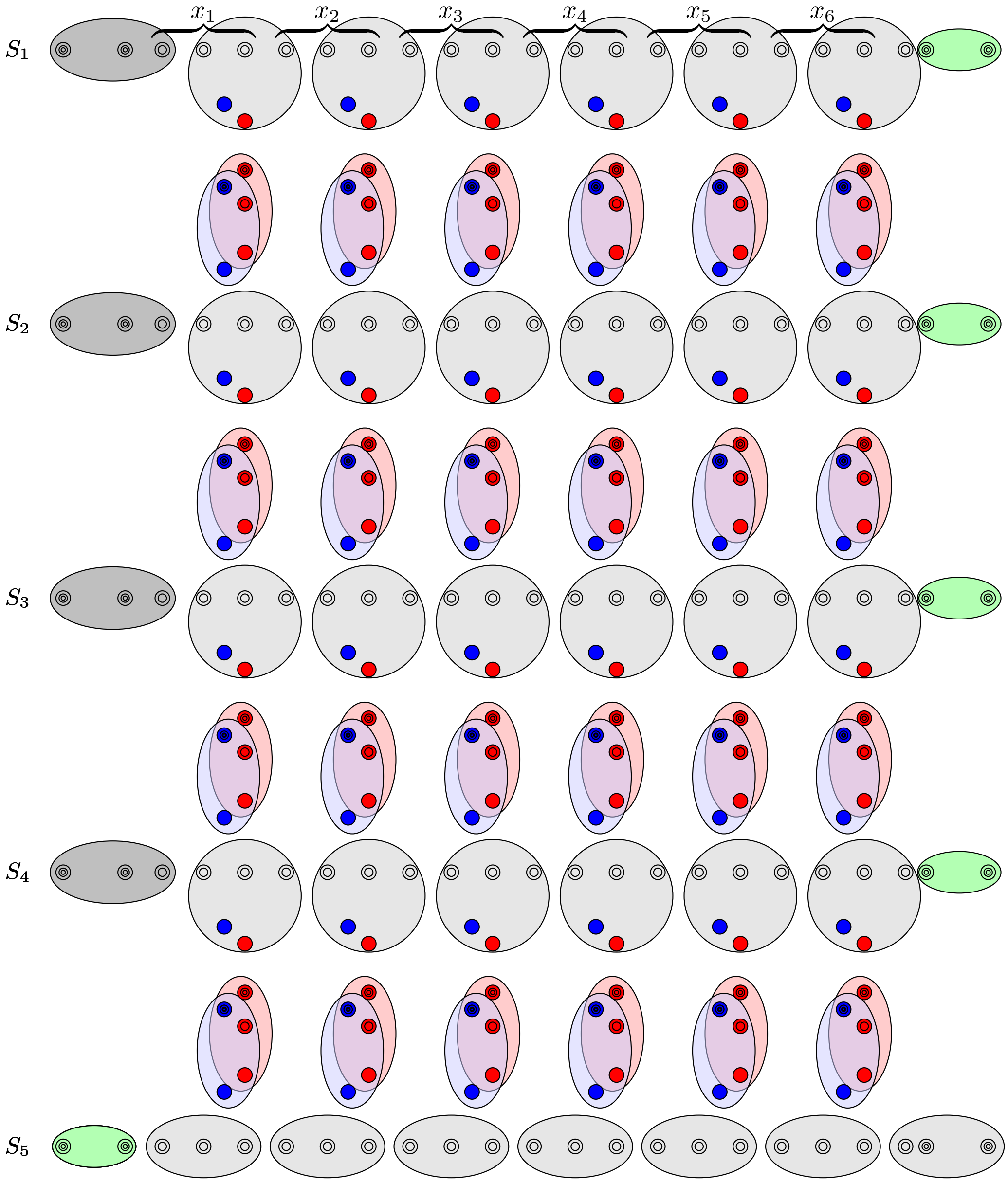


Figure 5: Basic partition for $r = 5, n = 6$, with $x_{i,j}, x'_{i,j}$ and corresponding $q_{i,j}, q'_{i,j}$ colored blue and $y_{i,j}, y'_{i,j}$ and corresponding $q_{i,j}, q'_{i,j}$ colored red. Clusters displayed in dark grey have diameter $2 + \lambda$, green clusters have a diameter of 1.9 and all others have a diameter of 2.

Now, fix $\lambda = 0.25$. Except for the points $s_i, s'_i, p^1_{i,1}$ $i \in \{1, \dots, r-1\}$ and t_n, t'_n, the cost produced by each point is minimal by Observation 1. Further, using $(x_{i,j}, y_{i,j})$, possibilities other than cluster-scheme 1 or 2 for $p_{i,j}$ to improve this standard solution for $s_i, s'_i, p^1_{i,1}$, $i \in \{1, \dots, r-1\}$ or t_n, t'_n are worse:

- Any cluster which includes both $y_{i,j}$ and $y'_{i,j}$ together has a diameter of at least 2.64 and even larger for $x_{i,j}$ and $x'_{i,j}$. Even if this strategy allows to cluster both s_i, s'_i and t_i, t'_i at their best diameter possible, the global cost for any 6-cluster which uses this option is larger than the cost for the basic partition as $6 \cdot 0.64 > 6 \cdot 0.1 + 8 \cdot \lambda$.

- Clustering s_i, s'_i and t_n, t'_n without any other points by building one cluster of diameter 3 within the points $p^1_{i,1}, \dots, p^3_{i,n}, l_i$ increases the global cost by at least $6 - 8(0.1 + \lambda) > 0$ compared to the basic partition.

The only option for points $q_{i-1,j}, q'_{i,j}$ at diameter 2 is in a cluster containing either $(x_{i-1,j}/y_{i-1,j})$ or $(x'_{i,j}/y'_{i,j})$, but not both, as this gives a cluster of diameter at least $d_2(x_{i-1,j}, x'_{i,j}) = 3.18$. A cluster containing both $q_{i,j}$ and $q_{i,j+1}$ or $q'_{i,j+1}$ has a diameter of 2.5, so an overall increase for the global cost of at least 3 compared to the basic partition. Clusters including $q_{i-1,j}, q'_{i,j}$ for some i will be called q_j-sets in the following, observe that there are exactly $r-1$ q_j-sets for each $j \in \{1, \dots, n\}$.

The only possibilities to improve on the basic partition is using cluster-scheme 1 for more than one of the sets P_i. If cluster-scheme 1 is used for P_i, the points $x_{i,j}$ and $x'_{i,j}$ with j such that $x_j \in S_i$ have to be in a cluster together with $\{q_{i,j}, q'_{i+1,j}\}$ and $\{q_{i-1,j}, q'_{i,j}\}$, respectively; otherwise the global cost increases by more than the 8λ saved by switching P_i to cluster-scheme 1. A partition which uses cluster-scheme 1 for all P_{i_w} with $w \in \{1 \dots, s\}$ is only better than a clustering which uses cluster-scheme 1 for all P_{i_w} with w in a strict subset of $\{1, \dots, s\}$, if it is possible to assign all existing points in $\{x_{i_w,j}, x'_{i_w,j}, y_{i_w,j}, y'_{i_w,j} \colon 1 \leq i \leq r,\ 1 \leq w \leq s\}$ at diameter 2.

With these observations on the basic partition and the possible improvements, we will show the following claim, which gives an idea of how the clustering corresponds to an exact cover:
Claim 1: A minimum 6-cluster for P uses cluster-scheme 1 for the sets P_{i_w} with $w \in \{1, \dots, s\}$ if and only if $|\bigcup_{l=1}^{s} S_{i_l}| = 3s$. Further, the global cost w.r.t. diameter and 1-norm for this 6-cluster is $\mathcal{C}' - 8s\lambda$.

Proof of Claim: Assume that cluster-scheme 1 is used for P_a and P_b, $a \neq b$, and there exists a $j \in \{1, \dots, n\}$ such that $x_j \in S_a \cap S_b$. There is no option to build clusters for all existing points in $\{x_{w,j}, x'_{w,j}, y_{w,j}, y'_{w,j} \colon 1 \leq i \leq r, 1 \leq w \leq n\}$ at diameter 2, simply because there are not enough q_j-sets. For every index $i \in \{2, \dots, r-1\}$, at least one of the points out of $\{(x_{i,j}/y_{i,j}), (x'_{i,j}/y'_{i,j})\}$ has to be contained in a distinct q_j-set. For P_a and P_b partitioned by cluster-scheme 1, the existing ($x_{h,j}$ only exists for $h < r$ and $x'_{h,j}$ only exists for $h > 1$) points out of $\{x_{a,j}, x'_{a,j}, x_{b,j}, x'_{b,j}\}$ all have to be included in a distinct q_j-set. This however requires $|\{1,\dots,r\}| - |\{1,\dots,r\} \cap \{a,b\}| + 2|\{a,b\}| - |\{a,b\} \cap \{1,r\}| = r$ such q_j-sets, which contradicts the fact that there can only be $r-1$.

On the other hand, if $|\bigcup\limits_{l=1}^{s} S_{i_l}| = ts$, it follows that for every $j \in \{1,\dots,n\}$ there is at most one index $l \in \{1,\dots,s\}$ such that $x_j \in S_{i_l}$. The following changes to the basic partition improve the global cost by 8λ compared to using cluster-scheme 2 for P_{i_l}:

- For all all j with $x_j \in S_{i_l}$, build the set $\{q_{i-1,j}, q'_{i,j}, (x'_{i,j}/y'_{i,j})\}$ for all indices i with $i \leq i_l$ and the set $\{q_{i,j}, q'_{i+1,j}, (x_{i,j}/y_{i,j})\}$ for all indices i with $i \geq i_l$.
- For all all j with $x_j \in S_{i_l}$, build the set $\{p^1_{i_l,j}, p^2_{i_l,j}, p^3_{i_l,j}\}$ and for all j with $x_j \notin S_{i_l}$ the set $\{p^1_{i_l,j}, p^2_{i_l,j}, p^3_{i_l,j}, y_{i_l,j}\}$.
- Build the sets $\{s_{i_l}, s'_{i_l}\}$ and $\{l_{i_l}, t_{i_l}, t'_{i_l}\}$.
- For all all j with $x_j \in S_{i_l}$, build the set $\{p^2_{i,j}, p^3_{i,j}, p^1_{i,j+1}, (x_{i,j}/y_{i,j})\}$ for each index i with $i < i_l$ and the set $\{p^2_{i,j}, p^3_{i,j}, p^1_{i,j+1}, (x'_{i,j}/y'_{i,j})\}$ for each index i with $i > i_l$.

As these changes only affect points with j-index such that $x_j \in S_{i_l}$ and the points $s_{i_l}, s'_{i_l}, l_{i_l}, t_{i_l}, t'_{i_l}$, we can apply these changes for each $l \in \{1,\dots,s\}$ without conflict as the sets S_{i_l} are disjoint and arrive at a 6-cluster with global cost $\mathcal{C}' - 8s\lambda$.

By the properties of the underlying problem Exact-3-Cover and Claim 1, it is clear that cluster-scheme 1 can be used for at most $\frac{n}{3}$ different sets P_{i_w} which means that $\operatorname{opt}(P, d_2, \|\cdot\|_1^w, \operatorname{diam}, 6) \geq \mathcal{C}' - 8\frac{n}{3}\lambda = \mathcal{C}$, for the instance (P, d_2) of $(\|\cdot\|_1^w, \operatorname{diam})$-$k$-cluster constructed for any given instance of Exact-3-Cover with r sets over a universe of size n. So if there exists a 6-cluster $\mathfrak{P}$ for P with global cost at most $\mathcal{C}$, $\mathfrak{P}$ is optimal with respect to diameter and 1-norm, cluster-scheme 1 is used optimally for exactly $\frac{n}{3}$ point sets P_{i_w}, $i_w \in \{1,\dots,r\}$. By Claim 1, this means that for the $\frac{n}{3}$ indices $i_1,\dots,i_{\frac{n}{3}}$ for which P_{i_w} is partitioned by cluster-scheme 1 in $\mathfrak{P}$ satisfy $|\bigcup\limits_{l=1}^{n/t} S_{i_l}| = t\frac{n}{3} = n$, so $S_{i_1},\dots,S_{i_{\frac{n}{3}}}$ is an exact cover for $\{x_1,\dots,x_n\}$.

Conversely, if $S_{i_1},\dots,S_{i_{\frac{n}{3}}}$ is an exact cover for $\{x_1,\dots,n\}$, then the basic partition for P can be improved by 8λ for each index $i_\ell \in \{i_1,\dots,i_{\frac{n}{3}}\}$ with the construction given in Claim 1. This gives a 6-cluster of global cost $\mathcal{C}$ with respect to diameter and 1-norm for (P, d_2).

All in all, $S_1,\dots,S_r$ is a "yes"-instance for Exact-3-Cover if and only if there exists a solution for the corresponding instance (P, d_2) of $(\|\cdot\|_1^w, \operatorname{diam})$-$k$-cluster with $k = 6$ of global cost $\mathcal{C}$. □

With points in three-dimensional space, this construction can be extended to a reduction to the problem variant with diameter and ∞-norm:

Theorem 29

Euclidean $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-cluster *with* $k = 4$ *and* $\delta = 3$ *is* NP*-hard.*

Proof. We reduce from Exact-3-Cover, with an instance given by a collection of sets $S_1, \ldots, S_r$ of cardinality 3 over a universe $\{x_1, \ldots, x_n\}$ and start with the construction for Theorem 28 with all counts larger than 1 reduced by 1, $\lambda = 0$ and at z-coordinate 0. Replace s_i, s'_i, t_i, t'_i for each $i \in \{1, \ldots, r\}$ by:

Name	Count	x-Coord	y-Coord	z-Coord
b_i	4	1	$(i-1)6.64$	0
e_i	4	$5 + 3n$	$(i-1)6.64$	0

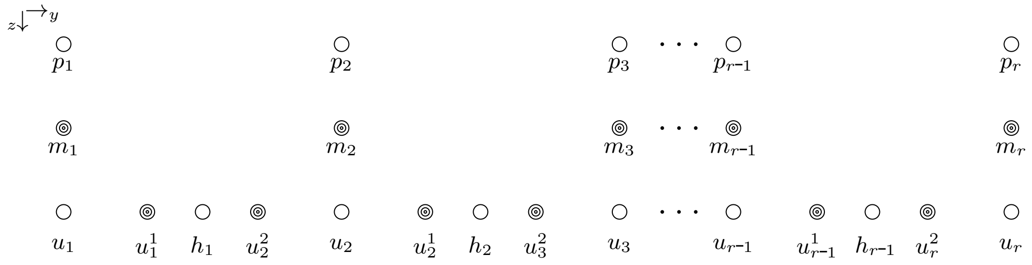
For all $i \in \{1, \ldots, r\}$ and $j \in \{1, \ldots, n\}$, add the following points (except $u^1_{r,j}, u^2_{1,j}, h_{r,j}$):

Name	Count	x-Coord	y-Coord	z-Coord
$u_{i,j}$	1	$3j + 1$	$(i-1)6.64$	5.73
$u^1_{i,j}$	3	$3j + 1$	$(i-1)6.64 + 2$	5.73
$u^2_{i,j}$	3	$3j + 1$	$(i-1)6.64 - 2$	5.73
$h_{i,j}$	1	$3j + 1$	$(i-1)6.64 + 3.32$	5.73

For pairs (i, j) with $x_j \in S_i$ further include the points:

Name	Count	x-Coord	y-Coord	z-Coord
$p_{i,j}$	1	$3j + 1$	$(i-1)6.64$	1.73
$m_{i,j}$	3	$3j + 1$	$(i-1)6.64$	3.73

For each $j \in \{1, \ldots, n\}$, these additional points build the following group at x-coordinate $3j + 1$ (where only the p_i and m_i with $x_i \in S_j$ exists):



Denote again by P the set of all points created by this construction. We claim that there exists a subset of $\{S_1,\dots,S_r\}$ which exactly covers $\{x_1,\dots,x_n\}$ if and only if there exists a 4-cluster for P of maximum diameter 2.

Assume there exists a 4-cluster $\mathfrak{P}$ of maximum diameter 2 for P. By the difference in the z-coordinate, no existing vertex from the set

$$\mathcal{Z} := \{p^h_{i,j}, x_{i,j}, x'_{i,j}, y_{i,j}, y'_{i,j}, q_{i,j}, q'_{i,j}, b_i, e_i \colon 1 \le i \le r, 1 \le j \le n, 1 \le h \le 3\}$$

is in the same cluster in $\mathfrak{P}$ as an existing vertex from

$$\mathcal{H} := \{u_{i,j}, u^1_{i,j}, u^2_{i,j}, h_{i,j}, m_{i,j} \colon 1 \le i \le r, 1 \le j \le n\}.$$

Only existing vertices from $\mathcal{U} := \{p_{i,j} \colon 1 \le i \le r, 1 \le j \le n\}$ can be in a set of diameter 2 with vertices from exclusively either $\mathcal{Z}$ or $\mathcal{H}$. As $d_2(p_{i,j}, (x_{i,j}/y_{i,j})) > 2.64$, for all $i \in \{1,\dots,r\}$ and $j \in \{1,\dots,n\}$, the vertices in $\mathcal{Z}$ at distance at most 2 from $p_{i,j}$ are $p^1_{i,j}, p^2_{i,j}, p^3_{i,j}$. So, if there is a set $S \in \mathfrak{P}$ with $S \cap \mathcal{Z} \neq \emptyset$ and $S \setminus \mathcal{Z} \neq \emptyset$, then $S = \{p_{i,j}, p^1_{i,j}, p^2_{i,j}, p^3_{i,j}\}$ for some $i \in \{1,\dots,r\}$ and $j \in \{1,\dots,n\}$. This means that the options to partition the vertices in $\mathcal{Z}$ into sets of minimum cardinality 4 and diameter at most 2 are very similar to the ones for Theorem 28, more precisely: For the points $p^h_{i,j}$, $i \in \{1,\dots,r\}$, $j \in \{1,\dots,n\}$ and $h \in \{1,2,3\}$, the only options to build sets of cardinality at least 4 and diameter at most 2 are, in a way, adjusted versions of cluster-scheme 1 and cluster-scheme 2 (observe that with the reduced cardinalities, some of the sets in the original cluster-schemes only have cardinality 3), with the following differences:

- The sets $\{s_i, s'_i\}$ and $\{t_i, t'_i\}$ are replaced by $\{b_i\}$ and $\{e_i\}$, respectively.
- For each $i \in \{1,\dots,r\}$ where cluster-scheme 1 is used for the points in P_i, the points $p^1_{i,j}, p^2_{i,j}, p^3_{i,j}$ build a set together with $p_{i,j}$, for all j with $x_j \in S_i$. For all j with $x_j \notin S_i$ the points $p^1_{i,j}, p^2_{i,j}, p^3_{i,j}$ build a set together with exclusively either $y_{i,j}$ or $y'_{i,j}$.
- For each $i \in \{1,\dots,r\}$ where cluster-scheme 2 is used for the points in P_i, the points $p^1_{i,j}, p^2_{i,j}, p^3_{i,j}$ build a set together with exactly one of the vertices $x_{i,j}$, $x'_{i,j}$, $y_{i,j}$, $y'_{i,j}$ for each $j \in \{1,\dots,n\}$ (this property also holds for the construction in Theorem 28).

For every $j \in \{1,\dots,n\}$, any attempt to partition a largest subset of the points in the set $\mathcal{H}_j := \{p_{i,j}, m_{i,j}, u_{i,j}, u^2_{i,j}, u^1_{i,j}, h_{i,j} \colon 1 \le i \le r\}$ into sets of minimum cardinality 4 and maximum diameter 2 excludes exactly one point from $\{p_{i,j}, u_{i,j}, h_{i,j}\}$ for some $i \in \{1,\dots,r\}$. This follows immediately from the given structure, as, for every point in $\mathcal{H}_j$, there are always at most two types of points at distance at most 2 to build a cluster of cardinalty at least 4. The points $u_{i,j}$ and $h_{i,j}$ have a distance of at least 5.73 from all points in $P \setminus \mathcal{H}_j$, so $\mathcal{H}_j$ is partitioned in $\mathfrak{P}$ such that for exactly one index $i \in \{1, \ldots, r\}$, the point $p_{i,j}$ is in a cluster only with points in $P \setminus \mathcal{H}_j$ for each $j \in \{1, \ldots, n\}$.

The only option to build a cluster of maximum diameter 2 and cardinality at least 4 containing $p_{i,j}$ but not $m_{i,j}$ is exactly the set $\{p_{i,j}, p^1_{i,j}, p^2_{i,j}, p^3_{i,j}\}$, as all points in $\mathcal{H}$ other than $m_{i,j}$ have distance at least 4 from $p_{i,j}$ and the only option with points from $\mathcal{Z}$ is $\{p_{i,j}, p^1_{i,j}, p^2_{i,j}, p^3_{i,j}\}$, as already shown above. If this set is included in $\mathfrak{P}$, then P_i is partitioned by the adjusted cluster-scheme 1 in $\mathfrak{P}$, which requires $\{p_{i,j'}, p^1_{i,j'}, p^2_{i,j'}, p^3_{i,j'}\} \in \mathfrak{P}$ for all $j' \in \{1, \ldots, n\}$ with $x_{j'} \in S_i$. As for each $j \in \{1, \ldots, n\}$ the partition $\mathfrak{P}$ contains the set $\{p_{i,j}, p^1_{i,j}, p^2_{i,j}, p^3_{i,j}\}$ for exactly one $i \in \{1, \ldots, r\}$ these properties imply that the collection of all sets P_i for which cluster-scheme 1 is used in $\mathfrak{P}$ is an exact cover for $\{x_1, \ldots, x_n\}$.

Conversely, if $S_{i_1}, \ldots, S_{i_{\frac{n}{3}}}$ is an exact cover for $\{x_1, \ldots, x_n\}$, then we can build a 4-cluster of maximum diameter 2 for the set P by first using the basic partition for the points inherited from Theorem 28 (with s_i, s'_i replaced by b_i and t_i, t'_i replaced by e_i for each $i \in \{1, \ldots, r\}$) and adjusting it by swapping to cluster-scheme 1 for $P_{i_1}, \ldots, P_{i_{\frac{n}{3}}}$ as done there. For the remaining points in P we do the following: For each $j \in \{1, \ldots, n\}$, let $i_j \in \{i_1, \ldots, i_{\frac{n}{3}}\}$ be the (unique) index such that $x_j \in S_{i_j}$. First, assign the point $p_{i_j,j}$ to the cluster $\{p^1_{i_j,j}, p^2_{i_j,j}, p^3_{i_j,j}\}$. Then, build the following sets:

- $\{u_{i,j}, u^1_{i,j}\}$ and $\{h_{i,j}, u^2_{i+1,j}\}$ for all $i \in \{1, \ldots, r\}$ with $i < i_j$,
- $\{m_{i_j,j}, u_{i,j}\}$,
- $\{u^1_{i,j}, h_{i,j}\}$ and $\{u^2_{i+1,j}, u_{i+1,j}\}$ for all $i \in \{1, \ldots, r-1\}$ with $i \geq i_j$ and
- $\{m_{i,j}, p_{i,j}\}$ for all $i \in \{1, \ldots, r\} \setminus \{i_j\}$ such that $x_j \in S_i$.

The collection of the resulting sets is obviously a partition of P. As $S_{i_1}, \ldots, S_{i_{\frac{n}{3}}}$ is an exact cover for $\{x_1, \ldots, x_n\}$, all sets of cardinality 3 from the basic partition of the form $\{p^1_{i_j,j}, p^2_{i_j,j}, p^3_{i_j,j}\}$ turn into the set $\{p_{i,j}, p^1_{i_j,j}, p^2_{i_j,j}, p^3_{i_j,j}\}$ of cardinality 4 and diameter 2 by the distribution of the remaining points. All other sets have cardinality at least 4 and diameter at most 2 by construction, which makes the partition constructed by this procedure a 4-cluster of maximum diameter 2 for P. □

With very little changes to the previous construction, the hardness result can be transferred to the weighted ∞-norm, observe that most clusters in the optimal solution already have the minimum cardinality of 4. In some sense, we will use the relation between weighted and unweighted ∞-norm described in Equation 1 to translate the hardness result from one norm to the other.

Proposition 30

Euclidean $(\|\cdot\|^w_\infty, \mathrm{diam})$-$k$-cluster *with* $k = 4$ *and* $\delta = 3$ *is* NP*-hard.*

Proof. Consider the reduction used in the proof of Theorem 29. The proof argues that each 4-cluster of maximum diameter 2 for the points P has a very specific form which contains only clusters of exactly cardinality 4, except for clusters of the form $\{b_i, p_{i,1}^1\}$ or $\{e_i, l_i\}$, which have cardinality 6. We change the construction from Theorem 29 only for the points b_i and e_i to decrease the weighted cost of such clusters in the following way: Change for every $i \in \{1, \ldots, r\}$ the x-coordinates of b_i and e_i to 1.67 and $3n+4.33$, respectively. With this adjustment, the only points at distance at most 2 from b_i and e_i are still $p_{i,1}^1$ and l_i, respectively, so the possibilities to build sets of minimum cardinality 4 and maximum diameter 2 do not change. The clusters $\{b_i, p_{i,1}^1\}$ and $\{e_i, l_i\}$ now have cardinality 6 and diameter 1.33, which yields a weighted local cost of $6 \cdot 1.33 < 8$.

With this adjustment, we claim that there exists a 4-cluster for P of maximum weighted diameter 8 if and only if the corresponding instance of Exact-3-Cover is a "yes"-instance.

If there is a 4-cluster for P of maximum weighted diameter 8, each set in this solution has diameter at most 2. Theorem 29 shows that the existence of a 4-cluster for P of maximum diameter 2 implies the existence of an exact cover.

Conversely, with the cost of the clusters $\{b_i, p_{i,1}^1\}$ and $\{e_i, l_i\}$ now reduced to less than 8, the 4-cluster constructed according to a given exact cover as in the proof of Theorem 29 has maximum weighted diameter 8. □

4.4 Radius

Theorem 31

Euclidean $(\|\cdot\|_1^w, \mathrm{rad})$-$k$-cluster *with* $k = 12$ *and* $\delta = 2$ *is* NP*-hard.*

Proof. We reduce from Exact-3-Cover, with an instance given by a collection of sets $S_1, \ldots, S_r$ of cardinality 3 over the universe $\{x_1, \ldots, x_n\}$. Let again $\lambda > 0$ be a constant, to be specified later for readability. For each $i \in \{1, \ldots, r\}$, $j \in \{1, \ldots, n\}$ and $h \in \{1, 2, 3\}$, introduce the following points:

Name	$p_{i,j}^h$	$\bar{p}_{i,j}^h$	l_i	$\bar{l}_i$	s_i^1	s_i^2	s_i^3	t_i^1	t_i^2
x-Coord	$9j+3h$	$9j+3h$	$9n+12$	$9n+12$	2.5	$2.5+\lambda$	7.5	$9n+16.5$	$9n+21.5$
y-Coord	$28.4i-2$	$28.4i+2$	$28.4i-2$	$28.4i+2$	$28.4i$	$28.4i$	$28.4i$	$28.4i$	$28.4i$
Count	2	2	2	2	10	1	1	1	11

Let $P_i := \{p_{i,j}^h, \bar{p}_{i,j}^h \colon 1 \le j \le n, 1 \le h \le 3\} \cup \{t_i^1, t_i^2, l_i, \bar{l}_i\} \cup \{s_i^h \colon 1 \le h \le 3\}$ for each $i \in \{1, \ldots, r\}$ denote the set of points representing the set S_i. With this construction, the points in P_i are arranged as illustrated in the picture below for each $i \in \{1, \ldots, r\}$.

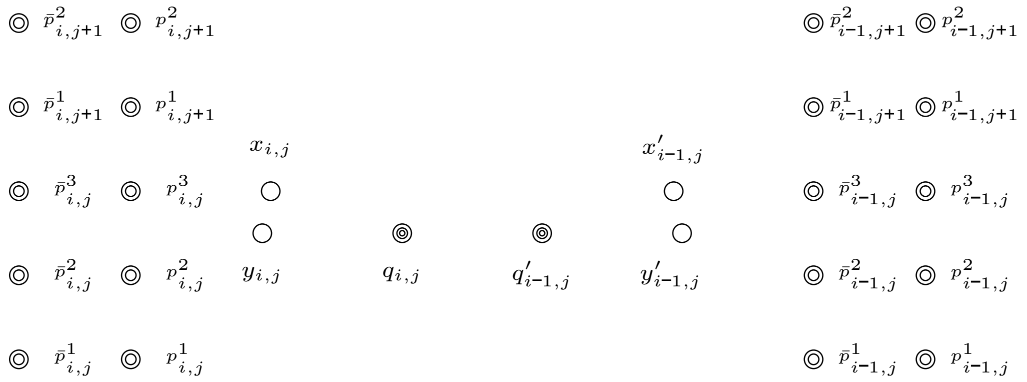
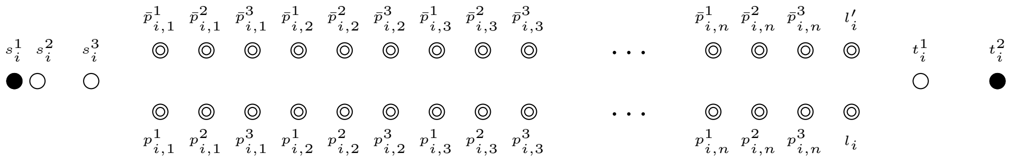


Figure 6: Layout of how the points $q_{i,j}, q'_{i-1,j}$, $x_{i,j}, y_{i,j}$ and $x'_{i-1,j}, y'_{i-1,j}$ are arranged between the points in P_{i-1} and P_i (rotated by 90°).



$\bar{p}^1_{i,1}$ $\bar{p}^2_{i,1}$ $\bar{p}^3_{i,1}$ $\bar{p}^1_{i,2}$ $\bar{p}^2_{i,2}$ $\bar{p}^3_{i,2}$ $\bar{p}^1_{i,3}$ $\bar{p}^2_{i,3}$ $\bar{p}^3_{i,3}$
$\bar{p}^1_{i,n}$ $\bar{p}^2_{i,n}$ $\bar{p}^3_{i,n}$ l'_i
s^1_i s^2_i s^3_i
t^1_i t^2_i
$p^1_{i,1}$ $p^2_{i,1}$ $p^3_{i,1}$ $p^1_{i,2}$ $p^2_{i,2}$ $p^3_{i,2}$ $p^1_{i,3}$ $p^2_{i,3}$ $p^3_{i,3}$
$p^1_{i,n}$ $p^2_{i,n}$ $p^3_{i,n}$ l_i

For each pair of indices (i, j) with $i \in \{1, \dots, r\}$ and $j \in \{1, \dots, n\}$ introduce the points $q_{i,j}, q'_{i,j}$ and if $x_j \in S_i$ the points $x_{i,j}, x'_{i,j}$, if $x_j \notin S_i$ the points $y_{i,j}, y'_{i,j}$ (except for $q_{r,j}, x_{r,j}, y_{r,j}$ and $q'_{1,j}, x'_{1,j}, y'_{1,j}$) with the following coordinates:

Name	$q_{i,j}$	$q'_{i,j}$	$y_{i,j}$	$y'_{i,j}$	$x_{i,j}$	$x'_{i,j}$
x-Coord	$9j+7.5$	$9j+7.5$	$9j+7.5$	$9j+7.5$	$9(j+1)$	$9(j+1)$
y-Coord	$28.4i-9.7$	$28.4i+13.7$	$28.4i-4.7$	$28.4i+8.7$	$28.4i-5$	$28.4i+9$
Count	6	6	1	1	1	1

These coordinates place the points $q_{i,j}, q'_{i-1,j}$ and either $x_{i,j}$ or $y_{i,j}$ and either $x'_{i-1,j}$ or $y'_{i-1,j}$ vertically between the points in P_{i-1} and P_i as illustrated in Figure 6.

Denote by P the set of all points introduced by this construction and consider P with the Euclidean norm as instance of $(\|\cdot\|_1^w, \mathrm{rad})$-$k$-CLUSTER. Let

$$C = 5 \cdot |P| - 12\frac{n}{3}\lambda = 26rn + 28r - 14n - 12\frac{n}{3}\lambda\,.$$

We claim that for $k = 12$ there exists a solution of global cost at most C for $(\|\cdot\|_1^w, \mathrm{rad})$-$k$-CLUSTER on P if and only if $S_1, \dots, S_r$ is a "yes"-instance for EXACT-3-COVER.

Assume there exists a 12-cluster $\mathfrak{P}$ for P of global cost at most C with respect to radius and weighted 1-norm. Define for each point $p \in P$ the cost $c(p)$ by the radius of the cluster in $\mathfrak{P}$ which contains p. The global cost of $\mathfrak{P}$

with respect to radius and 1-norm is exactly the sum of $c(p)$ over all points $p \in P$. Consider the minimum radius of a cluster of minimum cardinality 12 containing a point $p \in P$ as lower bound on the cost $c(p)$:[4]

- For $p = p^h_{i,j}$ or $p = \bar{p}^h_{i,j}$ for some $i \in \{1, \dots, r\}$, $j \in \{1, \dots, n\}$ and $h \in \{1, 2, 3\}$ with either $h > 1$ or $j > 1$, the construction implies $c(p) \geq 5$; observe that $d_2(p^h_{i,j}, \bar{p}^{h\%3+1}_{i,j}) = \sqrt{3^2 + 4^2} = 5$. This cost is only possible for the following types of clusters S containing p:

  – S contains a subset of cardinality at least 11 of $\{p^h_{i,j}, \bar{p}^h_{i,j} \colon 1 \leq h \leq 3\}$ and aside from these only possibly also $y_{i,j}$ or $y'_{i,j}$ but not both.

  – S contains a subset of $\{p^2_{i,j}, \bar{p}^2_{i,j}, p^3_{i,j}, \bar{p}^3_{i,j}, p^1_{i,j+1}, \bar{p}^1_{i,j+1}\}$ (for $j = n$, the set $\{p^2_{i,j}, \bar{p}^2_{i,j}, p^3_{i,j}, \bar{p}^3_{i,j}, l_i, \bar{l}_i\}$) of cardinality at least 11 and aside from these only possibly either $(x_{i,j}/y_{i,j})$ or $(x'_{i,j}/y'_{i,j})$ but not both.

  – S is the set $\{p^3_{i,j-1}, \bar{p}^3_{i,j-1}, p^1_{i,j}, \bar{p}^1_{i,j}, p^2_{i,j}, \bar{p}^2_{i,j}\}$ for $j > 1$.

  For other clusters, the cost $c(p)$ is larger than 5.8 (a smallest choice among these being a cluster with central vertex $y_{i,j}$ or $y'_{i,j}$).

- For $p = p^1_{i,1}$ or $p = \bar{p}^1_{i,1}$ for some $i \in \{1, \dots, r\}$, the $c(p)$ is at least 5. Additional to the clusters described for the other $p^h_{i,j}$ and $\bar{p}^h_{i,j}$, p has another option for a cluster with this minimum cost containing at least 8 vertices from $\{s^h_i \colon 1 \leq h \leq 3\}$. Otherwise, $c(p)$ is at least 6.

- For $p = l_i$ or $p = \bar{l}_i$ for some $i \in \{1, \dots, r\}$, the $c(p)$ is at least 5 and is only achieved in a cluster S which contains at least 12 vertices from either $\{p^2_{i,n}, \bar{p}^2_{i,n}, p^3_{i,n}, \bar{p}^3_{i,n}, l_i, \bar{l}_i, (x_{i,n}/y_{i,n})\}$, $\{p^2_{i,n}, \bar{p}^2_{i,n}, p^3_{i,n}, \bar{p}^3_{i,n}, l_i, \bar{l}_i, (x'_{i,n}/y'_{i,n})\}$ or $\{t^h_i, t^2_i\}$. For other clusters, $c(p)$ is at least 6.

- For $p = s^h_i$ for some $i \in \{1, \dots, r\}$ and $h \in \{1, 2, 3\}$, $c(p)$ is at least $5 - \lambda$. This cost is only possible if p is in exactly the cluster $\{s^h_i \colon 1 \leq h \leq 3\}$ for other clusters, the cost $c(p)$ is at least 5.

- For $p = t^h_i$ for some $i \in \{1, \dots, r\}$ and $h \in \{1, 2\}$, $c(p)$ is at least 5. This cost is only possible if p in exactly the cluster which is a subset of $\{t^1_i, t^2_i, l_i, \bar{l}_i\}$ for other clusters, the cost $c(p)$ is at least 7.7 .

- For $p = q_{i,j}$ for some $i \in \{1, \dots, r-1\}$ and $j \in \{1, \dots, n\}$, $c(p)$ is at least 5. This cost is only possible if p is in a cluster which contains at least 11 vertices from $\{q_{i,j}, q'_{i-1,j}\}$ and aside from these may only contain either $(x_{i,j}/y_{i,j})$ or $(x'_{i-1,j}/y'_{i-1,j})$. For other clusters, $c(p)$ is at least 5.8 (a cluster with central vertex $y_{i,j}$ and $y'_{i-1,j}$, respectively). Similarly for $p = q'_{i,j}$ for some $i \in \{2, \dots, r\}$ and $j \in \{1, \dots, n\}$ with at least 11 vertices from $\{q_{i+1,j}, q'_{i,j}\}$ and possibly either $(x'_{i,j}/y'_{i,j})$ or $(x_{i+1,j}/y_{i+1,j})$.

[4] The following considerations yield an equivalent to Observation 1 used for diameter.

- For $p = y_{i,j}$ for some $i \in \{1, \dots, r-1\}$ and $j \in \{1, \dots, n\}$, $c(p)$ is at least 5. This cost is only possible if p is in a cluster which otherwise only contains a subset of cardinality at least 11 from $\{p^h_{i,j}, \bar{p}^h_{i,j} \colon 1 \leq h \leq 3\}$ or $\{p^2_{i,j}, \bar{p}^2_{i,j}, p^3_{i,j}, \bar{p}^3_{i,j}, p^1_{i,j+1}, \bar{p}^1_{i,j+1}\}$ ($\{p^2_{i,n}, \bar{p}^2_{i,n}, p^3_{i,n}, \bar{p}^3_{i,n}, l_i, \bar{l}_i\}$, for $j = n$), or $\{q_{i,j}, q'_{i-1,j}\}$. For other clusters, $c(p)$ is at least 6.5. This holds similarly for $p = y'_{i,j}$ for some $i \in \{2, \dots, r\}$ and $j \in \{1, \dots, n\}$ (with $\{q_{i,j}, q'_{i-1,j}\}$ replaced by $\{q_{i+1,j}, q'_{i,j}\}$).

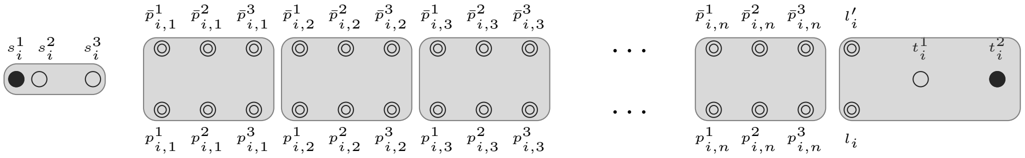
- For $p = x_{i,j}$ for some $i \in \{1, \dots, r-1\}$ and $j \in \{1, \dots, n\}$, $c(p)$ is at least 5. This cost is only possible if p is in a cluster which otherwise only contains a subset of cardinality at least 11 from exclusively either $\{p^2_{i,j}, \bar{p}^2_{i,j}, p^3_{i,j}, \bar{p}^3_{i,j}, p^1_{i,j+1}, \bar{p}^1_{i,j+1}\}$ ($\{p^2_{i,n}, \bar{p}^2_{i,n}, p^3_{i,n}, \bar{p}^3_{i,n}, l_i, \bar{l}_i\}$ for $j = n$, resp.) or $\{q_{i,j}, q'_{i-1,j}\}$. For other clusters, $c(p)$ is at least 5.8. This holds similarly for $p = x'_{i,j}$ for some $i \in \{2, \dots, r\}$ and $j \in \{1, \dots, n\}$ (with $\{q_{i,j}, q'_{i-1,j}\}$ replaced by $\{q_{i+1,j}, q'_{i,j}\}$).

Any 12-cluster for P hence has a minimum cost of at least $5 \cdot |V| - 12r\lambda$ with respect to radius and weighted 1-norm. In the following we set $\lambda = \frac{1}{2r}$, which implies by the observations about $c(p)$ above and the connection between $c(p)$ and the global cost of a clustering, that a 12-cluster $\mathfrak{P}$ of cost C for P is possible if and only if all points from $P \setminus \{s^h_i \colon 1 \leq i \leq r, 1 \leq h \leq 3\}$ have minimum cost $c(p)$ in $\mathfrak{P}$.

Assuming the existing of such a 12-cluster $\mathfrak{P}$ of cost at most C, we arrive at similar clustering-schemes for the points P_i as in the construction for diameter.

Observation 2*:* If s^h_i for some $i \in \{1, \dots, r\}$ and $h \in \{1, 2, 3\}$ has its minimum cost of $5 - \lambda$ in $\mathfrak{P}$, then $\mathfrak{P}$ restricted to the points in P_i induces the factorisation $\{s^h_i \colon 1 \leq h \leq 3\}, \{\{p^h_{i,j}, \bar{p}^h_{i,j} \colon 1 \leq h \leq 3\} \colon 1 \leq j \leq n\}, \{l_i, \bar{l}_i, t^1_i, t^2_i\}$.

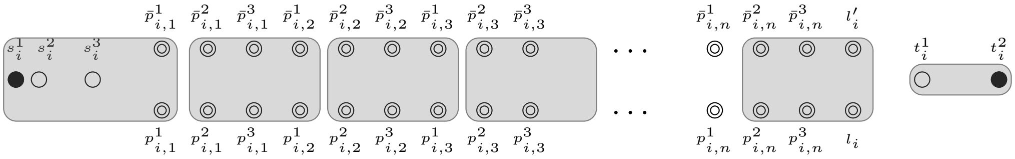
The factorisation described in Observation 2 is illustrated below:


s^1_i s^2_i s^3_i
$\bar{p}^1_{i,1}$ $\bar{p}^2_{i,1}$ $\bar{p}^3_{i,1}$ $\bar{p}^1_{i,2}$ $\bar{p}^2_{i,2}$ $\bar{p}^3_{i,2}$ $\bar{p}^1_{i,3}$ $\bar{p}^2_{i,3}$ $\bar{p}^3_{i,3}$
$\bar{p}^1_{i,n}$ $\bar{p}^2_{i,n}$ $\bar{p}^3_{i,n}$ l'_i
t^1_i t^2_i
$\dots$
$\dots$
$p^1_{i,1}$ $p^2_{i,1}$ $p^3_{i,1}$ $p^1_{i,2}$ $p^2_{i,2}$ $p^3_{i,2}$ $p^1_{i,3}$ $p^2_{i,3}$ $p^3_{i,3}$
$p^1_{i,n}$ $p^2_{i,n}$ $p^3_{i,n}$ l_i

This factorisation is its role the equivalent to cluster-scheme 1 introduced for the diameter measure, so we will use the same name here.

Observation 3*:* If, for some $i \in \{1 \dots, r\}$, the restriction of $\mathfrak{P}$ to the points in P_i is not cluster-scheme 1, then the restriction of $\mathfrak{P}$ to P_i is given by: $\{s^h_i \colon 1 \leq h \leq 3\} \cup \{p^1_{i,1}, \bar{p}^1_{i,1}\}$, $\{\{p^2_{i,j}, \bar{p}^2_{i,j}, p^3_{i,j}, \bar{p}^3_{i,j}, p^1_{i,j+1}, \bar{p}^1_{i,j+1}\} \colon 1 \leq j \leq n-1\}$, $\{p^2_{i,n}, \bar{p}^2_{i,n}, p^3_{i,n}, \bar{p}^3_{i,n}, l_i, \bar{l}_i\}$, $\{t^1_i, t^2_i\}$.

Just like for the diameter, denote the factorisation given in Observation 3 by *cluster-scheme 2*, see the picture below for an illustration:


$\bar p^1_{i,1}$ $\bar p^2_{i,1}$ $\bar p^3_{i,1}$ $\bar p^1_{i,2}$ $\bar p^2_{i,2}$ $\bar p^3_{i,2}$ $\bar p^1_{i,3}$ $\bar p^2_{i,3}$ $\bar p^3_{i,3}$
$\bar p^1_{i,n}$ $\bar p^2_{i,n}$ $\bar p^3_{i,n}$ l'_i
s^1_i s^2_i s^3_i
t^1_i t^2_i
...
...
$p^1_{i,1}$ $p^2_{i,1}$ $p^3_{i,1}$ $p^1_{i,2}$ $p^2_{i,2}$ $p^3_{i,2}$ $p^1_{i,3}$ $p^2_{i,3}$ $p^3_{i,3}$
$p^1_{i,n}$ $p^2_{i,n}$ $p^3_{i,n}$ l_i

The corresponding *basic partition* here is given by the following collection of sets (for an illustration see Figure 7):

Sets	Indices	Diameter
$\{s^1_i, s^2_i, s^3_i, p^1_{i,1}\}$	$1 \le i < r$	5
$\{p^2_{i,j}, p^3_{i,j}, p^1_{i,j+1}, p'^2_{i,j}, p'^3_{i,j}, p'^1_{i,j+1}, (x_{i,j} \backslash y_{i,j})\}$	$1 \le i < r,\ \ 1 \le j < n$	5
$\{p^2_{i,n}, p^3_{i,n}, p'^2_{i,n}, p'^3_{i,n}, l_i, l'_i, (x_{i,n} \backslash y_{i,n})\}$	$1 \le i < r$	5
$\{t^1_i, t^2_i\}$	$1 \le i < r$	5
$\{q_{i-1,j}, q'_{i,j}, (x'_{i,j} \backslash y'_{i,j})\}$	$1 < i < r,\ \ 1 \le j \le n$	5
$\{p^1_{r,j}, p^2_{r,j}, p^3_{r,j}, p'^1_{r,j}, p'^2_{r,j}, p'^3_{r,j}\}$	$1 \le j \le n$	5
$\{t_r, t'_r, l_r, l'_r\}$		5
$\{s^1_r, s^2_r, s^3_r\}$		$5 - \lambda$

This basic partition has a global cost of $5 \cdot |P| - 12\lambda$. Since the assumed 12-cluster $\mathfrak{P}$ has global cost at most $5 \cdot |P| - 12\frac{n}{3}\lambda$, there are at least $\frac{n}{3}$ indices i in $\{1, \ldots, r\}$ such that the points $\{s^1_i, s^2_i, s^3_i\}$ have their minimum cost of $5 - \lambda$ (observe that this is the only possible improvement, as all other points are forced to have their minimum cost $c(p)$), which means that P_i is clustered by cluster-scheme 1 in $\mathfrak{P}$. Let $P_{i_1}, \ldots, P_{i_s}$ be all sets for which cluster-scheme 1 is used in $\mathfrak{P}$. We claim that $S_{i_1}, \ldots, S_{i_s}$ is an exact cover for $\{x_1, \ldots, x_n\}$.

Assume that there is some $x_j \in \{x_1, \ldots, x_n\}$ and $a, b \in \{1, \ldots, s\}$, $a \neq b$ such that $x_j \in S_a \cap S_b$. Similar as for diameter, for each $i \in \{2, \ldots, r-1\}$, either $(x_{i,j}/y_{i,j})$ or $(x'_{i,j}/y'_{i,j})$ can not be in a cluster with points in P_i to create clusters of radius 5 which means that for each index $i \in \{2, \ldots, r-1\}$ the partition $\mathfrak{P}$ either contains the set $Q_{i,j} := \{q_{i,j}, q'_{i-1,j}, x_{i,j}\}$ or the set $Q'_{i,j} := \{q_{i+1,j}, q'_{i,j}, x'_{i,j}\}$. Since S_a and S_b are both clustered by cluster-scheme 1, the respective points $x_{a,j}, x'_{a,j}$ and $x_{b,j}, x'_{b,j}$ (which are in a cluster of radius 5 in $\mathfrak{P}$) can not be in a set with points from P_a and P_b, respectively. This means that $\mathfrak{P}$ contains $Q_{h,j}$ and $Q'_{h,j}$ for both $h = a$ and $h = b$ (considering non-existing border cases, i.e., $Q_{1,j}$ and $Q'_{r,j}$, as empty sets). Then however $\mathfrak{P}$ has to contain

$$|\{2, \ldots, r-1\}| - |\{2, \ldots, r-2\} \cap \{p, q\}| + 2|\{p, q\}| - |\{1, r\} \cap \{p, q\}| = r$$

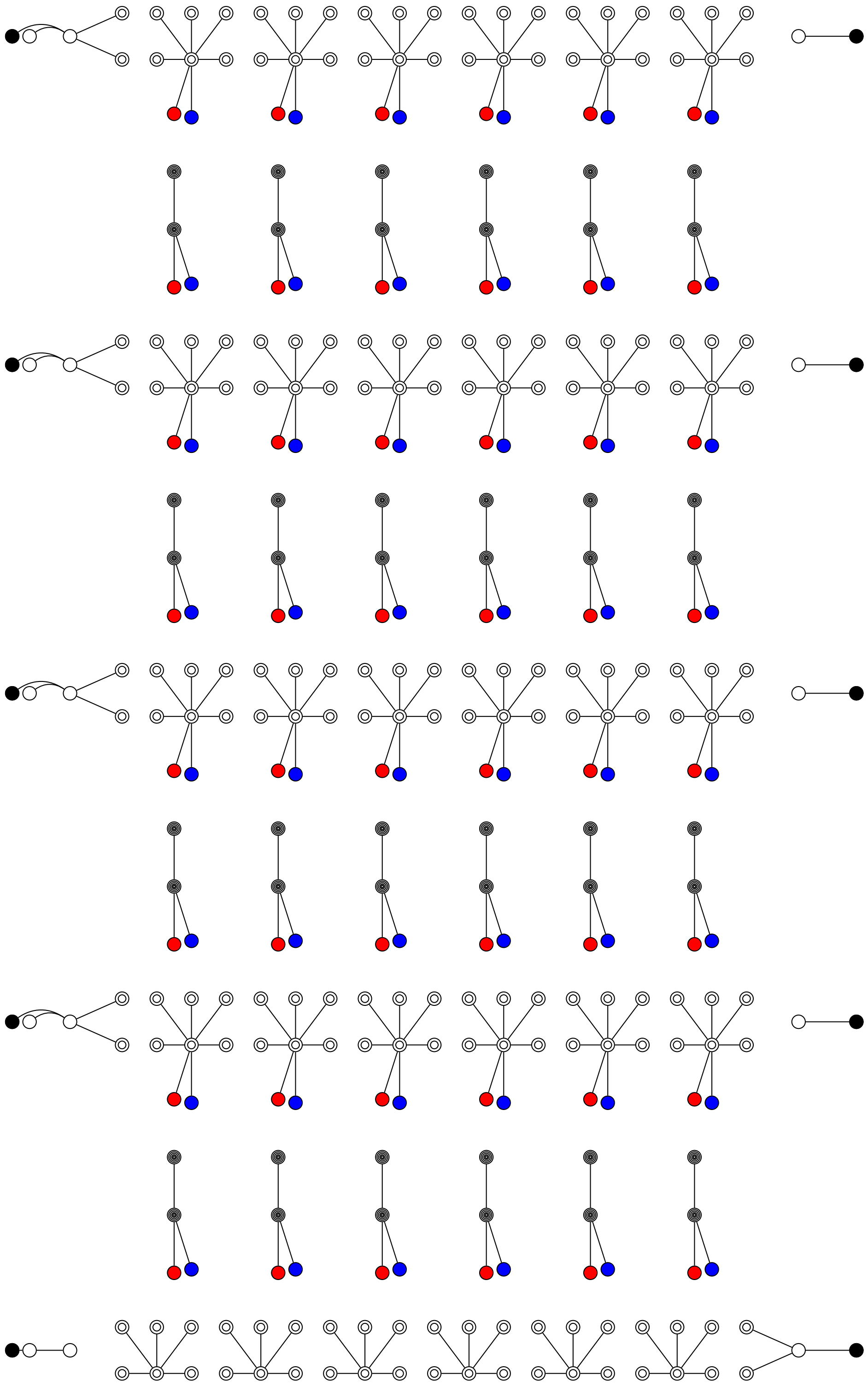


Figure 7: Example of the basic partition for Theorem 31 for $r = 5$ and $n = 6$. Edges illustrate the connection of a point to a central vertex in its cluster. Points $x_{i,j}$ in red and $y_{i,j}$ in blue.

sets from $\{Q_{i,j}, Q'_{i,j}\colon 1 \leq i \leq r\}$, which is not possible, as these sets have to be disjoint by definition, and there are only the $r-1$ groups of points $\{\{q_{i,j}, q'_{i-1,j}\}\colon 2 \leq i \leq r\}$ to build $Q_{i,j}$ or $Q'_{i,j}$. This means that $S_a \cap S_b = \emptyset$ for all $a, b \in \{i_1, \ldots, i_s\}$, $a \neq b$. Since $s \geq \frac{n}{3}$ by the global cost of $\mathfrak{P}$, it follows that $s = \frac{n}{3}$, as there can not be more than $\frac{n}{3}$ pairwise disjoint subsets of $\{x_1, \ldots, x_n\}$ of size 3, which makes $S_{i_1}, \ldots, S_{i_s}$ an exact cover for $\{x_1, \ldots, x_n\}$.

Conversely, if $S_{i_1}, \ldots, S_{i_{\frac{n}{3}}}$ is an exact cover for $\{x_1, \ldots, x_n\}$, a 12-cluster of global cost $5 \cdot |P| - 12\frac{n}{3}\lambda$ can be constructed by using cluster-scheme 1 for $P_{i_1}, \ldots, P_{i_{\frac{n}{3}}}$, cluster-scheme 2 for the remaining sets P_i and choosing $Q_{i,j}$ and $Q'_{i,j}$ accordingly. To be precise, denote for each $j \in \{1, \ldots, n\}$ by $z_j \in \{i_1, \ldots, i_{\frac{n}{3}}\}$ the index for which $x_j \in S_{z_j}$ and build the 12-cluster for P as follows:

- For each $j \in \{1, \ldots, n\}$, add the set $Q_{i,j}$ for all i with $i \leq z_j$ and the set $Q'_{i,j}$ for all i with $i \geq z_j$.
- For each $i \in \{i_1, \ldots, i_{\frac{n}{3}}\}$ and $j \in \{1, \ldots, n\}$ such that $j \in S_i$, include $\{p^h_{i,j}, \bar{p}^h_{i,j}\colon 1 \leq h \leq 3\}$.
- For each $i \in \{i_1, \ldots, i_{\frac{n}{3}}\}$ and $j \in \{1, \ldots, n\}$ such that $z_j > i$, add the set $\{p^h_{i,j}, \bar{p}^h_{i,j}, y_{i,j}\colon 1 \leq h \leq 3\}$.
- For each $i \in \{i_1, \ldots, i_{\frac{n}{3}}\}$ and $j \in \{1, \ldots, n\}$ such that $z_j < i$, add the set $\{p^h_{i,j}, \bar{p}^h_{i,j}, y'_{i,j}\colon 1 \leq h \leq 3\}$.
- For each $i \in \{i_1, \ldots, i_{\frac{n}{3}}\}$, build $\{s^h_i\colon 1 \leq h \leq 3\}$ and $\{t^1_i, t^2_i, l_i, \bar{l}_i\}$.
- For each $i \in \{1, \ldots, r\} \setminus \{i_1, \ldots, i_{\frac{n}{3}}\}$ and $j \in \{1, \ldots, n\}$ with $z_j > i$, add $\{p^2_{i,j}, p^3_{i,j}, p^1_{i,j+1}, \bar{p}^2_{i,j}, \bar{p}^3_{i,j}, \bar{p}^1_{i,j+1}, (x_{i,j}/y_{i,j})\}$.
- For each $i \in \{1, \ldots, r\} \setminus \{i_1, \ldots, i_{\frac{n}{3}}\}$ and $j \in \{1, \ldots, n\}$ with $z_j < i$, add $\{p^2_{i,j}, p^3_{i,j}, p^1_{i,j+1}, \bar{p}^2_{i,j}, \bar{p}^3_{i,j}, \bar{p}^1_{i,j+1}, (x'_{i,j}/y'_{i,j})\}$.
- For each $i \in \{1, \ldots, r\} \setminus \{i_1, \ldots, i_{\frac{n}{3}}\}$, add $\{s^h_i\colon 1 \leq h \leq 3\} \cup \{p_{i,1}, \bar{p}_{i,1}\}$ and $\{t^1_i, t^2_i\}$.

This construction obviously yields a 12-cluster for P with a global cost of $5|P| - 12\frac{n}{3}\lambda$. □

Remark 6*:* In the above construction, we used $\frac{1}{2r}$ as value for λ, for which one might argue that such small differences in distances do not occur in instances from the real world. We believe that the above proof also works for λ fixed to some constant which does not depend on r, similar to the construction for the diameter, but the argumentation just becomes much more complicated.

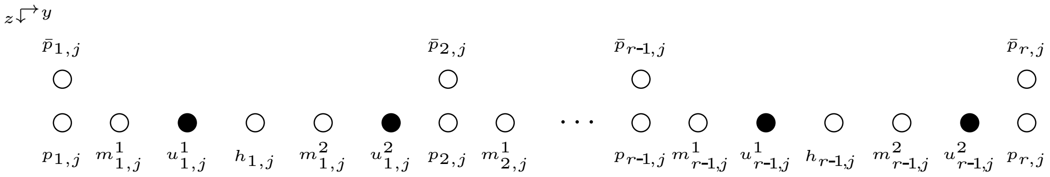


Figure 8: Layout of the points added to P for each $j \in \{1, \dots, n\}$ in Theorem 32, projected to the z- and y-axes.

Theorem 32

Euclidean $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-cluster *with* $k = 7$ *and* $\delta = 3$ *is* NP*-hard.*

Proof. We again reduce from Exact-3-Cover, with an instance given by a collection of sets $S_1, \dots, S_r$ of cardinality 3 over the universe $\{x_1, \dots, x_n\}$. We introduce the same set of points as in Theorem 31 just in the three dimensional space with z-coordinate set to 0, and the counts adjusted for each $i \in \{1, \dots, r\}$, $j \in \{1, \dots, n\}$ and $h \in \{1, 2, 3\}$, as follows:

Name	$p^h_{i,j}$	$\bar{p}^h_{i,j}$	l_i	$\bar{l}_i$	s^1_i	s^2_i	s^3_i	t^1_i	t^2_i	$q_{i,j}$	$q'_{i,j}$	$y_{i,j}$	$y'_{i,j}$	$x_{i,j}$	$x'_{i,j}$
Count	1	1	1	1	6	0	1	1	6	3	3	1	1	1	1

Additional to this set P, add for all $i \in \{1, \dots, r-1\}$ and $j \in \{1, \dots, n\}$ add the following points:

Name	Count	x-Coord	y-Coord	z-Coord
$m^1_{i,j}$	1	$9j + 6$	$28.4i + 6.2$	7.7
$u^1_{i,j}$	5	$9j + 6$	$28.4i + 11.2$	7.7
$h_{i,j}$	1	$9j + 6$	$28.4i + 16.2$	7.7
$m^2_{i,j}$	1	$9j + 6$	$28.4i + 21.2$	7.7
$u^2_{i,j}$	5	$9j + 6$	$28.4i + 26.2$	7.7

Further, for all pairs (i, j) with $i \in \{1, \dots, r\}$ and $j \in \{1, \dots, n\}$ include the point $p_{i,j}$ if $x_j \in S_i$ and the point $\bar{p}_{i,j}$ if $x_j \notin S_i$ with the following coordinates:

Name	Count	x-Coord	y-Coord	z-Coord
$p_{i,j}$	1	$9j + 6$	$28.4i + 2$	4.5
$\bar{p}_{i,j}$	1	$9j + 6$	$28.4i + 2$	7.7

For each $j \in \{1, \dots, n\}$, this construction adds the set of points arranged as illustrated in Figure 8.

Denote by P' the set of all points introduced by this construction and consider P' with the Euclidean norm as instance of $(\|\cdot\|_1^w, \mathrm{rad})$-$k$-CLUSTER. We claim that there exists a 7-cluster of maximum radius 5 for P' if and only if $S_1, \ldots, S_r$ is a "yes"-instance for EXACT-3-COVER.

Assume there exists a 7-cluster $\mathfrak{P}$ of maximum radius 5 for P'. First observe, simply by the difference in z-coordinate, that the only new points which have distance at most 5 from the points inherited from the construction for Theorem 31 are the points $p_{i,j}$, $i \in \{1, \ldots, r\}$, $j \in \{1, \ldots, n\}$. As the only points at distance at most 5 from $p_{i,j}$ are $m^1_{i,j}$, $p^2_{i,j}$ and $\bar{p}^2_{i,j}$ (each with count 1), there is no cluster in $\mathfrak{P}$ such that $p_{i,j}$ is central. Further, the only clusters in $\mathfrak{P}$ containing points both from P and $P' \setminus P$ are $\{p^h_{i,j}, \bar{p}^h_{i,j} \colon 1 \leq h \leq 3\} \cup \{p_{i,j}\}$ for some $i \in \{1, \ldots, r\}$ and $j \in \{1, \ldots, n\}$ with $x_j \in S_i$. By the observations already made in Theorem 31 (with adjusted cardinalities) and the simple linear layout of the new points, it is easy to see that further there are only the following options for subsets of P' of minimum cardinality 7 and maximum radius 5:

- Subsets of minimum cardinality 7 of $\{s^1_i, s^3_i, p^1_{i,1}, \bar{p}^1_{i,1}\}$ or $\{t^1_i, t^2_i, l_i, \bar{l}_i\}$, for $i \in \{1, \ldots, r\}$.
- The set $\{p^\ell_{i,j}, \bar{p}^\ell_{i,j} \colon 1 \leq \ell \leq 3\}$ together with exactly one of the points in $\{y_{i,j}, y'_{i,j}\}$, for $i \in \{1, \ldots, r\}$ and $j \in \{1, \ldots, n\}$ with $x_j \notin S_i$.
- The set $\{p^2_{i,j}, p^3_{i,j}, p^1_{i,j+1}, \bar{p}^2_{i,j}, \bar{p}^3_{i,j}, \bar{p}^1_{i,j+1}\}$ together with exactly one of the points in $\{(x_{i,j}/y_{i,j}), (x'_{i,j}/y'_{i,j})\}$, for $i \in \{1, \ldots, r\}$ and $j \in \{1, \ldots, n-1\}$; for $j = n$, the set $\{p^2_{i,j}, p^3_{i,j}, \{p^1_{i,j+1}, \bar{p}^2_{i,j}, l_i, \bar{l}_i\}$ with exactly one point from the set $\{(x_{i,n}/y_{i,n}), (x'_{i,n}/y'_{i,n})\}$. (Considering the border cases $i \in \{1, r\}$, there is no choice between $(x_{i,j}/y_{i,j})$ or $(x'_{i,j}/y'_{i,j})$ as only one of these points exists and the sets are hence completely fixed.)
- The set $\{q_{i,j}, q'_{i+1,j}\}$ together with exactly one of the points from the set $\{(x_{i,j}/y_{i,j}), (x'_{i+1}/y_{i+1,j})\}$ for $i \in \{1, \ldots, r-1\}$, $j \in \{1, \ldots, n\}$.
- $\{(p_{i,j}/\bar{p}_{i,j}), m^1_{i,j}, u^1_{i,j}\}$, for $i \in \{1 \ldots, r-1\}$ and $j \in \{1, \ldots, n\}$.
- $\{m^1_{i,j}, u^1_{i,j}, h_{i,j}\}$, for $i \in \{1 \ldots, r-1\}$ and $j \in \{1, \ldots, n\}$.
- $\{u^1_{i,j}, h_{i,j}, m^2_{i,j}\}$, for $i \in \{1 \ldots, r-1\}$ and $j \in \{1, \ldots, n\}$.
- $\{h_{i,j}, m^2_{i,j}\}, u^2_{i,j}$, for $i \in \{1 \ldots, r-1\}$ and $j \in \{1, \ldots, n\}$.
- $\{m^2_{i,j}, u^2_{i,j}, (p_{i+1,j}/\bar{p}_{i+1,j}), \}$, for $i \in \{1 \ldots, r-1\}$ and $j \in \{1, \ldots, n\}$.
- $\{u^2_{i,j}, (p_{i+1,j}/\bar{p}_{i+1,j}), m^1_{i+1,j}\}$, for $i \in \{1 \ldots, r-1\}$ and $j \in \{1, \ldots, n\}$.

For each $j \in \{1, \dots, n\}$, it follows by an argument inductive on $i = 1, \dots, r-1$, that the set in the 7-cluster $\mathfrak{P}$ which contains $m^{\ell}_{i,j}$ also contains all points $u^{\ell}_{i,j}$ for $\ell \in \{1, 2\}$. By the limited possibilities to build sets of cardinality 7 and radius at most 5, the clusters in $\mathfrak{P}$ containing $\{m^{\ell}_{i,j}, u^{\ell}_{i,j}\}$ also contain exclusively either $(p_{i+\ell-1,j}/\bar{p}_{i+\ell-1,j})$ or $h_{i,j}$ for each $i \in \{1, \dots, r-1\}$, $j \in \{1, \dots, n\}$ and $\ell \in \{1, 2\}$. The point $h_{i,j}$ on the other hand, also has to be in a cluster with one of the sets $\{m^{\ell}_{i,j}, u^{\ell}_{i,j}\}$, $\ell \in \{1, 2\}$. This means that for each index $j \in \{1, \dots, n\}$, exactly $r-1$ of the sets containing some $\{m^{\ell}_{i,j}, u^{\ell}_{i,j}\}$, $i \in \{1, \dots, r-1\}$, $j \in \{1, \dots, n\}$ and $\ell \in \{1, 2\}$ in $\mathfrak{P}$ also contain some point $h_{i,j}$, and also exactly $r-1$ contain $(p_{i,j}/\bar{p}_{i,j})$, which leaves exactly one index $i_j \in \{1, \dots, r\}$ such that $(p_{i_j,j}/\bar{p}_{i_j,j})$ is in a cluster in $\mathfrak{P}$ which otherwise only contains points from P, the cluster $\{p^h_{i_j,j}, \bar{p}^h_{i_j,j} \colon 1 \le h \le 3\} \cup \{p_{i_j,j}\}$ to be precise, which also requires that the index i_j is such that $x_j \in S_{i_j}$. For the set $\{p^h_{i_j,j}, \bar{p}^h_{i_j,j} \colon 1 \le h \le 3\} \cup \{p_{i_j,j}\}$ to be in $\mathfrak{P}$, the points from P_{i_j} have to be partitioned by cluster-scheme 1 (with adjusted cardinalities), which in turn means that the sets $\{p^h_{i_j,j'}, \bar{p}^h_{i_j,j'} \colon 1 \le h \le 3\} \cup \{p_{i_j,j'}\}$ are in $\mathfrak{P}$ for each $j' \in \{1, \dots, n\}$ with $x_{j'} \in S_{i_j}$, as the set $\{p^h_{i_j,j'}, \bar{p}^h_{i_j,j'} \colon 1 \le h \le 3\}$ only has cardinality 6 and $p_{i_j,j'}$ is the only point which can be added without increasing the radius. Just like in the proof of Theorem 31, using cluster-scheme 1 for two different sets P_a and P_b with $a \neq b$ such that there exists an index $j \in \{1, \dots, n\}$ with $x_j \in S_a \cap S_b$ leaves at least r vertices in $\{(x_{i,j}/y_{i,j}), (x'_{i,j}/y'_{i,j}) \colon 1 \le i \le r\}$ which have to be in a cluster with only the $r-1$ groups of points in $\{\{q_{i,j}, q'_{i-1,j} \colon 2 \le i \le r\}$. These properties imply that, just like in the proof of Theorem 31, the collection of sets S_i with i such that P_i is partitioned by cluster-scheme 1 is an exact cover for $\{x_1, \dots, x_n\}$.

Conversely, if $S_{i_1}, \dots, Si_{\frac{n}{3}}$ is an exact cover for $\{x_1, \dots, x_n\}$, a 7-cluster of maximum radius 5 for P' can be build by partitioning P just like described in Theorem 31 and then, with again z_j denoting the (unique) index in $\{i_1, \dots, i_{\frac{n}{3}}\}$ such that $x_j \in S_{z_j}$, partitioning the remaining points as follows:

- For each $j \in \{1, \dots, n\}$, add $p_{z_j,j}$ to the set $\{p^h_{z_j,j}, \bar{p}^h_{z_j,j} \colon 1 \le h \le 3\}$.
- For each $j \in \{1, \dots, n\}$, and $i \in \{1, \dots, r-1\}$ with $i < z_j$, build the sets $\{(p_{i,j}/\bar{p}_{i,j}), m^1_{i,j}, u^1_{i,j}\}$ and $\{h_{i,j}, m^2_{i,j}, u^2_{i,j}\}$.
- For each $j \in \{1, \dots, n\}$, and $i \in \{1, \dots, r-1\}$ with $i > z_j$, build the sets $\{m^1_{i,j}, u^1_{i,j}, h_{i,j}\}$ and $\{m^2_{i,j}, u^2_{i,j}, (p_{i+1,j}/\bar{p}_{i+1,j})\}$.

This resulting collection of sets is obviously a 7-cluster of maximum radius 5 for P'. □

Just like for the diameter measure, a translation to the weighted infinity norm now just requires little adjustment to assure that clusters of cardinality larger than k have a smaller radius to produce a smaller weighted cost.

Proposition 33
Euclidean $(\|\cdot\|_{\infty}^{w}, \mathrm{rad})$-$k$-cluster *with* $k = 7$ *and* $\delta = 3$ *is* NP*-hard.*

Proof. Start with the construction used for Theorem 32 and just replace the points s_i^1, s_i^3 by b_i^1, b_i^2, and the points t_i^1, t_i^2 by e_i^1, e_i^2 for each $i \in \{1, \ldots, r\}$ with coordinates:

Name	Count	x-Coord	y-Coord	z-Coord
b_i^1	6	4.8	$28.4i$	0
b_i^2	1	8.68	$28.4i$	0
e_i^1	1	$9n + 15.32$	$28.4i$	0
e_i^2	6	$9n + 19.2$	$28.4i$	0

This adjustment decreases the radius of the clusters $\{b_i^1, b_i^2, p_{i,1}^1, \bar{p}_{i_1}^1\}$ and $\{e_i^1, e_i^2, l_i, \bar{l}_i\}$ (which now replace the sets $\{s_i^1, s_i^3, p_{i,1}^1, \bar{p}_{i_1}^1\}$ and $\{t_i^1, t_i^2, l_i, \bar{l}_i\}$) to 3.88 (and the weighted radius to 34.92), without creating new possibilities to build sets of cardinality at least 7 and radius at most 5.

It is not hard to see that the argumentation for Theorem 32 can now be used to show that there exists a 7-cluster of weighted maximum radius at most 35 for P' if and only if P' was created for a "yes"-instance of Exact-3-Cover. □

4.5 Summary

The original construction from [53] which we adjusted here was for the k-Median problem in the plane, which measures the quality of a cluster with a function which relates best to what we have defined as average distortion. Though it seems that the constructions used for the radius measure probably also give a reduction for average distortion, a clean formal proof that this statement really holds, requires at the very least a much more involved case analysis of possible minimum costs $c(p)$ for each point p. We hence leave the NP-hardness of Euclidean $(\|\cdot\|, f)$-k-cluster for average distortion with dimension δ fixed to some constant as an open problem.

For radius and diameter with any norm, we have shown that there exist constant values for both k and w such that Euclidean $(\|\cdot\|, f)$-k-cluster remains NP-hard even when restricted to these. Table 3 summarises these values, for which we however do not know if they are optimal, in the sense of smallest possible; observe that hardness for larger values is implied by the reductions given in the stated results.

	rad	diam
$\|\cdot\|_\infty$	$k = 7$ $\delta = 3$ (Theorem 32)	$k = 4$ $\delta = 3$ (Theorem 29)
$\|\cdot\|_\infty^w$	$k = 7$ $\delta = 3$ (Proposition 33)	$k = 4$ $\delta = 3$ (Proposition 30)
$\|\cdot\|_1^w$	$k = 12$ $\delta = 2$ (Theorem 31)	$k = 6$ $\delta = 2$ (Theorem 28)

Table 3: Summary of NP-hardness results for Euclidean $(\|\cdot\|, f)$-cluster.

5 Non-Metric Instances

Until now, we have always restricted $(\|\cdot\|, f)$-k-CLUSTER to instances for which the distance d satisfies the triangle inequality. Proposition 17 appears to dismiss the possibility to find approximations if this property does not hold. Many other related problems show a similar behaviour with respect to their complexity for non-metric instances.

The problem UNCAPACITATED FACILITY LOCATION, for example, can be approximated with ratio 1.488 if restricted to metric instances, see [46]. For general, possibly non-metric, distances, it is only possible to compute a $\log(n)$-approximation, see [42] for one of many algorithms with this performance. The relation to SET COVER does not just provide the basis for this positive approximation result, but also transfers non-approximability. In particular, is known that that $\log(n)$ is the best approximation ratio for SET COVER by [26], assuming $\mathsf{P} \neq \mathsf{NP}$, and this hardness transfers to UNCAPACITATED FACILITY LOCATION by a very simple approximation-preserving reduction identifying sets with facilities and the universe with the set of customers.

Such helpful consequences of a restriction to triangle inequality have led to many approaches which assume that this property holds, as we did in Section 3. Another nice example of such an approach is given in [32], where the properties that come with a restriction to distances which satisfy triangle inequality are used to speed up the famous heuristic algorithm *k-means*, named after the clustering problem it is designed to approximate efficiently.

With our attempt to use the abstract problem $(\|\cdot\|, f)$-k-CLUSTER to approach clustering for recommender systems, we found that the assumption that d satisfies the triangle inequality is generally false. The so-called *Pearson correlation coefficient*, which is usually used as distance measure for recommendations, does not have this useful property and, as also observed in [57], practical instances show this non-metric behaviour. We therefore try in this section to find useful approximations even for what we informally refer to as *non-metric instances* as opposed to the metric instances discussed in Section 3.

One option that comes to mind, especially considering the general hardness from Proposition 17, is graph editing, i.e., a pre-processing step which tries to transform a given general instance, with preferably few changes, into an instance for which triangle inequality holds and the results from Section 3 can then be applied. This idea however has several drawbacks. Changes to a given instance always come at the price of distortion; altering edge-weights or even deleting vertices results in perturbation of the original data. This effect hence raises the task to find alterations which bring as little change to the original instance as possible. Such graph editing problems are then usually already hard problems themselves. The task to find a minimum number of vertices such that their removal from a given instance of $(\|\cdot\|, f)$-k-CLUSTER is metric, for example, is closely related to the minimum vertex cover problem.

In this section, we seek different approaches which include an extra treatment for violations of the triangle inequality within the approximations for metric instances given in Section 3. The basic idea is to investigate the consequences of violated triangle inequality and devise strategies to deal with those within moderate exponential time depending on, roughly speaking, how much the given distance d differs from a metric.

More precisely, we will first look at the set of edges $\{u, v\}$ which directly violate the triangle inequality, i.e., there exists another vertex x such that $d(u, v) > d(u, x) + d(v, x)$. We call such edges *conflicts*; observe that by our model, only distances defined by an edge can exhibit such behaviour. If the set of conflicts for a given instance is empty, the associated distance obviously satisfies the triangle inequality, which makes the cardinality of the set of conflicts a reasonable measure of how far the instance is from being a metric. Our strategy then is to alter the algorithms for metric instances in such a way that they still yield constant-factor approximations even if the input includes conflicts, while only spending exponentially more effort with respect to these conflicts. Formally, this gives a parameterised approximation with structural parameterisation by the number of conflicts.

This kind of parameterisation by conflicts to improve approximabilty can be seen as a generalisation of the *distance from triviality* approach introduced in [41]. The idea there is to define for a given problem some *distance* which specifies how much a given instance differs from some structural property which makes the problem easy to solve, and use this measure as parameter. The term *triviality* there already refers to the broader case of polynomial time solvable instances, not just trivial inputs as one might think, and in our case we go one step further and see the number of conflicts as the distance to an instance which can be approximated efficiently.

For this conceptual idea of parameterisation by conflicts, we will further discuss other related parameters, such as the number of vertices involved in a conflict (referred to as *conflict vertices*) and subsets of these. Very briefly, we will also consider *shortcut vertices*, a name we use to denote vertices which create a conflict by providing the shorter path of length 2, i.e., vertices x for which there exists an edge $\{u, v\}$ such that $d(u, v) > d(u, x) + d(v, x)$.

Aside from this parameterisation by conflicts, conflict vertices or shortcuts we will also discuss a very different approach to find approximations for $(\|\cdot\|, f)$-k-CLUSTER on non-metric instances. Instead of using conflicts as measurement to determine how much a given distance function d differs from a metric, we consider the magnitude to which the triangle inequality is violated. Formally this yields the notion of *α-relaxed triangle inequality.* For this relaxation, we investigate how much the performance ratio of the approximations for metric instances suffers if generalised to α-relaxed triangle inequality.

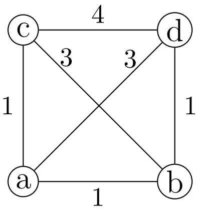


Figure 9: Edge $\{c, d\}$ has a weight larger than the shortest path from c to d but it is not a conflict. Lowering the weights on the edges $\{a, d\}$ and $\{b, c\}$ to 2 in order to remove these conflicts however results in an instance where $\{c, d\}$ becomes a conflict.

5.1 Definition of Conflicts

From a theoretical point of view in our abstract framework, for the given graph $G = (V, E)$, the edge-weight w_E might be such that an edge $\{u, v\} \in E$ has a weight larger than the cost of a cheapest path from u to v in G. Such edges result in an induced distance d which violates the triangle inequality for at least one pair of vertices on this cheapest path (including u and v). We define the set of *conflicts* with respect to the induced distance d as the collection C of of vertex pairs $\{u, v\}$ such that the triangle inequality is violated for u and v, formally:

$$C = \{\{u, v\} \in V \times V : \exists \ x \in V : d(u, v) > d(u, x) + d(v, x)\} .$$

Recall that violations of the triangle inequality for the distance d by definition only occur for edges of the input graph, so C is always a subset of E.

Observe that this set of conflicts is not necessarily the whole set of edges with a weight larger than the cheapest path in the graph (for a counterexample see Figure 9). Considering the option of graph editing with weight reduction to achieve triangle inequality, C might be smaller than the set of edge-weights which would have to be adjusted in order to arrive at a graph without conflicts.

We will also consider parameterisation by the cardinality of the set P of *conflict vertices*, which simply are the vertices involved in a conflict in C, formally defined by:

$$P = \bigcup_{\{u,v\} \in C} \{u, v\} .$$

While lowering the weights of edges in C does not always create a metric instance, deleting all vertices from P does. Vertex removal however may result in severe perturbation and we will see that our approximation strategies will work without such changes while some even allow for a parameterisation by strict subsets of P.

In the following we will use c and p for the parameters number of conflicts and number of conflict vertices, respectively. Parameterisation by p yields the same general tractability as parameterisation by p as the following relation holds:

$$p \; \leq \; 2c \; \leq \; (p(p-1))\,. \tag{12}$$

For the concrete running times, it is however still relevant to distinguish between parameters p and c as the bounds given in Equation 12 are sharp.

In the set C, we only store edges $\{u, v\}$ for which there exists a *shortcut vertex* $x \in V$, i.e., the inequality $d(u, v) > d(u, x) + d(v, x)$ holds. Removing such shortcut vertices is a different approach to resolve conflicts. Therefore, we will also briefly consider the set X of shortcut vertices, formally given by:

$$X = \{x \in V : \exists\ \{u, v\} \in C : d(u, v) > d(u, x) + d(v, x)\}\,.$$

Other than the fact that $X \neq \emptyset$ if and only if $C \neq \emptyset$, there is no general correlation between the cardinality of X and C. Parameterisation by the parameter $x := |X|$ hence gives a completely different perspective.

Another completely different way to measure the severity of conflicts, is the magnitude to which the triangle inequality is violated. The following relaxation of the triangle inequality constraint is quite obvious and is hence discussed under different names in the literature. The *Encyclopedia of Distances* [24] uses the name *C-relaxed triangle inequality.* As we have already reserved the letter C to denote set of conflicts, we will switch to the symbol α and formally say that the distance function d satisfies *α-relaxed triangle inequality*[5] for some $\alpha \geq 0$ if:

$$d(u, v) \leq \alpha \cdot (d(u, x) + d(v, x)) \text{ for all } u, v, x \in V \text{ with } \quad x \notin \{u, v\}\,.$$

5.2 Parameterisation by Conflicts

Starting with the greedy procedure introduced for the infinity norm with radius or diameter in Section 34, it is not too hard to see that, at least for the radius measure, a constant number of conflicts does not yield too much trouble. More precisely, the following result holds:

Theorem 34

A 2-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(n^p)$.

Proof. Given an instance (G, d) for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER with $G = (V, E)$ and conflict vertices $P \subset V$, we first guess for all $v \in P$ which vertex $c(v) \in V$

[5] The case $\alpha < 1$ is stricter than the usual triangle inequality and will yield even better approximation results; hence the restriction $x \notin \{u, v\}$ to enable this case.

is central in the cluster containing v for some optimal k-cluster for G. For a fixed choice of central vertices $Z := \{c(v) \colon v \in P\}$ we run the greedy procedure from Theorem 20 with the following small alterations:

- We only consider values D for which $d(v, c(v)) \leq D$ for all $v \in P$.
- We first greedily build the clusters $P(c(v))$ for all $v \in P$. More precisely, we first choose centres from the set Z, and add to $P(z)$ with $z \in Z$ the vertex z and all unclustered vertices $w \in V \setminus (P \cup Z)$ with $d(z, w) \leq D$ and all vertices $v \in P$ with $c(v) = z$.
- In the max-flow procedure, we only allow reassignment of the vertices in $V \setminus P$.

It is not hard to see that, for the correct choice of central vertices, this procedure still yields a 2-approximation for the radius measure. The arguments given in the proof of Theorem 20 to show this still hold with the additional property, that the vertices in Z are central by their correct choice, and according vertices to build clusters of cardinality at least k and radius at most D hence exist. The vertices in P are assigned optimally by the correct choice. For the vertices in $V \setminus P$, the triangle inequality holds and all previously used properties remain true. This means that the greedy procedure is successful for the correct choice of central vertices for each $v \in P$ and for D chosen as twice the optimum value for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER on (G, d).
Deterministically, the guessing of central vertices means trying all $\mathcal{O}(n^p)$ possible combinations of choices. Running the polynomial approximation procedure and then picking the choice which yields the solution of smallest global cost hence yields an overall running time in $\mathcal{O}^*(n^p)$ for the described parameterised 2-approximation. □

In terms of parameterised complexity, this result can be interpreted as XP-membership of the 2-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER from Theorem 20 for non-metric instances with respect to parameterisation by p (and also by c, by Equation 12). This result raises the question whether an improvement to a more efficient running time with respect to this parameterisation is possible, i.e., some constant factor approximation in fpt-time; recall that an improvement of the approximation ratio of 2 remains unlikely by Theorem 8. For the diameter, the concept of central vertices does not have an appropriate meaning, so the above considerations already indicate that this local measure requires a different strategy.

5.2.1 Resolving Conflicts in the Greedy Approximation

In the following we will try to use parameterisation by p to resolve the influence of conflicts on the 2-approximation algorithm presented in Theorem 20 with a running time that is only exponential in p. While the polynomial procedure described in this algorithm is identical for radius and diameter, we find that the influence of conflicts is different for these two local measures. Therefore, we will develop independent strategies for radius and diameter. Still, the basic idea of fixing a maximum radius D, greedily building a preliminary clustering with clusters of radius D and then balancing the cardinalities with a network always remains.

For the following result, recall that B_n denotes the nth Bell number, bounded by $B_n < \left(\frac{0.792n}{\log(n+1)}\right)^n$.

Theorem 35

A 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(B_p)$.

Proof. Let d be the distance function induced by the input graph $G = (V, E)$ with edge-weights w_E. We try for each $D \in \{d(v, w) \colon v, w \in V\}$ to build a k-cluster with maximum radius D, such that the diameter is at most $2D$. For each such fixed D we consider the following subset of the conflict set C:

$$C_D := \{\{u, v\} \in C \colon d(u, v) > 2D, \exists\, x \in V \colon\; d(u, x), d(v, x) \leq D\}\,.$$

And the corresponding subset of P:

$$P_D = \bigcup_{\{u,v\} \in C_D} \{u, v\}\,.$$

For each partition $C^1, \ldots, C^r$ for P_D such that $\mathrm{diam}(C^i) \leq D$ for all $i \in \{1, \ldots, r\}$, build a partition for the whole set of vertices with the following strategy:

(a) Iteratively, for $i = 1, \ldots, r$, pick an arbitrary vertex $c_i \in C^i$ and create a cluster $P(c_i)$ by adding to C^i all at this point unclustered vertices v in $V \setminus P_D$ with $d(c_i, v) \leq D$.

(b) If all clusters $P(c_i)$ are built but $\bigcup_{i=1}^r P(c_i) \neq V$, then repeat the following until all vertices are clustered: Pick any unclustered vertex z and create a cluster $P(z)$ containing all unclustered vertices which have distance at most D from z (including z itself).

Let $z_1, \ldots, z_q$ be the vertices chosen in step (b). The sets $P(c_1), \ldots, P(c_r)$, $P(z_1), \ldots, P(z_q)$ are a partition of V but not necessarily a k-cluster, as the cardinality constraint might not be satisfied. We now try to reassign vertices from $V \setminus P_D$ in order to move at least k vertices into each cluster while maintaining the property that all vertices in $P(c_i)$ ($P(z_j)$) have distance at most D from c_i (z_j resp.) for all $1 \leq i \leq r$ ($1 \leq j \leq q$ resp.). Observe that by the strategy used to build the clusters, possible vertices outside $P(c_i)$ at distance at most D from c_i can only be in clusters $P(c_j)$ with $j < i$. Vertices outside $P(z_i)$ at distance at most D from z_i are either in a set $P(c_j)$ for some $j \in \{1, \dots, r\}$ or in a set $P(z_l)$ with $l < i$. Hence, we define the sets:

- $S_c(i,j) := \{v \in P(c_j) \setminus P_D \colon d(v,c_i) \leq D\}$ for all $1 \leq j < i \leq r$ to collect all vertices which can be moved from cluster $P(c_j)$ to cluster $P(c_i)$,
- $S(i,j) := \{v \in P(c_j) \setminus P_D \colon d(v,z_i) \leq D\}$ for all $1 \leq j \leq r$ and $1 \leq i \leq q$ to collect all vertices which can be moved from cluster $P(c_j)$ to cluster $P(z_i)$ and
- $S_z(i,j) := \{v \in P(z_j) \setminus \{z_j\} \colon d(v,z_i) \leq D\}$ for all $1 \leq j < i \leq q$ to collect all vertices which can be moved from cluster $P(z_j)$ to cluster $P(z_i)$.

So, if $\sum_{j=1}^{i-1} |S_c(i,j)| < k - |P(c_i)|$ for some $i \in \{1, \dots, r\}$ or $\sum_{j=1}^{r} |S(i,j)| + \sum_{\ell=1}^{i-1} |S_z(i,\ell)| < k - |P(z_i)|$ for some $i \in \{1, \dots, q\}$, there exist no reassignment of vertices to turn the given partition into a k-cluster while maintaining the minimum distance D from the chosen central vertices. This especially holds if $|P(c_1)| < k$. In such a case we abort the iteration for this choice of D and partition for P_D. Otherwise, we try to reassign the vertices in the set:

$$\mathcal{S} := (\bigcup_{j=1}^{r-1} \bigcup_{i=j+1}^{r} S_c(i,j)) \cup (\bigcup_{j=1}^{r} \bigcup_{i=1}^{q} S(i,j)) \cup (\bigcup_{j=1}^{q-1} \bigcup_{i=j+1}^{q} S_z(i,j)) .$$

We build a network to move at least k vertices into each cluster in the following way:

- The network has a source s and target t.
- For each $i \in \{1, \dots, r\}$ we create a network vertex c_i' representing $P(c_i)$. If $|P(c_i)| > k$ we add the arc (s, c_i') with capacity $|P(c_i)| - k$. If $|P(c_i)| < k$ we add the arc (c_i', t) with capacity $k - |P(c_i)|$.
- For each $i \in \{1, \dots, q\}$ we create a network vertex z_i' representing $P(z_i)$. If $|P(z_i)| > k$ we add the arc (s, z_i') with capacity $|P(z_i)| - k$. If otherwise $|P(z_i)| < k$ we add the arc (z_i', t) with capacity $k - |P(z_i)|$.
- For each vertex $v \in \mathcal{S}$ we create a corresponding network vertex v' in the network connected with the following arcs, each of capacity 1.
  - (c_i', v') for all $i \in \{1, \dots, r\}$ and $v \in P(c_i)$,
  - (z_i', v') for all $i \in \{1, \dots, q\}$ and $v \in P(z_i)$,
  - (v', c_i') for all $i \in \{2, \dots, r\}$ with $v \in S_c(j,i)$ for some $j < i$,
  - (v', z_i') for all $i \in \{1, \dots, q\}$ with $v \in S(j,i)$ for some $j \in \{1, \dots, r\}$,
  - (v', z_i') for all $i \in \{2, \dots, q\}$ with $v \in S_z(j,i)$ for some $j < i$.

With these definitions, the maximum flow from s to t in this network is at most:

$$\sum_{i=2}^{r} \max\{0, k - |P(c_i)|\} + \sum_{j=1}^{q} \max\{0, k - |P(z_j)|\}\,.$$

A flow with this maximum capacity exists if and only if we can find a reassignment of some vertices in $\mathcal{S}$ to turn $P(c_1), \dots, P(c_r)$, $P(z_1), \dots, P(z_q)$ into a k-cluster while maintaining the maximum radius of D with respect to the chosen central vertices for each cluster. We claim that for the optimal diameter D^* there exists a partition for C_{D^*} such that the above procedure successfully produces a k-cluster for V with maximum diameter at most $2D^*$. Let $S_1, \dots, S_y$ be an optimal solution for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER on instance G. Consider running the above procedure for $D = D^*$ and the partition $C^1, \dots, C^r$ for P_D given by $C^i = P_D \cap S_{j_i^c}$ with $\{j_1^c, \dots, j_r^c\} = \{i \colon 1 \leq i \leq y, P_D \cap S_i \neq \emptyset\}$ such that $j_1^c < j_2^c < \cdots < j_r^c$.

Let $c_1, \dots, c_r$ be an arbitrary choice of representatives $c_i \in C^i$, $1 \leq i \leq r$ to assign vertices in step (a). By definition, for each $1 \leq i \leq r$ the set C^i is included in a different optimal cluster $S_{j_i^c}$. For the vertices $z_1, \dots, z_q$ chosen in step (b), we know that $d(z_i, c_j) > D$ and $d(z_i, z_l) > D$ for all $1 \leq i < l \leq q$ and $1 \leq j \leq r$, so each z_i belongs to a distinct cluster $S_{j_i^z}$ of the chosen optimal solution with $j_i^z \notin \{j_1^c, \dots, j_r^c\}$ and $j_i^z \neq j_l^z$ for all $1 \leq i < l \leq q$. So there exist at least the k vertices from $S_{j_i^c}$ at distance at most D from c_i to be assigned to $P(c_i)$ for each $1 \leq i \leq r$ and, similarly, at least the k vertices from $S_{j_i^z}$ at distance at most D from z_i to be assigned to $P(z_i)$ for each $1 \leq i \leq q$. The max-flow procedure can hence successfully build a k-cluster from the partition $P(c_1), \dots, P(c_r), P(z_1), \dots, P(z_q)$.

On the other hand, in case the described procedure is successful in computing a partition $P'(c_1), \dots, P'(c_r)$, $P'(z_1), \dots, P'(z_q)$ for some value D, it is obvious that these sets are a partition of V and that the reassignment of vertices with the max-flow procedure makes sure that each set contains at least k vertices, so the partition $P'(c_1), \dots, P'(c_r)$, $P'(z_1), \dots, P'(z_q)$ is a k-cluster for V. Consider the maximum diameter for this k-cluster:

- Each vertex $v \in P'(c_i)$ for some $i \in \{1, \dots, r\}$ is from C^i or it was either assigned to $P(c_i)$ in step (a) or moved by the max-flow procedure. In the latter two cases, v is not in P_D and was included in $P'(c_i)$ because $d(v, c_i) \leq D$ holds.
  - For $v, w \in C^i$ it follows that $d(v, w) \leq D$ as the set C^i in the chosen partition of P_D has diameter at most D by definition.
  - For $v \in C^i$ and $w \in P'(c_i) \setminus C^i$, we know that $d(v, c_i) \leq D$ (from the fact that $v, c_i \in C^i$) and $d(w, c_i) \leq D$. Since $w \notin P_D$ it follows that especially w and v do not create a conflict in C_D, so $d(v, w) \leq d(v, c_i) + d(c_i, w) \leq 2D$.

  - For $v, w \in P'(c_i) \setminus C^i$, we know that $v, w \notin P_D$,which implies $d(v, w) \leq d(v, c_i) + d(c_i, w) \leq 2D$.

- Each vertex $v \in P'(z_i)$ for some $i \in \{1, \ldots, q\}$ is not in P_D and was either assigned to $P(z_i)$ in step (b) or moved by the max-flow procedure. For all such vertices, $d(v, z_i) \leq D$ holds, so since no vertex from $P'(z_i)$ is included in a conflict in C_D, it follows that $d(v, w) \leq d(v, z_i) + d(w, z_i) \leq 2D$ for all $v, w \in P'(z_i)$.

The k-cluster $P'(c_1), \ldots, P'(c_r), P'(z_1), \ldots, P'(z_q)$ hence is a 2-approximation for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER on G. As there are $B_{|P_D|} \leq B_p$ partitions for P_D, the overall running time of the approximation procedure is in $\mathcal{O}^*(B_p)$. □

Theorem 36

A 3-approximation for $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(2^p)$.

Proof. Let d be the distance function induced by the input graph $G = (V, E)$. By binary search among the values $D \in \{d(v, w) \colon v, w \in V\}$ we determine the smallest value D for which the clustering procedure described below successfully builds a k-cluster. For each such fixed D we consider the following subset of vertices which are involved in a conflict:

$$P_D := \{u \colon \exists\, v, x \in V \colon (\{u, v\} \in C) \wedge (d(u, x), d(v, x) \leq 2D)\}\,.$$

We guess which of the vertices in P_D are a central vertex in their cluster by considering all $2^{|P_D|}$ subsets of P_D. For each such $P' \subseteq P_D$ we try to compute a k-cluster for V by successively building clusters until all vertices are partitioned with the following strategy:

(a) Pick, while such a vertex exists, a $v \in P'$ that is not assigned to any cluster yet and build a new cluster $P(v)$ with center v by collecting v and all unclustered vertices in $V \setminus P'$ which have distance at most D from v.

(b) If all vertices in P' are clustered, pick any $v \in V \setminus P_D$ that is not clustered yet and build a new cluster $P(v)$ with center v by collecting v and all unclustered vertices in V which have distance at most $2D$ from v.

(c) If the only unclustered vertices are in $P_D \setminus P'$, choose any of these unclustered vertices w and find a vertex $v_w \in V \setminus P_D$ of minimum distance to w. If this minimum distance is larger than D, abort the clustering process: Otherwise add w to the cluster which contains v_w.

Let $P' = \{p_1, \ldots, p_r\}$ and let $z_1, \ldots, z_q$ be the vertices chosen in step (b) to build clusters $P(z_1), \ldots, P(z_t)$ by the above procedure. We now try to turn the partition $P(p_1), \ldots, P(p_r), P(z_1), \ldots, P(z_q)$ into a k-cluster for V by reassigning some vertices. Here, the aim is to keep a maximum radius of D for $P(p_1), \ldots, P(p_r)$ and a maximum radius of $2D$ for $P(z_1), \ldots, P(z_q)$, which yields the following types of vertices which are allowed to be moved:

- $S_p(i,j) := \{v \in P(p_j) \backslash \{p_j\} \colon d(v, p_i) \leq D\}$ for all $1 \leq j < i \leq r$ to collect all vertices which can be moved from cluster $P(p_j)$ to cluster $P(p_i)$,
- $S(i,j) := \{v \in P(p_j) \setminus \{p_j\} \colon d(v, z_i) \leq 2D\}$ for all $1 \leq j \leq r$ and $1 \leq i \leq q$ to collect all vertices which can be moved from cluster $P(p_j)$ to cluster $P(z_i)$ and
- $S_z(i,j) := \{v \in P(z_j) \setminus \{z_j\} \colon d(v, z_i) \leq 2D\}$ for all $1 \leq j < i \leq q$ to collect all vertices which can be moved from $P(z_j)$ to $P(z_i)$.

Just like in Theorem 35, we try to turn $P(p_1), \ldots, P(p_r), P(z_1), \ldots, P(z_w)$ into a k-cluster by reassigning vertices in the above described sets with the help of a max-flow formulation. Denote in case of a successful reassignment, the resulting k-cluster by $P'(p_1), \ldots, P'(p_r)$, $P'(z_1), \ldots, P'(z_q)$. We claim that for $D = r^*$, there exists a subset $P' \subseteq P_D$ such that the above clustering procedure successfully computes a k-cluster $P'(p_1), \ldots, P'(p_r)$, $P'(z_1), \ldots, P'(z_q)$. Let $\{S_1, \ldots, S_y\}$ be any optimal solution for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER on input $G = (V, E)$ with distance d. Fix some central vertex s_i for S_i for each $i \in \{1, \ldots, y\}$. Consider running the described greedy procedure for $D = r^*$ and the subset $P' = P_D \cap \{s_1, \ldots, s_y\}$. First, observe that steps $(a) - (c)$ are successful in finding a preliminary clustering $P(p_1), \ldots, P(p_r), P(z_1), \ldots, P(z_q)$, since for each $v \in P_D \setminus P'$, there exists at least one vertex $s \in \{s_1, \ldots, s_y\}$ at distance at most D from v and, by the choice of P', this vertex s is not in $P_D \setminus P'$.

Further, any two different vertices in $P' \cup \{z_1, \ldots, z_q\}$ belong to different clusters in the chosen solution $\{S_1, \ldots, S_y\}$: For two vertices from P' this is true by the choice of P'. For a vertex z_i and any $w \in P'$, we know that $d(z_i, w) > D = r^*$, so, since w is central for some cluster S_j which has radius at most r^*, z_i cannot belong to S_j. For any two vertices z_i, z_j with $i < j$, we know that z_j was not clustered in $P(z_i)$ because $d(z_i, z_j) > 2D$. If there was a cluster S_h in the optimal solution such that $\{z_i, z_j\} \subseteq S_h$, then this would imply that $d(s_h, z_i) \leq r^*$ and $d(s_h, z_j) \leq r^*$, while $d(z_i, z_j) > 2D$, which would mean that $\{z_i, z_j\}$ is a conflict for this choice of D which is not possible (recall that the vertices z_i and z_j are chosen as central vertices in step (b), so they do not belong to P_D).

By the choice of P' there exist at least $k-1$ distinct vertices at distance at most D for each p_i. Since the vertices z_j are not in P_D, they have a distance of at most $2D$ from each vertex in their respective optimal cluster, so there are also at least $k-1$ vertices of distance at most $2D$ to cluster with them; the reassignment procedure described by the max-flow can hence successfully build a k-cluster.

If, for some D, the above described procedure successfully builds a partition $P'(p_1), \ldots, P'(p_r), P'(z_1), \ldots, P'(z_q)$, it is again clear that these sets are a k-cluster for V. We claim that this solution has a maximum radius less than or equal to $3D$. To see this, first consider a set $P'(z_i)$ for some $i \in \{1, \ldots, q\}$. All vertices added in step (b) or by the max-flow reassignment are picked because they are at distance most $2D$ from z_i. For any vertex $w \in P_D$ added to $P(z_i)$ in step (c), we know that there exists a vertex v_w which was placed in $P(z_i)$ in step (b) and hence has distance at most $2D$ from z_i and has, by the choice in step (c), distance at most D from w. Since $z_i \notin P_D$, it follows that $d(z_i, w) \leq 3D$. For a set $P'(p_i)$ with $i \in \{1, \ldots, r\}$, all vertices added by step (a) or by the max-flow reassignment have distance at most D from p_i. Let $w_1, \ldots, w_h$ be the vertices added to $P(p_i)$ in step (c). Further, let v_{w_j} be a vertex in $P(p_i) \setminus P_D$ at distance at most D from w_j. Since v_{w_1} is not in P_D it follows that $d(v, v_{w_1}) \leq d(v_{w_1}, p_i) + d(p_i, v) \leq D + D$ for every $v \in P'(p_i) \setminus \{w_1, \ldots, w_h\}$. Further, triangle inequality gives: $d(v_{w_1}, w_i) \leq d(v_{w_1}, v_{w_i}) + d(v_{w_i}, w_i) \leq 2D + D$ for every $i \in \{1, \ldots, h\}$. With central vertex v_{w_1}, $P'(p_i)$ hence has radius at most $3D$.

At last, the running time of this approximation algorithm is in $\mathcal{O}^*(2^p)$, as it only requires polynomial effort for each set $P' \subseteq P_D$ and $P_D \subseteq P$. □

5.2.2 Translation to Other Measures

With the simple observation that Equation 2 from Section 3.3 does not require the distance d to satisfy the triangle inequality, we can translate at least one of the parameterised approximations in the same way as we did in the metric case. Recall the statement for Equation 2 which holds for every choice of $f \in \{\text{rad,diam,avg}\}$:

$$\mathrm{opt}(G, d, f, \|\cdot\|_\infty^w, k) \geq k \cdot \mathrm{opt}(G, d, f, \|\cdot\|_\infty, k)\,.$$

Proposition 37

A 4-approximation for $(\|\cdot\|_\infty^w, \mathrm{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(B_p)$.

Proof. Let (G, d) be a given instance of $(\|\cdot\|_\infty^w, \mathrm{diam})$-$k$-CLUSTER. Observe that Corollary 1 also does not require the distance function d to satisfy the triangle inequality but holds in general. This allows us to first use the parameterised approximation from Theorem 35 to compute a k-cluster $\mathfrak{P}$ for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER on (G, d). As $\mathfrak{P}$ is a 2-approximation, it has a maximum diameter of at most $D := 2\mathrm{opt}(G, d, \mathrm{diam}, \|\cdot\|_\infty, k)$. Then, we to turn this partition into a k-cluster $\mathfrak{P}'$ for (G, d) of global cost at most D and with $|S| \leq 2k - 1$ for all $S \in \mathfrak{P}'$. Just like in Proposition 24, Equation 2 shows that the resulting k-cluster $\mathfrak{P}'$ is a 4-approximation for $(\|\cdot\|_\infty^w, \mathrm{diam})$-$k$-CLUSTER. □

For the radius measure, an equivalent to Proposition 37 is not clear. While the cluster cardinality for the diameter can be easily restricted by Corollary 1, an according result for radius does not hold in case of a non-metric distance measure. In particular, the idea of starting with a k-cluster $\mathfrak{P}$ for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER and then splitting up clusters of a large cardinality to obtain a solution which gives a provable approximation for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER with the help of Equation 2 is problematic in the case where $\mathfrak{P}$ contains a cluster S of large cardinality for which the set $C \setminus P$ is either empty or only contains the unique central vertex of C. As the procedure used for Theorem 36 does not allow a restriction to solutions for which such cases are avoided, it appears to be difficult to build an approximate solution for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER for non-metric instances with an idea similar to the one used in the metric case in Proposition 24. We can however at least translate the weaker result from Theorem 34 for radius, which yields:

Proposition 38

A 4-approximation for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(n^p)$.

Proof. For a given instance (G, d) of $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER with conflict vertex set $P \subseteq V$, we first build a k-cluster with the procedure described in Theorem 34. Denote by r^* the maximum radius of the optimum solution for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER; observe that r^* is not necessarily identical to $\mathrm{opt}(G, d, \|\cdot\|_\infty, \mathrm{rad}, k)$ but might be larger. If the choice of central vertices for P is optimal with respect to radius and the weighted infinity norm, the resulting k-cluster will have a maximum radius of at most $2r^*$. For the vertices in $V \setminus P$, the same strategy as in Proposition 24 can be used to split up clusters of cardinality more than $2k - 1$. Each set S in the resulting k-cluster hence satisfies one of the following conditions:

- S has a radius of at most $2r^*$ and a cardinality of at most $2k - 1$.
- S has a radius of at most $4r^*$ and a cardinality of k. This case occurs if S is the result of a splitting procedure as described in Proposition 24.
- S has a radius of at most r^* and contains at most one vertex from $V \setminus P$.

The last case is the only one where S can have a cardinality of more than $2k$, as the procedure from Proposition 24 allows us to split up such a cluster as soon as there is more than one vertex from $V \setminus P$ in S. For a choice of central vertices which is optimal with respect to the weighted infinity norm, this last case means that the cluster S is a subset of a cluster in an optimal solution for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER. Overall, this means that the resulting k-cluster has a global cost of at most $4 \cdot \mathrm{opt}(G, d, \|\cdot\|_\infty^w, \mathrm{rad}, k)$. □

A translation of Proposition 25 to non-metric instances is also difficult. First of all, the relation between diameter and average distortion used for this result does no longer hold if d violates the triangle inequality and unfortunately this property can not be saved by any reasonable pre-processing. It is however possible to relate $(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER to $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER to translate approximation results from there. This idea then runs exactly into the same problem of the impossibility to restrict cluster cardinalities as encountered above for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER and hence only gives a parameterised approximation in xp-time with respect to p.

Proposition 39

$(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER *is* $4k$*-approximable in* $\mathcal{O}^*(n^p)$.

Proof. Given an instance (G, d) of $(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER, first observe that the following relation holds, even if d violates the triangle inequality:

$$\mathrm{opt}(G, d, \mathrm{avg}, \|\cdot\|_\infty^w, k) \geq \mathrm{opt}(G, d, \mathrm{rad}, \|\cdot\|_\infty, k) \tag{13}$$

This is not hard to see, as for every non-empty set $P \subseteq V$ we can quite trivially bound its radius by the weighted average distortion, more precisely, with c chosen as some central vertex in P, it follows that

$$\begin{aligned} |P| \cdot \mathrm{avg}(P) = {} & \min\{\sum_{p \in P} d(c,p) \colon c \in P\} \\ \geq {} & \max\{d(c,v) \colon v \in P\} \\ \geq {} & \min\{\max\{d(u,v) \colon v \in P\} \colon u \in P\} \\ = {} & \mathrm{rad}(P) \end{aligned}$$

For an optimal k-cluster $\mathfrak{P}$ for $(\|\cdot\|_\infty, \mathrm{avg})$-$k$-CLUSTER we can hence conclude that the global cost $\max\{|P|\mathrm{avg}(P) \colon P \in \mathfrak{P}\}$ is at most the maximum radius of $\mathfrak{P}$, which yields Equation 13.

Now consider an approximation procedure very similar to the one described in Proposition 38 for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER with the only difference that we assume that the guessed central vertices are optimal for $(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER. Equation 13 is constructive in the sense that the optimal k-cluster for $(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER of global cost r^*, can be interpreted as a k-cluster for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER of global cost at most r^*. This means that the restriction of $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER to solutions for which an assignment of central vertices which is optimal with respect to $(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER still yields the opportunity to build a k-cluster of maximum radius at most r^*. The approximation procedure described in Theorem 34 run for the choice of central vertices which is optimal for $(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER hence yields a k-cluster $\mathfrak{P}$ of maximum radius at most $2r^*$ and can with the help of the procedure which splits up large clusters as described in the proof of Proposition 24 be turned into a k-cluster for which each set S satisfies one of the following properties:

- S has radius at most $2r^*$ and cardinality at most $2k-1$.
- S has radius at most $4r^*$ and cardinality k.
- S only contains vertices from P and their mutual fixed central vertex.

These properties hold, since each cluster $S \in \mathfrak{P}$ of cardinality more than $2k-1$ which contains a vertex in $V \setminus P$ which is not fixed to be the central vertex of S, can be split up by the procedure used in Proposition 24. If a cluster S of cardinality more than $2k-1$ remains, it only contains vertices in P and their correctly chosen central vertex. This means that S is a subset of an optimal cluster for $(\|\cdot\|_\infty, \text{avg})$-$k$-CLUSTER, and by the monotonicity of the associated global cost, we can conclude that the weighted average distortion $|S|\text{avg}(S)$ is bounded by $\text{opt}(G, d, \text{avg}, \|\cdot\|_\infty^w, k)$. Overall, the solution computed with this procedure yields a k-cluster of maximum weighted average distortion at most $4kr^*$. By the definition of r^*, this makes the described procedure a $4k$-approximation for $(\|\cdot\|_\infty^w, \text{avg})$-$k$-CLUSTER.

The complete approximation algorithm is given by the procedure given in Theorem 34 with additional split-up of clusters of cardinality more than $2k-1$ for which more than just the fixed central vertex is in $V \setminus P$ and obviously picking as output the choice of fixed centers for P which yields the solution of smallest global cost with respect to $(\|\cdot\|_\infty, \text{avg})$-$k$-CLUSTER. The running time of this approximation is dominated by the guessing of the central vertices which can be done in $\mathcal{O}^*(n^p)$. □

5.2.3 Resolving Conflicts in the Constraint Forest Approximations

In the following we will discuss the strategy of parameterisation by conflicts for the approximation procedures from Section 3 which are based on a reduction to constraint forest problems. While a first xp-time approximation can be obtained quite trivially, these algorithms have a very different structure which does not seem to allow an easy generalisation to non-metric instances.

Simple guessing for each $v \in V$ and each vertex $p \in P$ if they lie together in the same cluster in an optimal solution gives the complete sets which contain conflict vertices. The remaining vertices which are not assigned to clusters by this guessing are not involved in any conflict and can hence be partitioned using the approximation procedures for metric instances. This simple idea of parameterisation by conflicts translates the results from Section 3.2 to the non-metric case with a running time dominated by the number of possible guesses, which immediately gives the following result.

Corollary 40

$(\|\cdot\|_1^w, \text{avg})$-$k$-CLUSTER, $(\|\cdot\|_1^w, \text{diam})$-$k$-CLUSTER and $(\|\cdot\|_1^w, \text{rad})$-$k$-CLUSTER can be approximated with a running time in $\mathcal{O}^*(n^p)$ and ratio $2k$, $8k-1$ and $16k$, respectively.

Aside from this rather brute-force approach, it appears to be quite difficult to incorporate parameterisation into the approximation strategies based on tree partitioning. While the approximation for tree partitioning itself as given in [43] also works for non-metric distances, conflicts are very troublesome for the translation from a given tree partitioning to a k-cluster. More precisely, if the vertex c which we chose as the center to build a star from a given tree as in the proof of Theorem 21 is a conflict vertex, we can no longer assume any constant approximation ratio.

For the results based on Lower Capacitated Path Partitioning even this vague idea is no longer applicable as the 4-approximation used to find the path partitioning already requires an instance which satisfies the triangle inequality. The idea to also use tree partitioning, as indicated in Remark 4, is not enough to resolve this, since it has the same problem as an attempt to derive an approximation for path partitioning from tree partitioning in case of non-metric instances; in short, it is not trivial to transform the tree into a path without uncontrollable blow-up of edge-costs. It seems therefore that an approximation for $(\|\cdot\|_1^w, \mathrm{rad})$-$k$-cluster and $(\|\cdot\|_1^w, \mathrm{diam})$-$k$-cluster for non-metric instances with parameterisation by conflicts definitely requires a different strategy for the metric case first.

5.2.4 Summary of Parameterisation by Conflicts

Without even a known approximation for $(\|\cdot\|_\infty, \mathrm{avg})$-$k$-cluster on metric instances to start from, it appears pointless to try to come up with a useful parameterisation by conflicts for this problem variant. Therefore we end up with the results summarised in Table 4.

The increase in approximation ratio from 2 to 3 for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-cluster that comes with the change from only requiring xp-time to the more efficient fpt-time, is quite peculiar. The lower bounds discussed in Section 5.3.3 will at least give a partial explanation for this. Now, in the following section, we will first try to improve the worst-case running time without worsening the approximation ratio.

5.3 Structural Parameters of the Conflict Graph

One possibility to speed up the parameterised approximation algorithms presented so far is to choose a smaller parameter. In these procedures, we technically do not always consider the whole set P but just the subset P_D of vertices which are involved in a conflict of a certain type for each fixed value D. For the diameter-measure, for example, we only consider vertices which are involved in a conflict in the set:

$$C_D := \{\{u, v\} \in C \colon d(u, v) > 2D, \exists\, x \in V \colon\; d(u, x), d(v, x) \leq D\}\,.$$

	rad		diam	avg
$\|\cdot\|_\infty$	2 $\mathcal{O}^*(n^p)$ (Theorem 34)	3 $\mathcal{O}^*(2^p)$ (Theorem 36)	2 $\mathcal{O}^*(B_p)$ (Theorem 35)	?
$\|\cdot\|_\infty^w$	4 $\mathcal{O}^*(n^p)$ (Proposition 38)		4 $\mathcal{O}^*(B_p)$ (Proposition 37)	$4k$ $\mathcal{O}^*(n^p)$ (Proposition 39)
$\|\cdot\|_1^w$	$16(k-1)$ $\mathcal{O}^*(n^p)$ (Corollary 40)		$8(k-1)$ $\mathcal{O}^*(n^p)$ (Corollary 40)	$2k$ $\mathcal{O}^*(n^p)$ (Corollary 40)

Table 4: Summary of the parameterised approximation results, ratio and asymptotic running time with respect to the number of conflict vertices p.

These types of conflicts are the only ones that would lead to a solution of diameter larger than $2D$ in the greedy procedure described for Theorem 20. Generally, it is difficult to say how much the cardinality of P_D differs from the number of vertices involved in any (small) conflict, in the worst case, it is of course possible that all conflicts are of the type described in C_D and hence P_D is equal to P.

Designing approximation algorithms in this way with a parameter which is provably smaller than P is hence only possible by either considering a conflict set provably smaller than C or a set of conflict vertices provably smaller than P. This leads to two different approaches to decrease the size of the set of vertices which require exponential effort.

A conflict in C in the approximation algorithms for metric distances from Section 3 increases the approximation ratio proportionally to the magnitude of the violation. Considering the notion of α-relaxed triangle inequality introduced in Section 5.1, it seems that smaller values for α do not affect the quality of the derived approximations too much. We will discuss the idea to ignore such in a sense less serious conflicts to speed up approximation procedures for the price of a controllable increase of the performance guarantee in Section 5.5.

In this section, we want to focus on strategies to only spend exponential time for vertices in a subset of P.

5.3.1 Vertex Cover

Looking closer at the problems caused by the conflicts C_D in the polynomial approximation algorithms, it is not necessary to consider all vertices in P but it appears to be sufficient to pick a subset of P_D which covers all conflicts in C_D. Formally, this idea translates into parameterisation by a vertex cover for the subgraph of G, given by $G_c = (P, C)$. In the following, we will use p_c to denote the size of a minimum vertex cover for G_c and discuss parameterised approximation with respect to this parameter. Again, as a first easy observation, an xp-time result for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER with this parameterisation is pretty easy to see. In fact, we can simply switch from the set P to a minimum vertex cover for G_c, which yields the following result.

Theorem 41

A 2-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(n^{p_c})$.

Proof. Consider the procedure Theorem 34 with the only alteration that the set P is replaced by a subset P_c of P which is a minimum vertex cover for G_c. The arguments given for Theorem 34 remain exactly the same:

- Vertices in Z are central by the correct choice and according vertices to build clusters of cardinality at least k and radius at most D hence exist.
- Vertices in P_c are assigned optimally by the correct choice.
- Vertices in $V \setminus P_c$ are not in conflict as P_c is a vertex cover for G_c. Triangle inequality hence holds for all pairs of vertices in $V \setminus P_c$ and all arguments used in the original approximation from Theorem 20 remain true.

This means that the greedy procedure is successful for the correct choice of central vertices for each $v \in P_c$ and for D chosen as twice the optimum value for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER on (G, d).

The asymptotic running time is dominated by guessing the correct central vertices for the clusters containing vertices in P_c which can be done in $\mathcal{O}^*(n^{p_c})$; observe that a minimum vertex cover for G_c only has to be computed once in the beginning and this can be done in $\mathcal{O}^*(1.2738^{p_c})$ by [19]. □

The fpt-time parameterised approximations from Section 5.2.1 also require little algorithmic adjustment to switch from parameter p to parameter p_c. Proving the correctness of the given procedures, i.e., guaranteeing a performance ratio, is however more complicated.

Theorem 42

A 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(B_{p_c} + 1.1996^p)$.

Proof. Let d be the distance function induced by the input graph $G = (V, E)$. Just like in the procedure described for Theorem 35 we consider for fixed $D \in \{d(v, w) \colon v, w \in V\}$ the subset of conflicts:

$$C_D := \{\{u, v\} \in C \colon d(u, v) > 2D, \exists\, x \in V \colon\ d(u, x), d(v, x) \leq D\}\,.$$

and the corresponding subset of P:

$$P_D = \bigcup_{\{u,v\} \in C_D} \{u, v\}\,.$$

Let $G_c^D = (P_D, C_D)$ be the conflict graph with respect to D and let $\mathcal{V}_D$ be a minimum vertex cover for G_c^D. As G_c^D is a subgraph of G_c, the cardinality of $\mathcal{V}_D$ is at most p_c.

We use almost the same procedure as described for Theorem 35. The main difference is that we only iterate over partitions of $\mathcal{V}_D$ instead of the whole set P_D, so, let $C^1, \ldots, C^r$ be a partition of $\mathcal{V}_D$ with $C^i \neq \emptyset$ and $\mathrm{diam}(C^i) \leq D$ for all $i \in \{1, \ldots, r\}$. In step (a) of the greedy pre-clustering process, we no longer pick a representative c_i but now, in a sense, always consider the whole subset C^i as center:

(a) Iteratively, for $i = 1, \ldots, r$, create a cluster (for the sake of simplicity still named) $P(c_i)$ by adding to C^i all at this point unclustered vertices v in $V \setminus P_D$ with $d(u, v) \leq D$ for all $u \in C^i$.

Also, the sets $S_c(i, j)$ are similarly now defined with respect to the whole set C^i. Further, we now only fix the vertices in $\mathcal{V}_D$ and not the whole set P_D, which gives the following different definitions of the sets $S_c(i, j)$ and $S(i, j)$:

- $S_c(i, j) := \{v \in P(c_j) \setminus \mathcal{V}_D \colon d(u, v) \leq D\ \ \forall\, u \in C^i\}$ for all $1 \leq j < i \leq r$,
- $S(i, j) := \{v \in P(c_j) \setminus \mathcal{V}_D \colon d(v, z_i) \leq D\}$ for all $1 \leq j \leq r$ and $1 \leq i \leq q$.

Other than these adjustments, we use the same procedure as for Theorem 35 and claim that it also produces a 2-approximation for D chosen as the optimum value $\mathrm{opt}(G, d, \|\cdot\|_\infty, \mathrm{diam}, k)$.

Let $S_1, \ldots, S_y$ be an optimal solution for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER on instance (G, k). Consider running the above procedure for D chosen as the optimum value $\mathrm{opt}(G, d, \|\cdot\|_\infty, \mathrm{diam}, k)$ and the partition $C^1, \ldots, C^r$ for $\mathcal{V}_D$ given by $C^i = \mathcal{V}_D \cap S_{j_i^c}$ with $\{j_1^c, \ldots, j_r^c\} = \{i \colon 1 \leq i \leq y, \mathcal{V}_D \cap S_i \neq \emptyset\}$ such that $j_1^c < j_2^c < \cdots < j_r^c$. Very similar to the proof of Theorem 35, it can be shown that the clustering procedure successfully builds a k-cluster $P'(c_1), \ldots, P'(c_r), P'(z_1), \ldots, P'(z_q)$ for this choice of D and $C^1, \ldots, C^r$: By definition, for each $1 \leq i \leq r$, the set C^i is included in a different optimal cluster $S_{j_i^c}$. For the vertices $z_1, \ldots, z_q$ chosen in step (b), we know that $d(z_i, u) > D$ for at least one vertex $u \in C^i$ and $d(z_i, z_l) > D$ for each $1 \leq i < l \leq q$ and all

$1 \leq j \leq r$, so each z_i belongs to a distinct cluster $S_{j_i^z}$ of the chosen optimal solution with $j_i^z \notin \{j_1^c, \ldots, j_r^c\}$ and $j_i^z \neq j_l^z$ for all $1 \leq i < l \leq q$. So there exist at least the k vertices from $S_{j_i^c}$ at distance at most D from all vertices in C^i to be assigned to $P(c_i)$ for each $1 \leq i \leq r$ and, similarly, at least the k vertices from $S_{j_i^z}$ at distance at most D from z_i to be assigned to $P(z_i)$ for each $1 \leq i \leq q$. The max-flow procedure can hence successfully build a k-cluster.

It is again clear that any successful run of the above procedure yields a k-cluster for V. For the approximation ratio, we have to be more careful than in Theorem 35. Let $P'(c_1), \ldots, P'(c_r), P'(z_1), \ldots, P'(z_q)$ be a solution computed for some value D:

- Each vertex $v \in P'(c_i)$ for some $i \in \{1, \ldots, r\}$ is from C^i or was either assigned to $P(c_i)$ in step (a) or moved by the max-flow procedure. In the latter two cases, v is not in $\mathcal{V}_D$ and was included in $P'(c_i)$ because $d(u, v) \leq D$ holds for all $u \in C^i$.
  - For $v, w \in C^i$ it follows that $d(v, w) \leq D$ as the set C^i has diameter at most D by choice.
  - For $w \in C^i$ and $v \in P'(c_i) \setminus C^i$, we know that $d(v, w) \leq D$.
  - For $v, w \in P'(c_i) \setminus C^i$, we know that $v, w \notin \mathcal{V}_D$, so $\{v, w\}$ is not an edge in G_c^D as it would otherwise not be covered by $\mathcal{V}_D$. As $C^i \neq \emptyset$, there exists at least one vertex $u \in C^i$ and for this vertex u, we know that $d(u, v) \leq D$ and $d(u, w) \leq D$. By the definition of the edge set C_D this means (with shortcut $x = u$) that $d(v, w) \leq 2D$.
- Each vertex $v \in P'(z_i)$ for some $i \in \{1, \ldots, q\}$ was either assigned to $P(z_i)$ in step (b) or moved by the max-flow procedure, in both cases because $d(v, z_i) \leq D$ holds. Further, $P'(z_i)$ contains no vertices from $\mathcal{V}_D$, so for any two vertices $v, w \in P'(z_i)$, $\{v, w\}$ is not an edge in G_c^D. With shortcut $x = z_i$ in the definition of the edge set C_D, this implies $d(v, w) \leq 2D$ for all $v, w \in P'(z_i)$.

As the procedure is successful for the optimum diameter chosen as D, the k-cluster $P'(c_1), \ldots, P'(c_r), P'(z_1), \ldots, P'(z_q)$ produced for this D hence is a 2-approximation for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER on G.

A minimum vertex cover for G_c^D can be computed in $\mathcal{O}^*(1.1996^{|P_D|})$ by [65]. Together with the upper bound of $B_{|\mathcal{V}_D|} \leq B_{p_c}$ on the number of partitions that have to be checked, this gives the claimed running time. □

Alternatively to an exact algorithm to solve MINIMUM VERTEX COVER on G_c^D, we could also use a parameterised algorithm to arrive at a parameterisation only by p_c. MINIMUM VERTEX COVER with standard parameterisation can be solved in $\mathcal{O}^*(1.2738^{p_c})$ by [19] which gives the following result:

Corollary 43

A 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(B_{p_c})$.

For $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER, it is not obvious how to reduce the parameter in Theorem 36 to a structural parameter of the input graph. With an algorithm which additionally guesses a partition, like for the diameter measure, it is possible to find an approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER parameterised by the minimum vertex cover of the conflict graph, leading to the following.

Theorem 44

A 3-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(2^{p_c} \cdot B_{p_c} + 1.1996^p)$.

Proof. We modify the algorithm presented for Theorem 36, so let for an input graph $G = (V, E)$ with distances d, D be as defined there and we use the same subset of conflict vertices given by:

$$P_D := \{u \colon \exists\, v, x \in V \colon (\{u, v\} \in C) \wedge (d(u, x), d(v, x) \leq 2D)\}\,.$$

Let C_D be the corresponding set of conflicts, i.e.:

$$C_D := \{\{u, v\} \colon \exists\, x \in V \colon d(u, x), d(v, x) \leq 2D\}\,.$$

Let $\mathcal{V}_D$ be a minimum vertex cover for $G_c^D := (P_D, C_D)$. In addition to guessing which vertices in this set $\mathcal{V}_D$ are central in an optimal solution, we also guess their partition as in the procedures described for the diameter measure. These two guesses should be consistent with a solution of maximum radius D, so we only consider a subset $P' \subseteq \mathcal{V}_D$ of central vertices together with a partition $W_1, \ldots, W_r$ of $\mathcal{V}_D$ if for all $i \in \{1, \ldots, r\}$ the following two properties hold:

1. $|W_i \cap P'| \leq 1$ and
2. if $W_i \cap P' \neq \emptyset$ then $w \in W_i \cap P'$ satisfies $d(w, v) \leq D$ for all $v \in W_i$.

Let $P' \subseteq \mathcal{V}_D$ and $W_1, \ldots, W_r$ be a partition of $\mathcal{V}_D$ with these properties. For every such set, we first compute a central vertex w_i for each W_i, $i \in \{1, \ldots, r\}$. If $W_i \cap P' \neq \emptyset$, we set w_i to be the only vertex in this non-empty intersection. Let, w.l.o.g., $\{1, \ldots, h\}$ be the subset of indices j from $\{1, \ldots, r\}$ for which $W_j \cap P' = \emptyset$. In order to find a central vertex for each W_j, $j \in \{1, \ldots, h\}$, we compute for each $j \in \{1, \ldots, h\}$ the sets $Q_j = \{v \in V \setminus \{\mathcal{V}_D\} \colon d(v, w) \leq D \text{ for all } w \in W_j\}$ of potential central vertices. As these sets do not have to be disjoint, we use maximum matching, which can be computed in $O(n^3)$ for a graph with n vertices (see [31]), on the graph $\bar{G} := (\bar{V}, \bar{E})$ with $\bar{V} := \{u_1, \ldots, u_h\} \cup \bigcup_{j=1}^{h} Q_j$ and $\bar{E} := \bigcup_{j=1}^{h} \{\{u_j, v\} \colon v \in Q_j\}$ to try to assign one central vertex to each set W_j. If no matching of cardinality h exists for $\bar{G}$, we abort the algorithm for this choice of P' and $W_1, \ldots, W_r$. Otherwise, we assign the central vertex w_j for each $j \in \{1, \ldots, h\}$ to be the vertex from C^j which is matched with u_j in the maximum matching for $\bar{G}$. With these central vertices, build a pre-clustering with the following steps:

(a) For $i = 1, \ldots, h$, build a cluster $P(w_i)$ including W_i and all at this point unclustered vertices v from $V \setminus \mathcal{V}_D$ with $d(v, w_i) \leq 3D$.

(b) For $i = h+1, \ldots, r$, build a cluster $P(w_i)$ including W_i and all at this point unclustered vertices v from $V \setminus \mathcal{V}_D$ with $d(v, w_i) \leq D$.

(c) While there are still unclustered vertices, pick an arbitrary unclustered vertex z and build a cluster $P(z)$ including z and all unclustered vertices v with $d(v, z) \leq 2D$.

Let, as usual, $z_1, \ldots, z_q$ be the vertices chosen in step (c) to build clusters $P(z_1), \ldots, P(z_t)$ by the above procedure. For the reassignment which balances cardinalities, we aim to keep a maximum radius of $3D$ for $P(w_1), \ldots, P(w_h)$, a maximum radius of D for $P(w_{h+1}), \ldots, P(w_r)$ and a maximum radius of $2D$ for $P(z_1), \ldots, P(z_q)$, which yields the following types of vertices which are allowed to be moved:

- $S_p(i,j) := \{v \in P(w_j) \setminus (\mathcal{V}_D \cup \{w_j\}) \colon d(v, w_i) \leq 3D\}$ for all $1 \leq j < i \leq h$
- $S_{\bar{p}}(i,j) := \{v \in P(w_j) \setminus (\mathcal{V}_D \cup \{w_j\}) \colon d(v, w_i) \leq D\}$ for all $1 \leq j < i \leq r$, $h < i$.
- $S(i,j) := \{v \in P(w_j) \setminus \mathcal{V}_D \colon d(v, z_i) \leq 2D\}$ for all $1 \leq j \leq r$ and $1 \leq i \leq q$ and
- $S_z(i,j) := \{v \in P(z_j) \setminus \{z_j\} \colon d(v, z_i) \leq 2D\}$ for all $1 \leq j < i \leq q$.

If it is possible to move vertices according to these sets such that each cluster contains at least k vertices, such a reassignment can be found with a max-flow in a network designed very similar to the one for Theorem 35. In case of a successful run of the described algorithm, the resulting partition obviously is a k-cluster for V of maximum radius at most $3D$.

We claim that for $D = r^*$, there exists a subset $P' \subseteq \mathcal{V}_D$ and a partition $W_1, \ldots, W_r$ such that the above clustering procedure is successful. Let $\{S_1, \ldots, S_y\}$ be any optimal solution for $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER on input $G = (V, E)$ with distance d. Fix some central vertex s_i of S_i for each $i \in \{1, \ldots, y\}$. Consider running the described greedy procedure for $D = r^*$, the subset $P' = \mathcal{V}_D \cap \{s_1, \ldots, s_y\}$ and the partition $W_1, \ldots, W_r$ with $W_i = S_{j_i} \cap \mathcal{V}_D$, where $\{j_1, \ldots, j_r\}$ is the set of indices j in $\{1, \ldots, y\}$ with $S_j \cap \mathcal{V}_D \neq \emptyset$. As, by definition, $W_i \cap P' \subset \{s_{j_i}\}$ for each $i \in \{1, \ldots, r\}$, this choice of P' and $W_1, \ldots, W_r$ has properties 1 and 2 and is hence valid for the algorithm.

If $\{1,\dots,h\}$ is the set of indices $i \in \{1,\dots,r\}$ for which $W_i \cap P' = \emptyset$, we know by definition that the vertex $s_{j_i} \notin W_i$ is a valid center for the cluster which includes W_i, so $s_{j_i} \in Q_i$. The graph $\bar{G}$ hence has a maximum matching of cardinality h, as the set $\{\{u_i, s_{j_i}\}\colon 1 \leq i \leq h\}$ is such a matching. The algorithm hence successfully assigns a central vertex w_i to a set W_i for each $i \in \{1,\dots,r\}$.

The sets $W_1,\dots,W_r$ are by definition from different sets in the fixed optimal solution. For any z chosen in step (c) we know that $d(w_i, z) > 3D$ for all $i \in \{1,\dots,h\}$. We claim that this implies that z is not in S_{j_i} for any $i \in \{1,\dots,r\}$. Assume on the contrary that $z \in S_{j_i}$, which means that $d(z, s_{j_i}) \leq D$. If $i > h$, z can not be unclustered in step (c), as $s_{j_i} = w_i$ and step (b) for i would, in case z was not clustered even earlier, put z into $P(w_i)$. If $i < h$, then $s_{j_i} \notin \mathcal{V}_D$ and also $w_i \notin \mathcal{V}_D$ by definition, so $\{w_i, s_{j_i}\}$ is not a conflict in C_D. As for all vertices v in the nonempty set W_i we know that $d(s_{j_i}, v) \leq D$ and $d(w_i, v) \leq D$, it follows that $d(s_{j_i}, w_i) \leq 2D$. As z is also not included in the vertex cover of the conflict graph, the set $\{w_i, z\}$ is not a conflict which implies $d(w_i, z) \leq d(s_{j_i}, w_i) + d(s_{j_i}, z) \leq 3D$, so, unless z is already clustered even earlier, step (a) would move z into $P(w_i)$. Two vertices $z, \bar{z}$ with $z \neq \bar{z}$ chosen in step (c) belong to different clusters S_j, $j \in \{1,\dots,q\}$, as $d(z, \bar{z}) > 2D$ and $z, \bar{z} \notin \mathcal{V}_D$, so $\{z, \bar{z}\}$ is not a conflict, hence there can not be an index $j \in \{1,\dots,y\}$ such that $d(s_j, z) \leq D$ and $d(s_j, \bar{z}') \leq D$. So, each vertex z_i chosen in step (c) belongs to a different cluster $S_{j_i^z}$, $i \in \{1,\dots,q\}$ with $j_i^z \notin \{j_1,\dots,j_r\}$. The algorithm can successfully assign k vertices to each cluster since:

- For the clusters $P(w_i)$ with $i \leq h$, we already showed the relation $d(s_{j_i}, w_i) \leq 2D$. For all vertices $v \in S_{j_i} \setminus \mathcal{V}_D = S_{j_i} \setminus W_i$ it follows that $d(v, w_i) \leq d(v, s_{j_i}) + d(s_{j_i}, w_i) \leq 3D$, as $v, w_i, s_{j_i} \notin \mathcal{V}_D$, so no two of these vertices build a conflict. Consequently, there are at least the vertices from the set S_{j_i} (which has cardinality at least k) at distance at most $3D$ from w_i to move into $P(w_i)$.

- For the clusters $P(w_i)$ with $i > h$, the algorithm knows the correct center $w_i = s_{j_i}$ and can find all vertices in S_{j_i} at distance at most D from w_i. So there are at least the vertices from S_{j_i} at distance at most D from w_i to move into $P(w_i)$.

- For the clusters $P(z_i)$, $i \in \{1,\dots,q\}$, we know that z_i belongs to a cluster $S_{j_i^z}$ which does not contain vertices from $\mathcal{V}_D$. As z_i is also not in $\mathcal{V}_D$, $\{v, z_i\}$ is not a conflict for each $v \in S_{j_i^x}$, so $d(v, z_i) \leq d(v, s_{j_i^z}) + d(s_{j_i^z}, z_i) \leq D + D$. So there are at least the vertices from $S_{j_i^z}$ at distance at most $2D$ from z_i to move into $P(z_i)$.

Overall, this shows that the described algorithm is successful for D chosen as the optimum value $\operatorname{opt}(G, d, \|\cdot\|_\infty, \mathrm{rad}, k)$, and produces a k-cluster of maximum radius $3D$. A minimum vertex cover for G_c^D can be computed in

$\mathcal{O}^*(1.1996^{|P_D|})$ by [65]. The running time of the remaining algorithm is dominated by guessing the subset P' and the partition $W_1, \dots, W_r$ of the vertex cover $\mathcal{V}_D$. As G_c^D is a subgraph of G_c, the vertex cover $\mathcal{V}_D$ has a cardinality of at most p_c, so there are at most $\mathcal{O}^*(2^{p_c} \cdot B_{p_c})$ possibilities to check for P' and $W_1, \dots, W_r$, which yields the claimed overall running time. □

Just like for the diameter measure, we can estimate the running time solely by the vertex cover number of the conflict graph which yields:

Corollary 45

A 3-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(2^{p_c} \cdot B_{p_c})$.

5.3.2 $\mathcal{P}_3$-Covers

With more changes to the algorithm introduced in Theorem 43, it is possible to further reduce the size of the subset of P_D which requires the expensive guessing of the partition. When building the greedy pre-clustering, it is always possible to correctly assign conflict vertices to a set by branching on the conflicts to decide which vertex has to be excluded. This way it is possible to find a correct choice of central vertices in step (*b*). The network used to model vertex-reassignment can be altered to prevent two conflict vertices to move into the same cluster, by routing their flow through an additional network-vertex with a capacity of only 1 to move into a cluster. If the conflicts are isolated, an additional network-vertex for each conflict can be used to correctly model all conflict-free reassignments. We can of course not assume that the conflicts are pairwise disjoint, but we can fix the partition of a subset of conflict vertices, as we did for the vertex cover of the conflict graph, and use the above ideas for the remaining vertices which induce a graph with isolated conflicts. The set of vertices which have to be removed in order to arrive at a graph with isolated conflicts is smaller than the vertex cover of the conflict graph (unless the distance is a metric). Formally, the smallest set for which this property holds is called a $\mathcal{P}_3$-*cover* of the conflict graph. The corresponding problem of finding a smallest $\mathcal{P}_3$-cover of a given graph is formally defined by:

$\mathcal{P}_3$-COVER

Input: Graph $G = (V, E)$, $\ell \in \mathbb{N}$.

Parameter: ℓ

Question: Does there exists a subset $\mathcal{F} \subseteq V$ such that the degree of each vertex in $G[V \setminus \mathcal{F}]$ is at most 1?

$\mathcal{P}_3$-COVER can be solved in $\mathcal{O}^*(1.3659^n)$ time and space or in $\mathcal{O}^*(1.4656^n)$ time and polynomial space, see [51]. Using the more expensive previously used strategy of guessing the correct partition only for a minimum $\mathcal{P}_3$-cover of the conflict graph, branching on the remaining isolated conflicts for the pre-clustering and modifying the network to avoid conflicts as described above gives the following result, where we in the following always use p_{3c} to denote the cardinality of a minimum $\mathcal{P}_3$-cover for G_c.

Theorem 46

A 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(\sqrt{2}^p \cdot B_{p_{3c}} + 1.4656^p)$.

Proof. Let d be the distance function induced by the input graph $G = (V, E)$. As before, we use binary search to find the smallest value D for which we can construct a k-cluster with the procedure described in the following. So, let $D \in \{d(u, v)\colon u, v \in V\}$ be fixed and let G_c^D be defined as in the proof of Theorem 42. Let $\mathcal{F}_D$ be a minimum $\mathcal{P}_3$ cover for G_c^D.

The idea now for the greedy pre-clustering is to only guess the partition for the set $\mathcal{F}_D$, so we consider all partitions $F_1, \ldots, F_r$ of $\mathcal{F}_D$ with $F_i \neq \emptyset$ and $\mathrm{diam}(F_i) \leq D$ for all $i \in \{1, \ldots, r\}$. Let $e_1, \ldots, e_b$ be the edges remaining in G_c^D after removing the vertices in $\mathcal{F}_D$. Since $\mathcal{F}_D$ is a $\mathcal{P}_3$-cover, it follows that $e_i \cap e_j = \emptyset$ for all $i, j \in \{1, \ldots, b\}$ with $i \neq j$. These edges are the only conflicts from C_D which remain in the set $V \setminus \mathcal{F}_D$. When building the pre-clustering, which we want to construct such that each set has a diameter of at most $2D$, we now have to be careful not to include both vertices of an edge e_i in the same set. We therefore guess for each edge e_i which of its adjacent vertices is not included in the set in question[6]. Denote for this purpose the two vertices adjacent to e_ℓ by u_ℓ^0 and u_ℓ^1 for each $\ell \in \{1, \ldots, b\}$. As we only want to guess once in the beginning, we fix an arbitrary ordering on the vertices in V to make our pre-clustering algorithm, in a way, deterministic. For each partition $F_1, \ldots, F_r$ of $\mathcal{F}_D$ and each vector $g \in \{0, 1\}^b$, we build a pre-clustering in the following way:

(*a*) Iteratively, for $i = 1, \ldots, r$ do the following:

- Create a cluster $P(f_i)$ and add all vertices from F_i to it.
- Collect in the set R_i all at this point unclustered vertices v in $V \setminus \mathcal{F}_D$ with $d(w, v) \leq D$ for all $w \in F_i$.
- While there exists an index $\ell \in \{1, \ldots, b\}$ with $e_\ell \subseteq R_i$, remove the vertex $u_\ell^{g[\ell]}$ from R_i.
- Add R_i to $P(f_i)$.

(*b*) If all clusters $P(f_i)$ are built but $\bigcup_{i=1}^r P(f_i) \neq V$, then repeat the following until all vertices are clustered:

[6] See Remark 7 for a different view on this type of guessing.

- Pick the unclustered vertex z of smallest index in the fixed order on V and create a new cluster $P(z) = \{z\}$.
- Collect in the set R_z all at this point unclustered vertices v with $d(z, v) \leq D$.
- While there exists an index $\ell \in \{1, \ldots, b\}$ with $e_\ell \subseteq R_z$, remove the vertex $u_\ell^{g[\ell]}$ from R_z.
- Add R_z to $P(z)$.

Let $z_1, \ldots, z_q$ be the vertices chosen in step (b). In order to turn the preclustering into a k-cluster, define the sets of vertices which can be moved similar as for Theorem 42, so with the notation here and the $\mathcal{P}_3$-cover $\mathcal{F}_D$ instead of the vertex cover $\mathcal{V}_D$, given by:

- $S_f(i, j) := \{v \in P(f_j) \setminus \mathcal{F}_D \colon d(u, v) \leq D \ \ \forall\, u \in F_i\}$ for all $1 \leq j < i \leq r$,
- $S(i, j) := \{v \in P(f_j) \setminus \mathcal{F}_D \colon d(v, z_i) \leq D\}$ for all $1 \leq j \leq r$ and $1 \leq i \leq q$, and
- $S_z(i, j) := \{v \in P(z_j) \setminus \{z_j\} \colon d(v, z_i) \leq D\}$ for all $1 \leq j < i \leq q$

Again, collect all vertices which can be moved in the set

$$\mathcal{S} := (\bigcup_{j=1}^{r-1} \bigcup_{i=j+1}^{r} S_f(i, j)) \cup (\bigcup_{j=1}^{r} \bigcup_{i=1}^{q} S(i, j)) \cup (\bigcup_{j=1}^{q-1} \bigcup_{i=j+1}^{q} S_z(i, j))\,.$$

When reassigning vertices from these sets, we now have to be careful not to move a conflict-pair in the same cluster; observe that $V \setminus \mathcal{F}_D$ now may still contain vertices involved in a conflict namely exactly the vertices adjacent to the edges $e_1, \ldots, e_b$. We alter the network to make sure that no two conflict vertices end up in the same cluster. Start with the network introduced in the proof of Theorem 35, where now we rename c_i' to f_i' as representative for $P(f_i)$ for all $i \in \{1, \ldots, r\}$. Remove in this network all arcs from network-vertices v' representing a vertex $v \in V \setminus \mathcal{F}_D$ which is included in a conflict-edge e_ℓ for some $\ell \in \{1, \ldots, b\}$. We add the following nodes and arcs to the network:

- For each $\ell \in \{1, \ldots, b\}$, we introduce a network-vertex e_ℓ'.
- For each vertex $v \in \mathcal{S}$ with $v \in e_\ell$ for some $\ell \in \{1, \ldots, b\}$ add the following arcs, each of capacity 1:
  - (v', e_ℓ'),
  - (v', w') for $e_\ell = \{v, w\}$ if there exist $i \in \{2, \ldots, r\}$, $j \in \{1, \ldots, i-1\}$ such that $v \in S_c(j, i)$ and $w \in P(f_i)$,

– (v', w') for $e_\ell = \{v, w\}$ if there exists an index $i \in \{1, \dots, r\}$ such that $v \in (\bigcup_{j=1}^{t} S(j,i)) \cup (\bigcup_{j=1}^{i-1} S_z(j,i))$ and $w \in P(z_i)$,

– (e'_ℓ, f'_i) for all $i \in \{1, \dots, r\}$ with $e_\ell \subseteq \bigcup_{j=1}^{i-1} S_c(j,i)$,

– (e'_ℓ, z'_i) for all $i \in \{1, \dots, q\}$ with $e_\ell \subseteq (\bigcup_{j=1}^{q} S(j,i)) \cup (\bigcup_{j=1}^{i-1} S_z(j,i))$,

– (v', f'_i) for all $i \in \{2, \dots, r\}$ and $v \in V$ such that $v \notin P(f_i)$ and $(P(f_i) \cup \bigcup_{j=1}^{i-1} S_c(j,i)) \cap e_\ell = \{v\}$,

– (v', z'_i) for all $i \in \{1, \dots, q\}$ and $v \in V$ such that $v \notin P(z_i)$ and $(P(z_i) \cup (\bigcup_{j=1}^{q} S(j,i)) \cup (\bigcup_{j=1}^{i-1} S_z(j,i))) \cap e_\ell = \{v\}$.

When reassigning vertices in $\mathcal{S}$ according to a max-flow for this network, we interpret a flow over an arc (v', w') with $v, w \in V$ as replacing v by w in the cluster that initially contains v. We will later show that these adjustments make sure that for a reassignment of vertices according to a network flow, no two vertices involved in a conflict e_ℓ end up in the same set. This is important to prove that a partition created from reassigning vertices in $\mathcal{S}$ creates a partition with maximum diameter at most $2D$. The maximum flow in this network is at most:

$$M := \sum_{i=2}^{r} \max\{0, k - |P(f_i)|\} + \sum_{j=1}^{q} \max\{0, k - |P(z_j)|\}\,.$$

If there exists a maximum flow of value M in the network, denote by $P'(f_1), \dots, P'(f_r)$, $P'(z_1), \dots, P'(z_q)$ the partition built from $P(f_1), \dots, P(f_r)$, $P(z_1), \dots, P(z_q)$ by reassigning vertices according to this flow. Obviously, this partition is a k-cluster, as a max-flow of value M balances the cardinalities such that each cluster contains at least k vertices. If the maximum flow in the network has a value less than M, abort and try a larger value for D.

We claim that the above described algorithm successfully computes a k-cluster of maximum diameter $2D$ for $D = \operatorname{opt}(G, d, \|\cdot\|_\infty, \operatorname{diam}, k)$.

Let $S_1, \dots, S_y$ be an optimal solution for $(\|\cdot\|_\infty, \operatorname{diam})$-$k$-CLUSTER on instance G. Consider running the above procedure for the optimum value D and the partition $F_1, \dots, F_r$ for $\mathcal{F}_D$ given by $F_i = \mathcal{F}_D \cap S_{j_i^f}$ with $\{j_1^f, \dots, j_r^f\} = \{i\colon 1 \le i \le y, \mathcal{F}_D \cap S_i \neq \emptyset\}$ such that $j_1^f < j_2^f < \dots < j_r^f$. By definition, for each $1 \le i \le r$, the set F_i is included in a different optimal cluster $S_{j_i^f}$. We claim that there exists a vector $g \in \{0,1\}^b$ such that the greedy pre-clustering chooses $z_1, \dots, z_q$ by resolving conflicts according to this vector g such that the following two properties hold:

1. $\{z_1, \dots, z_q\} \cap S_{j_i^f} = \emptyset$ for all $i \in \{1, \dots, r\}$, and

2. in the fixed optimal solution, no two vertices in $\{z_1, \ldots, z_q\}$ belong to the same cluster, formally: $|S_j \cap \{z_1, \ldots, z_q\}| \leq 1$ for all $j \in \{1, \ldots, y\}$.

For property 1, fix the following entries in g (recall the notation $e_\ell = \{u_\ell^0, u_\ell^1\}$): Let for each $\ell \in \{1, \ldots, b\}$, in case it exists, $i \in \{1, \ldots, r\}$ be the smallest index such that $d(w, u_\ell^0) \leq D$ and $d(w, u_\ell^1) \leq D$ for all $w \in F_i$ (these properties mean that $e_\ell \subseteq R_i$ when building $P(f_i)$ in step (a)). Assign:

$$g[\ell] = \begin{cases} 0 & \text{if } \ u_\ell^0 \notin S_{j_i^f} \\ 1 & \text{else .} \end{cases}$$

As e_ℓ is a conflict in C_D, we know that $d(u_\ell^0, u_\ell^1) > 2D$, so if $u_\ell^0 \in S_{j_i}^f$, it follows that $u_\ell^1 \notin S_{j_i}^f$. For a vector g with these entries, step (a) of the pre-clustering procedure does not put a vertex z in cluster $P(f_i)$ if either $d(z, w) > D$ for at least one vertex $w \in F_i$ (which implies $z \notin S_{j_i}^f$ as the optimal solution has maximum diameter D) or if z is a vertex in a conflict e_ℓ and $z \notin S_{j_i}^f$. Observe that each conflict e_ℓ only causes at most one decision in step (a) namely at the smallest index i for which both u_ℓ^0 and u_ℓ^1 are in R_i; afterwards one of the vertices involved is included in the set $P(f_i)$ and not considered in a successive step. Hence, after step (a), no unclustered vertex is contained in a set $S_{j_i}^f$, $i \in \{1, \ldots, r\}$, so especially $\{z_1, \ldots, z_q\} \cap S_{j_i^f} = \emptyset$ for all $i \in \{1, \ldots, r\}$.

For property 2, we show inductively that there is a vector g such that z_i is chosen appropriately for each i. Let R be the set of vertices remaining unclustered after step (a) for a vector g with entries according to the setting for property 1; observe that this set R only depends on the partition $F_1, \ldots, F_r$ and the entries for g which we have fixed for property 1. Step (b) then picks as vertex z_1 the vertex of smallest index in R (choice only depends on R). Let j_1^z be the index in $\{1, \ldots, y\}$ with $z_1 \in S_{j_1^z}$. For all $\ell \in \{1, \ldots, b\}$ with $e_\ell \subseteq R$ and $d(z_1, u_\ell^0) \leq D$ and $d(z_1, u_\ell^1) \leq D$, fix $g[\ell] = 0$ if $u_\ell^0 \notin S_{j_1^z}$ and $g[\ell] = 1$ otherwise. Since $e_\ell \subseteq R$, this conflict has not been considered so far and the entry for g is hence not fixed by a previous step. For g with these entries, the set $R \setminus P(z_1)$ only contains vertices w with either $d(w, z_1) > D$ or conflict vertices outside $S_{j_1^z}$, so $R \cap S_{j_1^z} \subseteq P(z_1)$, hence if $R \setminus P(z_1) \neq \emptyset$, step (b) chooses as z_2 a vertex that lies in an optimal cluster $S_{j_z^2}$ with $j_z^2 \notin \{j_1^z\} \cup \{j_1^f, \ldots, j_r^f\}$. Inductively, if $z_1, \ldots, z_h$ are picked with $z_i \in S_{j_i^z}$ and g fixed such that $R \cap S_{j_i^z} \subseteq \bigcup_{s=1}^i P(z_s)$ for all $i \in \{1, \ldots, r\}$ and $R^{h+1} := R \setminus (\bigcup_{i=1}^h P(z_i)) \neq \emptyset$, then z_{h+1} picked in step (b) satisfies $z_{h+1} \in S_{j_{h+1}^z}$ with $j_{h+1}^z \notin \{j_1^z, \ldots, j_h^z\} \cup \{j_1^f, \ldots, j_r^f\}$. With g such that $g[\ell] = 0$ if $u_\ell^0 \notin S_{j_{h+1}^z}$ and $g[\ell] = 1$ otherwise for all $\ell \in \{1, \ldots, b\}$ with $e_\ell \subseteq R^{h+1}$ and $d(z_{h+1}, u_\ell^0) \leq D$ and $d(z_{h+1}, u_\ell^1) \leq D$, it follows that $R^{h+1} \cap S_{j_{h+1}^z} \subseteq P(z_{h+1})$.

For at least one vector g, $z_1, \ldots, z_q$ are chosen such that each z_i belongs to a distinct cluster $S_{j_i^z}$ of the chosen optimal solution with $j_i^z \notin \{j_1^f, \ldots, j_r^f\}$ and $j_i^z \neq j_l^z$ for all $1 \leq i < l \leq q$. So there exist at least the k vertices from $S_{j_i^f}$

at distance at most D from all vertices in F_i to be assigned to $P(f_i)$ for each $1 \leq i \leq r$ and, similarly, at least the k vertices from $S_{j_i^z}$ at distance at most D from z_i to be assigned to $P(z_i)$ for each $1 \leq i \leq q$. For each conflict e_ℓ, no set S_j contains both vertices in e_ℓ, so the adjustments made to the network from Theorem 35 preventing both u_ℓ^0 and u_ℓ^1 to move to the same cluster do not affect the fact that there are at least k vertices which can be assigned to each set in the pre-clustering. The max-flow procedure can hence successfully build a k-cluster $P'(c_1), \ldots, P'(c_r), P'(z_1), \ldots, P'(z_q)$.

Any sets $P'(c_1), \ldots, P'(c_r), P'(z_1), \ldots, P'(z_q)$ produced by a successful run of the above procedure for some value D is obviously a k-cluster for V. We claim that the maximum radius of this solution is at most $2D$:

- Each vertex $v \in P'(f_i)$ for some $i \in \{1, \ldots, r\}$ is from F_i or was either assigned to $P(f_i)$ in step (a) or moved by the max-flow procedure. In the latter two cases, v is not in $\mathcal{F}_D$ and was included in $P'(f_i)$ because $d(v, w) \leq D$ holds for all $w \in F_i$.
  - For $v, w \in F_i$ it follows that $d(v, w) \leq D$ as the set F_i has diameter at most D by choice.
  - For $w \in F_i$ and $v \in P'(f_i) \setminus F_i$, then $d(v, w) \leq D$ holds as already mentioned above.
  - For $v, w \in P'(f_i) \setminus F_i$, we know that $v, w \notin \mathcal{F}_D$ and as $F_i \neq \emptyset$, there exists at least one vertex $u \in F_i$ and, by definition, $d(u, v) \leq D$ and $d(u, w) \leq D$. If $d(v, w) > 2D$, it follows that $\{v, w\} \in C_D$, which means that $\{v, w\} = e_\ell$ for some $\ell \in \{1, \ldots, b\}$. As step (a) only includes at most one vertex from each such conflict, at least v or w was added by the max-flow procedure, so assume w.l.o.g. $v \notin P(f_i)$. Since v is moved into $P'(f_j)$ by the max-flow, v is in $S_c(i, j)$ for some $j < i$. If $w \in P(f_i)$, this means that the network only contains the arc (v', w') to move v into $P'(f_i)$, which for a feasible flow however requires w to to be moved to a different cluster. If $w \notin P(f_i)$, then $w \in S_c(i, j')$ for some $j' < i$. Then $e_\ell \subseteq \bigcup_{j=1}^{i-1} S_c(i, j)$, hence the network only contains the arcs (v', e'_ℓ), (w', e'_ℓ) and (e'_ℓ, f'_i) to move v and w into $P'(f_i)$, the arc (e'_ℓ, r_i^f) however only has capacity 1 and can only move either v or w. Overall, this means that $d(v, w) > 2D$ is not possible.
- Each vertex $v \in P'(z_i)$ for some $i \in \{1, \ldots, q\}$ was either assigned to $P(z_i)$ in step (b) or moved by the max-flow procedure, in both cases because $d(v, z_i) \leq D$ holds. For two vertices $v, w \in P'(z_i) \setminus \{z_i\}$ it follows that if $d(v, w) > 2D \geq d(z_i, v) + d(z_i, w)$ then $\{v, w\} = e_\ell$ for some $\ell \in \{1, \ldots, b\}$. Similar to $v, w \in P'(f_i)$, this is not possible by the construction of the partition.

Overall, this shows that the procedure described above is an approximation algorithm with performance ratio 2 for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER.

For each D, a minimum $\mathcal{P}_3$-cover for G_c^D can be computed in $\mathcal{O}^*(1.4656^{|P_D|})$, so especially in $\mathcal{O}^*(1.4656^{|P|})$ as $P_D \subseteq P$ for all D. The number of edges remaining in the graph $G_c^D[V \setminus \mathcal{F}_D]$ is bounded by $\frac{1}{2}|P_D \setminus \mathcal{F}_D| \leq \frac{1}{2}|P|$, as $\mathcal{F}_D$ is a $\mathcal{P}_3$-cover which means that $G_c^D[V \setminus \mathcal{F}_D]$ only contains single edges and isolated vertices as connected components. The number of vectors g to be checked for each D is hence bounded by $2^{\frac{1}{2}|P|}$. The Bell-Number $B_{|\mathcal{F}_D|}$ bounds the number of partitions to be checked for each D. As G_c^D is a subgraph of G_c for each D, it follows that $|\mathcal{F}_D| \leq p_{3c}$. The algorithm described above considers at most $\log(|V|^2)$ values for D which yields an overall running time in $\mathcal{O}^*(1.4656^p + \sqrt{2}^p \cdot B_{p_{3c}})$. □

With the fpt-algorithm for $\mathcal{P}_3$-COVER from [64] which runs in $\mathcal{O}^*(1.7485^{p_{3c}})$, the worst-case running time of the algorithm presented for Theorem 46 can be estimated differently which does not give parameterisation solely by p_{3c} but still yields the following result:

Corollary 47

A 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(\sqrt{2}^p \cdot B_{p_{3c}})$.

Remark 7: The described guessing of a vector $g \in \{0,1\}^b$ with a fixed vertex-ordering is an iterative way to implement the intuitively described idea of branching on the conflicts. From a human perspective, it is probably easier to think of the pre-clustering as a procedure which greedily builds sets as in the original algorithm from Theorem 20 and, whenever this strategy tries to put two vertices which build a conflict into the same set, branches on which vertex can not stay in this set. We chose the described iterative approach instead for two reasons: Firstly, it is easier to maintain in the implementation, especially considering space-requirements; concretely, we found that the information that had to be stored for recursive calls would either require copies of large parts of the input, or an unnecessarily complicated back-tracking strategy. In comparison, the described iterative strategy always only needs one copy of the input and also had the advantage that it required very little changes to the original polynomial algorithm. Secondly, proving that there exists a successful run of the approximation procedure for the optimal value for D is cleaner with a concrete definition of the decisions g which correspond to a valid solution.

As already mentioned, branching on conflicts for the pre-clustering works for any set of conflicts, not just for the restriction to isolated ones. The reassignment-restriction for conflict vertices modelled with the capacities in the network however requires a situation where, in case of conflict, at most one vertex can be moved into a cluster. Capacities on arcs from some additional network-vertices which handle conflicts can not model a scenario where out of three vertices u, v, w, a cluster is restricted to either only contain u or any subset of $\{v, w\}$; this situation occurs when u is in conflict with v and w but $\{u, w\}$ is not a conflict. This structure means that the vertices u, v, w induce a $\mathcal{P}_3$ in the conflict graph. If the conflict graph, or any graph for that matter, does not contain an induced $\mathcal{P}_3$, its connected components are cliques. For this structure, the network can be adjusted to correctly model conflict-free vertex-reassignments. The problem to find, for a given graph, a smallest vertex set whose removal yields a $\mathcal{P}_3$-free graph is sometimes called Induced $\mathcal{P}_3$-Cover, but we will refer to it by Cluster Vertex Deletion, a name associated to the structure of the resulting graph. Formally, we consider the following parameterised definition of this problem:

Cluster Vertex Deletion

Input: Graph $G = (V, E)$, $\ell \in \mathbb{N}$.

Parameter: ℓ

Question: Does there exists a subset $\mathcal{F} \subseteq V$ such that $G[V \setminus \mathcal{F}]$ does not contain $\mathcal{P}_3$ as an induced subgraph?

Currently, there does not seem to be a non-trivial exact algorithm for Cluster Vertex Deletion in the literature so we use the exact algorithm from [63] for 3-Hitting Set which runs in $\mathcal{O}^*(1.6538^n)$, where n is the number of vertices in the hypergraph. By 3-Hitting Set, we denote the following problem:

3-Hitting Set

Input: Hypergraph $H = (V, F)$ such that $|f| = 3$ for all hyperedges $f \in F$, $\ell \in \mathbb{N}$.

Question: Does there exists a subset $\mathcal{C} \subseteq V$ such that $f \cap \mathcal{C} \neq \emptyset$ for all $f \in F$?

Cluster Vertex Deletion can be modelled as an instance of 3-Hitting Set by keeping the same bound ℓ and vertex set and introducing a hyperedge for each induced $\mathcal{P}_3$.

We will now, in a sense, generalise the algorithm for Theorem 46 to consider a cluster vertex deletion set instead of a $\mathcal{P}_3$-cover to reduce the cost for guessing the partition. We denote the corresponding parameter, the size of a minimum cluster vertex deletion set for the conflict graph, by p_{3d}. While the relation $p_{3d} \leq p_{3c}$ obviously makes this generalisation an improvement, we have to pay for this in the branching for the pre-clustering, as the remaining conflicts are no longer bounded by $\frac{1}{2}|P|$. The generalisation also requires more changes to the network and yields the following result:

Theorem 48

A 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(2^c \cdot B_{p_{3d}} + 1.6538^p)$.

Proof. We will alter the algorithm presented for Theorem 46, so let d, $G = (V, E)$, D and G_c^D be defined as in the proof of Theorem 46. Let now $\mathcal{F}_D$ be a minimum *induced* $\mathcal{P}_3$-cover for G_c^D and we consider all partitions $F_1, \dots, F_r$ of this set $\mathcal{F}_D$ with $F_i \neq \emptyset$ and $\mathrm{diam}(F_i) \leq D$ for all $i \in \{1, \dots, r\}$. Let further $\{e_1, \dots, e_b\}$ be the set of edges of the graph $G_c^D[P_D \setminus \mathcal{F}_D]$. We do exactly the same pre-clustering procedure as for Theorem 46, observe that we did not require the edges $\{e_1, \dots, e_b\}$ to be disjoint for this part of the algorithm.

Now that a vertex in $P_D \setminus \mathcal{F}_D$ can be involved in more than one remaining conflict $\{e_1, \dots, e_b\}$, we have to alter the network used for balancing cardinalities to make sure that no conflicts occur in the same set of a resulting k-cluster. Let R be the set of vertices in $P_D \setminus (\mathcal{F}_D \cup \{z_1, \dots, z_q\})$ which are involved in at least two remaining conflicts $e_1, \dots, e_b$, not including conflicts with a vertex in $\{z_1, \dots, z_q\}$. The vertices in R will be modelled differently in the new network to deal with the conflicts they are involved in. For this purpose, first remove from $P(f_i)$ and $P(z_j)$ all vertices from R for all $i \in \{1, \dots, r\}$ and $j \in \{1, \dots, q\}$. Build for the resulting sets $P(f_1), \dots, P(f_r), P(z_1), \dots, P(z_q)$ the network as for Theorem 46. Add to this network the following vertices and arcs to model reassignments for the vertices in R:

- For each vertex $v \in R$, create a network-vertex v'.
- Add an arc of capacity 1 from s to v' for each $v \in R$.
- For each $i \in \{1, \dots, r\}$ with $\{v \in R \colon d(v, w) \leq D \text{ for all } w \in F_i\} =: F_i^R \neq \emptyset$, let $F_i^1, \dots, F_i^{n_i}$ be the vertex sets corresponding to the connected components of the graph $G_c^D[F_i^r]$. Introduce new network-vertices ${f_i}^1, \dots, {f_i}^{n_i}$ representing these sets.
- Similarly for each $i \in \{1, \dots, q\}$ with $Z_i^R := \{v \in R \colon d(v, z_i) \leq D\} \neq \emptyset$, let $Z_i^1, \dots, Z_i^{m_i}$ be the vertex sets corresponding to the connected components of the graph $G_c^D[Z_i^R]$. Introduce new network-vertices $z_i^1, \dots, z_i^{m_i}$ representing these sets.
- Add the arc (v', f_i^j) of capacity 1 for each $v \in F_i^j$, $i \in \{1, \dots, r\}$ and $j \in \{1, \dots, n_i\}$.
- Add the arc $(v', {z_i}^j)$ of capacity 1 for each $v \in X_i^j$, $i \in \{1, \dots, q\}$ and $j \in \{1, \dots, m_i\}$.
- Add the arc (f_i^j, f_i') of capacity 1 for each $i \in \{1, \dots, r\}$, $j \in \{1, \dots, n_i\}$.
- Add the arc $({z_i}^j, z_i')$ of capacity 1 for each $i \in \{1, \dots, q\}$ and $j \in \{1, \dots, m_i\}$.

We claim that these newly defined network-vertices and arcs exactly model the possibilities to assign the vertices in R to the clusters $P(f_1), \dots, P(f_r)$, $P(z_1), \dots, P(z_q)$ such that the maximum diameter does not exceed $2D$, and hence allows to build a k-cluster in case the partition of $\mathcal{F}_D$ is chosen correctly and $z_1, \dots, z_q$ are chosen such that in an optimal solution they are not in a cluster with vertices from $\mathcal{F}_D$ and also pairwise lie in different clusters. First observe that the graph $G_c^D[F_i^j]$ is a clique for each $i \in \{1, \dots, r\}$ and $j \in \{1, \dots, n_i\}$, as the vertices in F_i^j are connected and $G_c^D[F_i^j]$ is an induced subgraph of the cluster graph $G_c^D[P_D \setminus \mathcal{F}_D]$. This means that in a partition of maximum diameter $2D$, no two vertices from F_i^j are included in the same set. In the network, at most one network-vertex v' with $v \in F_i^j$ can send a flow through the corresponding network-vertex f_i^j to f_i' (and then to t), hence at most one vertex $v \in F_i^j$ is moved into the cluster $P(f_i)$. The same holds for the vertices in Z_i^j for each $i \in \{1, \dots, q\}$, $j \in \{1 \dots, m_i\}$. All vertices outside R which are reassigned by this procedure are not in conflict with vertices in R and are correctly modelled like for Theorem 46, as they have exactly the same properties and exactly like in the proof of Theorem 46 it follows that for D chosen as the optimum diameter, there exists a vector g for which this approximation procedure is successful in creating a k-cluster.

In case of a successful vertex-reassignment according to a max-flow for the above network, the maximum diameter of the resulting k-cluster is at most $2D$. This follows exactly like for Theorem 46 with the only additional observation, that vertices from R can never end up in the same cluster, as they are removed and re-assigned separately by the adjusted max-flow procedure.

Overall, this shows that the described adjustment of the algorithm from Theorem 46 still produces a 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER while now only guessing the partition of an induced $\mathcal{P}_3$-cover for G_c^D. By definition, G_c^D is a subgraph of G_c which means that the cardinality of this cover is bounded by our parameter p_{3d} and guessing of the partition can hence be done in $\mathcal{O}^*(B_{p_{3d}})$, while the computation of this set can be done in $\mathcal{O}^*(1.6538^n)$ by [63]. Since now the remaining conflicts do not have to be isolated, we can bound z only by c which, with the still at most $\log(|V|^2)$ many values to check for D, gives an overall running time in $\mathcal{O}^*(2^c \cdot B_{p_{3d}} + 1.6538^n)$. □

Instead of the exact algorithm for 3-HITTING SET, we can also use the currently best known parameterised algorithm for CLUSTER VERTEX DELETION from [15] which runs in $\mathcal{O}^*(1.9102^\ell)$. This adjustment yields:

Corollary 49

A 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(2^c \cdot B_{p_{3d}})$.

All results for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER with respect to the structural parameters of the conflict graph translate to a 4-approximation for the problem

p_c	p_{3c}	p_{3d}
$\mathcal{O}^*(B_{p_c} + 1.1996^p)$ (Theorem 42)	$\mathcal{O}^*(\sqrt{2}^p B_{p_{3c}} + 1.4656^p)$ (Theorem 46)	$\mathcal{O}^*(2^c B_{p_{3d}} + 1.6538^p)$ (Theorem 48)
$\mathcal{O}^*(B_{p_c})$ (Corollary 43)	$\mathcal{O}^*(\sqrt{2}^p B_{p_{3c}})$ (Corollary 47)	$\mathcal{O}^*(2^c B_{p_{3d}})$ (Corollary 49)

Table 5: Summary of the running time of the parameterised 2-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER (4-approximation for $(\|\cdot\|_\infty^w, \mathrm{diam})$-$k$-CLUSTER), where p_c, p_{3c} and p_{3d} denote the size of a minimum vertex, $\mathcal{P}_3$ and induced $\mathcal{P}_3$-cover for the conflict graph, respectively.

variant $(\|\cdot\|_\infty^w, \mathrm{diam})$-$k$-CLUSTER with the same arguments used in Proposition 37 to show how Theorem 35 also applies for the weighted infinity norm. This immediately yields the following result:

Corollary 50

A 4-approximation for $(\|\cdot\|_\infty^w, \mathrm{diam})$-$k$-CLUSTER *can be computed in*

- $\mathcal{O}^*(B_{p_c} + 1.1996^p)$ *or* $\mathcal{O}^*(B_{p_c})$*, where* p_c *is the size of a minimum vertex cover for the conflict graph.*
- $\mathcal{O}^*(\sqrt{2}^p B_{p_{3c}} + 1.4656^p)$ *or* $\mathcal{O}^*(\sqrt{2}^p B_{p_{3c}})$*, where* p_{3c} *is the size of a minimum* $\mathcal{P}_3$*-cover for the conflict graph.*
- $\mathcal{O}^*(2^c B_{p_{3d}} + 1.6538^p)$ *or* $\mathcal{O}^*(2^c B_{p_{3d}})$ *where* p_{3d} *is the size of a minimum induced* $\mathcal{P}_3$*-cover for the conflict graph.*

Remark 8*:* The results to improve the parameterised approximation from Theorem 42 are here presented in a way which suggest a stepwise improvement of the running time. In principle, reducing the number of vertices which require partitioning appears to be the best option. But the reductions of this set used for Theorems 46 and 48 require additional branching costs on conflict vertices and conflicts, respectively. Depending on the structure of the conflict graph, any one of the three algorithms can have the best worst-case running time. An overview of the parameterised approximations for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER with respect to the structural parameters of the conflict graph discussed in this section is given in Table 5.

5.3.3 Lower Bounds

In this section, we investigate the limitations of parameterised approximation for $(\|\cdot\|, f)$-k-CLUSTER with structural parameters of the conflict graph. Especially the increase from ratio 2 for metric instances to ratio 3 for non-metric instances in fpt-time for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER appears strange. We will however see that the approach we use to design parameterised approximations is limited to the performance ratio of 3. Further, we will discuss the limits of choosing structural parameters for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER; more precisely we will see that while the previous section gave approaches to move from p to p_c, p_{3c} and p_{3d}, a next step towards a parameterisation by a split vertex deletion set (definition explained in detail later) does not give a constant factor approximation in fpt-time.

To show the negative results of this section, we use a kind of reduction which links the existence of a parameterised approximation with certain ratio to an fpt-algorithm for a parameterised problem which is believed not to be in FPT. For the first reduction of this type, we consider the following parameterised problem:

MULTICOLOURED DOMINATING SET

Input: Graph $G = (V, E)$, with vertex partition $V = V_1 \cup \cdots \cup V_\ell$.

Parameter: ℓ

Question: Does there exists a subset $\mathcal{D} \subseteq V$ such that $N[\mathcal{D}] = V$ ($\mathcal{D}$ is a dominating set for G) and $|\mathcal{D} \cap V_i| = 1$ for all $i \in \{1, \ldots, \ell\}$?

The colour-coding technique from [6] shows that the $\mathsf{W}[2]$-hardness of the classical MINIMUM DOMINATING SET problem, which is shown in [29], transfers to this restricted version we called *multicoloured* in reference to MULTICOLOURED CLIQUE and the corresponding reduction technique introduced in [36], which denotes the clique problem with the same kind of colour-coding technique.

We will in the following give a reduction from MULTICOLOURED DOMINATING SET to $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER which will show that an approximation in fpt-time for the clustering problem could be used to solve the $\mathsf{W}[2]$-hard domination problem in fpt-time. This kind of reduction shows a lower bound for parameterised approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER under the assumption $\mathsf{FPT} \neq \mathsf{W}[2]$. In particular, we arrive at the following result.

Theorem 51
There exists no $(3 - \varepsilon)$-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER with a running time in $\mathcal{O}^*(f(p_c))$ for any $\varepsilon > 0$ and computable function f, unless $\mathsf{FPT} = \mathsf{W}[2]$.

Proof. Assume there was an $(3-\varepsilon)$-approximation algorithm $\mathcal{A}$ for $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER with a worst-case running time in $\mathcal{O}^*(f(p_c))$ for some computable function f. We will show that the existence of this algorithm implies $\mathsf{FPT} = \mathsf{W}[2]$ by building from $\mathcal{A}$ an algorithm to solve MULTICOLOURED DOMINATING SET with a running time in $\mathcal{O}^*(f(\ell))$, contradicting the W[2]-hardness of the problem.

Let $G = (V, E)$ with $V = V_1, \dots, V_\ell$ and $|V| = n$ be an instance of MULTICOLOURED DOMINATING SET. We describe a polynomial algorithm $\mathcal{R}$ which computes from this given instance a graph $G' = (V', E')$ and edge-weight function $w_{E'}$ as input for $\mathcal{A}$ in the following way (for an illustration of the constructed graph, see Figure 10):

- Build the vertex set V' from the vertices V of G, a copy of V, denoted $\bar{V} = \{\bar{v} \colon v \in V\}$, $\ell + 2$ vertices (which will become the vertex cover of the conflict graph) denoted $u_1, \dots, u_\ell$ and u_r, u_R and an additional set A of $(\ell - 1)n + \ell$ vertices.
- For each $i \in \{1, \dots, \ell\}$ add to E' the edge $\{u_i, v\}$ with $w_{E'}(\{u_i, v\}) = 1$ for all $v \in V_i$.
- For each $v \in V$ add to E' the edge $\{v, u_r\}$ with weight 1
- For each $\{v, w\} \in E$ add to E' the edges $\{v, \bar{w}\}$ and $\{\bar{w}, v\}$ both with weight 1.
- For each $w \in A$ and $v \in V \cup \{u_r\}$ add to E' the edge $\{w, v\}$ with weight 1.
- Add to E' the edge $\{u_r, u_R\}$ with weight 1.
- For each $i \in \{1, \dots, \ell\}$ add to E' the edge $\{v, u_i\}$ with weight 3 for all $v \in V' \setminus V_i$.
- For each $v \in \bar{V}$ add to E' the edge $\{u_r, v\}$ with weight 3.
- For each $v \in V' \setminus \{v_r, v_R\}$ add to E' the edge $\{u_R, v\}$ with weight 3.

Consider G' with $w_{E'}$ as instance of $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER with $k = n + 2$ and let d be the distance induced by $w_{E'}$. If G is a "yes"-instance for MULTICOLOURED DOMINATING SET, let $\{v_1, \dots, v_\ell\}$ be a colourful dominating set for G. Let further $W_1, \dots, W_\ell$ be a partition of V such that $W_i \subseteq N_G[v_i]$ for all $i \in \{1, \dots, \ell\}$; such a partition exists, since $\{v_1, \dots, v_\ell\}$ dominates V. Build from this partition for V a partition for V' in the following way:

- For each $i \in \{1, \dots, \ell\}$ build a cluster $P_i = \{\bar{v} \colon v \in W_i\} \cup \{u_i, v_i\}$ and add to this set $n - |W_i|$ vertices from A. Observe that there are enough vertices in A to distribute them among all sets P_i in such a way as $\sum_{i=1}^{\ell} n - |W_i| = (\ell - 1)n < |A|$.

- Build a cluster P_r of the remaining vertices from V'.

The sets $P_1, \dots, P_\ell, P_r$ are obviously a partition of V'. Each set P_i, $i \in \{1, \dots, \ell\}$ contains exactly $n - |W_i| + |W_i| + 2 = n + 2$ vertices. The set P_r contains the vertices u_r and u_R, exactly ℓ remaining vertices from A and the $n - \ell$ vertices in $V \setminus \{v_1, \dots, v_\ell\}$, so $|P_r| = n + 2$. The partition $P_1, \dots, P_\ell, P_r$ hence is a k-cluster for G'. By definition of the edges in E', the maximum radius of $P_1, \dots, P_\ell, P_r$ is 1, as:

- $\{v_i, w\} \in E$ for all $w \in W_i$ by the definition of W_i for each $i \in \{1, \dots, \ell\}$, which means that $\{v_i, \bar{w}\} \in E'$ with weight 1. Also, $d(v_i, z) = 1$ for all $z \in A$ and $i \in \{1, \dots, \ell\}$ and $d(v_i, u_i) = 1$, which together yields $\text{rad}(P_i) \leq \max\{d(v_i, v) \colon v \in P_i\} = 1$.
- For P_r, we know that $P_r \subset A \cup V \cup \{u_r, u_R\}$ with $u_r \in P_r$. Since each edge $\{u_r, v\} \in E'$ has a weight of 1 for all $v \in A \cup V \cup \{u_R\}$, it follows that $\text{rad}(P_r) \leq \max\{d(u_r, v) \colon v \in P_r\} = 1$.

This means that $P_1, \dots, P_\ell, P_r$ is a solution for $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER with $k = n + 2$ for G' of optimal value 1.

Conversely, assume there exists a k-cluster $\mathfrak{P}$ with $k = n + 2$ for G' such that $\text{rad}(P) \leq 3 - \varepsilon$ for all $P \in \mathfrak{P}$ for some $\varepsilon > 0$. First of all, observe that, since $|V'| = 2n + \ell + 2 + (\ell - 1)n + \ell = (\ell + 1)(n + 2)$ and $k = n + 2$, $\mathfrak{P}$ contains at most $\ell + 1$ clusters. Denote by P_i the cluster in $\mathfrak{P}$ which contains the vertex u_i for each $i \in \{1, \dots, \ell\}$ and by P_r the cluster in $\mathfrak{P}$ which contains u_R. Because of the edges of weight 3 introduced from u_i to all vertices in $V' \setminus V_i$, it follows that, since $\text{rad}(P_i) \leq 3 - \varepsilon$, there exists a vertex $v_i \in P_i \cap V_i \cup \{u_i\}$ which is central in P_i for each $i \in \{1, \dots, \ell\}$. Also $d(u_R, v) < 3$ if and only if $v \in \{u_r, u_R\}$, which means that u_r is central for P_r (if u_R itself was central, P_r could only contain two vertices at radius smaller than 3). As $d(v_i, u_j) = 3$ for all $i \neq j$ $i, j \in \{1, \dots, \ell\}$, it follows that $P_i \neq P_j$ for all $i \neq j$ and, since $d(u_j, u_r) = 3$ it follows that $P_i \neq P_r$ for all $i \in \{1, \dots, \ell\}$. With the above mentioned constraint of at most $\ell + 1$ sets, $\mathfrak{P}$ only contains exactly the $\ell + 1$ sets $P_1, \dots, P_\ell$ and P_r. We claim that $\mathcal{D} := \{v_1, \dots, v_\ell\} \cap V$ is a dominating set for V. Assume on the contrary that there is some $w \in V$ such that $w \notin N_G[v]$ for all $v \in \mathcal{D}$. The vertex $\bar{w} \in V'$ has distance 3 from u_r, so $\bar{w} \in P_i$ for some $i \in \{1, \dots, \ell\}$. As $d(u_i, \bar{w}) = 3$, it follows that $v_i \in V_i$, hence $v_i \in \mathcal{D}$. So, if $\{w, v_i\} \notin E$, there is no edge (of weight 1) between $\bar{w}$ and v_i in G'. A shortest path between $\bar{w}$ and v_i in G', which defines the distance $d(\bar{w}, v_i)$, can not have length 2, as vertices in $\bar{V}$ only have vertices in V at distance 1 while vertices in V are not adjacent in G' and have distance 2 from each other. Since all distances have integer values (in fact they are in $\{0, 1, 2, 3\}$), this means that if $d(\bar{w}, v_i) > 1$, then $d(\bar{w}, v_i) = 3$, which is a contradiction to $\text{rad}(P_i) \leq 3 - \varepsilon$. The set $\mathcal{D}$ hence is a dominating set for G which also satisfies $|V_i \cap \mathcal{D}| = |\{v_i\} \cap V| \leq 1$ and if

$|\{v_1, \ldots, v_\ell\} \cap V| < \ell$, adding an arbitrary vertex $v \in V_i$ for each $i \in \{1, \ldots, \ell\}$ with $V_i \cap \mathcal{D} = \emptyset$ yields a colourful dominating set for G.

Given any multicoloured graph G, we can now decide whether G is a "yes"-instance for MULTICOLOURED DOMINATING SET or not by first running algorithm $\mathcal{R}$ to compute the corresponding graph G' and then running $\mathcal{A}$ for G'. If $\mathcal{A}$ returns a solution of maximum radius less than 3, G is a "yes"-instance for MULTICOLOURED DOMINATING SET, otherwise G is a "no"-instance. This decision is correct, as we have shown that a "yes"-instance G yields a graph G' for which there exists a k-cluster of maximum radius 1 while a "no"-instance G yields a graph G' for which a k-cluster has a maximum radius of at least 3. The $(3 - \varepsilon)$-approximation algorithm $\mathcal{A}$ hence yields a solution of maximum radius less than 3 for G' if and only if G is a "yes"-instance for MULTICOLOURED DOMINATING SET.

The reduction algorithm $\mathcal{R}$ obviously runs in polynomial time and especially produces a graph G' with a size polynomial in the size of G. By definition, the approximation algorithm $\mathcal{A}$ runs in time $\mathcal{O}^*(f(p_c))$, where p_c is the size of a minimum vertex cover for the conflict graph of the input. For G', the conflict graph is a subgraph of the graph induced by all edges of weight 3; observe that the set of conflicts is always a subset of the edges and that all other edges have weight 1 which is the minimum distance in the given graph and hence can not produce a conflict. All edges of weight 3 involve at least one of the vertices in $U := \{u_1, \ldots, u_\ell\} \cup \{u_r, u_R\}$, which means that U is (or rather, more precisely, contains) a vertex cover for the conflict graph of G'. This means that $p_c \leq \ell + 2$ which means that $\mathcal{A}$, and overall also the decision-routine to solve MULTICOLOURED DOMINATING SET, only requires a running time in $\mathcal{O}^*(f(\ell))$. With the W[2]-hardness of MULTICOLOURED DOMINATING SET, the existence of the assumed $(3-\varepsilon)$-approximation algorithm $\mathcal{A}$ for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER with running time in $\mathcal{O}^*(f(p_c))$ hence implies $\mathsf{FPT} = \mathsf{W}[2]$. □

Remark 9*:* The algorithm $\mathcal{R}$ described in the proof of Theorem 51 can be seen as a so-called *fpt gap-reduction* introduced in [10] from MULTICOLOURED DOMINATING SET to $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER with g and ρ in this definition set to the constant functions $g \equiv 3 - \varepsilon$ and $\rho \equiv 3 - \varepsilon$.

Remark 10*:* The reduction used to prove Theorem 51 also illustrates why our parameterised approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER with parameter p given in Theorem 36 really does not have a performance ratio better than 3. The situation illustrated in Figure 10, which shows the gap of 3, is also a case where our algorithmic strategy of picking a central vertex and greedily building clusters fails; if the algorithm chooses v_2' instead of v_2 as central vertex (observe that these two are both not in the set P), then q is one of the vertices in $P \setminus P'$ which have to be assigned by step (c) at the worst possible distance (3 times the optimum) from the central vertex.

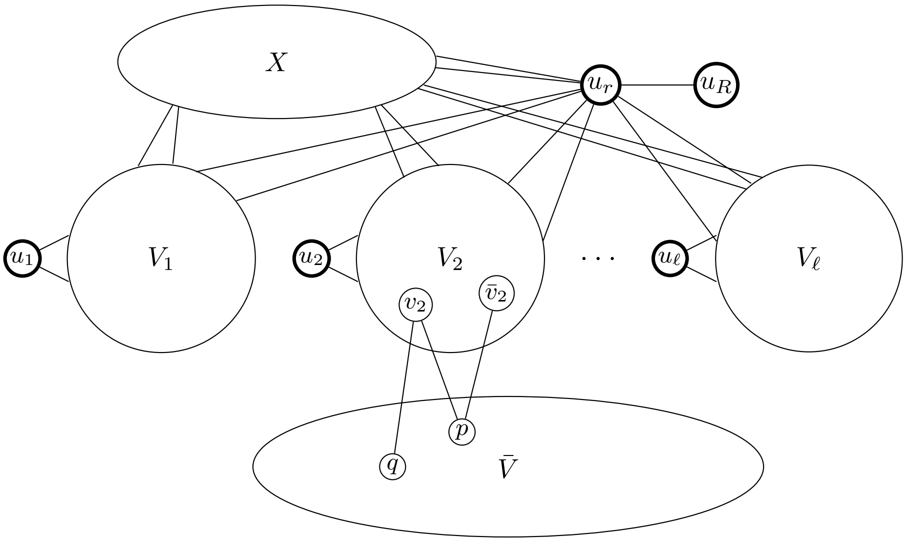


Figure 10: Illustration of the reduction used for Theorem 51.

A graph $G = (V, E)$ is called a *split graph*, if its vertex set can be partitioned into two disjoint sets A and B such that A is an independent set in G and $G[B]$ is the complete graph on vertex set B. Especially for the application to ratings to build recommendation systems, it appears that the conflict graph almost has the structure of a split graph; with a small set of users which give unusual ratings and are hence in conflict among each other (set B) and with a larger set of more average users (set A). This observation raises the question whether it is helpful to turn the conflict graph into a split graph, as this transformation appears to require very little change.

Formally, a *split vertex deletion set* of a graph $G = (V, E)$ is a subset V' of V such that $G[V \setminus V']$ is a split graph. Looking at the previous strategies to lower the parameter from vertex cover to $\mathcal{P}_3$-cover to cluster vertex deletion, the size of a minimum split vertex deletion set appears to be a promising next smaller parameter-choice. Unfortunately, it seems that this parameterisation can not be used for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER as the following result will show. We will use a similar kind of fpt gap-reduction between $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER parameterised by split deletion set and the following problem:

LIST COLOURING(τ)

Input: Graph $G = (V, E)$, colours $\{1, \ldots, \ell\}$ and a list of possible colours for each vertex $v \in V$, given by a list $L(v) \subseteq \{1, \ldots, \ell\}$ for each $v \in V$.

Parameter: $\tau(G)$ (cardinality of a minimum vertex cover for G).

Question: Does there exists a colouring $f \colon V \to \{1, \ldots, r\}$ such that $f(v) \in L(v)$ for all $v \in V$ and $f(v) \neq f(w)$ for all $\{v, w\} \in E$?

With the above chosen parameterisation by the vertex cover number τ, List Colouring is W[1]-hard [34, 37].

Theorem 52

There exists no constant factor approximation for $(\|\cdot\|_\infty, \mathrm{diam})$*-$k$-cluster with a running time in* $\mathcal{O}^*(f(p_s))$*, for any computable function* f*, where* p_s *is the size of a minimum split vertex deletion set, unless* $\mathsf{FPT} = \mathsf{W}[1]$.

Proof. Assume there exists an r-approximation algorithm $\mathcal{A}$ for the problem $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-cluster with worst-case running time in $\mathcal{O}^*(f(p_s))$ for some constant $r > 1$. Similar to the proof of Theorem 51, we will use $\mathcal{A}$ to solve List Colouring(τ), so let $G = (V, E)$, $\{1, \dots, \ell\}$ and $\{L(v)\colon v \in V\}$ be an instance of List Colouring(τ). We describe a reduction algorithm $\mathcal{R}$ which creates from this instance, an instance for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-cluster. On input $G = (V, E)$, $\{1, \dots, \ell\}$ and $\{L(v)\colon v \in V\}$ with $|V| = n$, the algorithm $\mathcal{R}$ creates a graph $G' = (V', E')$ with weight function $w_{E'}$ given by:

- $V' = V \cup L_1 \cup \dots \cup L_\ell$ where each L_i, $i \in \{1, \dots, \ell\}$, is a set of $n+2$ new vertices denoted by $l_i^1, \dots, l_i^{n+2}$.
- E' contains edges such that each $v \in V$ is connected to all vertices in V', each set L_i, $i \in \{1, \dots, \ell\}$ is a clique and the set $\{l_1^1, \dots, l_\ell^1\}$ is also a clique.
- Edges in E have weight $r + 1$.
- The edges among the vertices in $\{l_1^1, \dots, l_\ell^1\}$ have weight $r + 1$.
- For each $c \in \{1, \dots, \ell\}$ and $v \in V$ such that $c \notin L(v)$, $w_{E'}(\{l_c^1, v\}) = r+1$.
- All other edges in E' have weight 1.
- As a technicality to simplify the conflict argumentation, we add one further vertex x to V' which has distance 1 from all vertices in V'.

Consider G' with distances d induced by $w_{E'}$ as instance for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-cluster with $k = n + 2$.

If the given instance G with L is a "yes"-instance for List Colouring(τ), let f be a feasible list colouring. The sets $L_i \cup \{v\colon f(v) = i\}$, $i \in \{1, \dots, \ell\}$ (with x added to one of the sets, arbitrarily) are a k-cluster for G' with maximum diameter 1 as the cardinality constraint is obviously satisfied and for each $i \in \{1, \dots, \ell\}$ and all $u, v \in L_i \cup \{v\colon f(v) = i\}$, it follows that $d(u, v) = 1$ because one of the following cases holds:

1. $u, v \in L_i$, so $d(u, v) = w_{E'}(\{u, v\}) = 1$.

2. $u, v \in V$ which means that $\{u, v\} \notin E$ as $f(u) = f(v) = i$ and f is a feasible colouring, so $d(u, v) = w_{E'}(\{u, v\}) = 1$.

3. $u \in V$ and $v \in L_i \setminus \{l_i^1\}$, which means $d(u, v) = w_{E'}(\{u, v\}) = 1$ by definition.

4. $u \in V$ and $v = l_i^1$. As f is a feasible list colouring, it follows that $i \in L(u)$, so $w_{E'}(\{u, l_i^1\}) = 1$.

5. The additional vertex x has distance 1 from every vertex, and can hence not increase the diameter.

On the other hand, if $\mathfrak{P}$ is a k-cluster for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER with $k = n + 2$ of maximum radius at most r, denote for each $i \in \{1, \ldots, \ell\}$ by P_i the cluster in $\mathfrak{P}$ which contains l_i^1. As $d(l_i^1, l_j^1) = w_{E'}(\{l_i^1, l_j^1\}) = r + 1$ for all $i \neq j$, it follows that $P_i \neq P_j$ for all $i \neq j$, $i, j \in \{1, \ldots, \ell\}$. Since $|V'| = (n+2)\ell + n + 1$, the lower bound on the cluster cardinality of $k = n + 2$ implies that there are at most ℓ sets in the partition $\mathfrak{P}$, so $\mathfrak{P}$ contains exactly the sets $P_1, \ldots, P_\ell$. We claim that the colouring $f(v) = i$ for all $v \in V \cap P_i$, $i \in \{1, \ldots, \ell\}$ is a feasible list colouring for G; it is defined for all $v \in V$ as each vertex in V has to be in one of the sets $P_1, \ldots, P_\ell$. Assume on the contrary that f is not a feasible list colouring, which means one of the following to situations occur:

- $f(v) \notin L(v)$ for some $v \in V$,
- $f(v) = f(w)$ for some $\{v, w\} \in E$.

The first case $f(v) \notin L(v)$ with $f(v) = i$ means that $d(v, l_i^1) = w_{E'}(\{v, l_i^1\}) = r + 1$, which contradicts $\text{diam}(P_i) \leq r$ as, by definition $v, l_i^1 \in P_i$. The second case $f(v) = f(w)$ for some $\{v, w\} \in E$ implies $d(v, w) = w_{E'}(\{v, w\}) = r + 1$. As the colourings $f(v)$ and $f(w)$ were assigned according to the clustering, the set P_i with $i = f(v) = f(w)$ contains both v and w which also is a contradiction to the maximum diameter of r for P_i.

Given graph $G = (V, E)$ with colour-lists $\{L(v) \colon v \in V\}$, we can decide if there exists a feasible list colouring by first using $\mathcal{R}$ to create the corresponding instance $G', w_{E'}$ for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER with $k = n + 1$ and then running algorithm $\mathcal{A}$ on it. The above properties of the reduction show that G with L is a "yes"-instance of LIST COLOURING(τ) if and only if the r-approximation algorithm $\mathcal{A}$ returns a solution of maximum diameter at most r.

The reduction algorithm $\mathcal{R}$ runs in polynomial time and especially produces a graph G' with a size polynomial in the size of G. The algorithm $\mathcal{A}$ runs in time $\mathcal{O}^*(f(p_s))$, where p_s is the size of a minimum split vertex deletion set for the conflict graph of the input. Similar to the argument in Theorem 51, the conflict graph for G', is a subgraph of the graph induced by all edges of weight $r + 1$. Here with the vertex x added to the construction

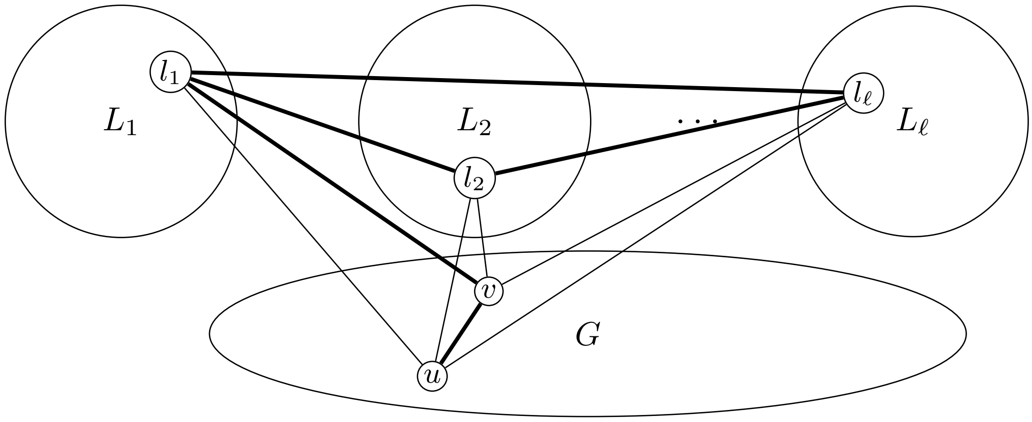


Figure 11: Illustration of the reduction used for Theorem 52, conflict edges (weight $r+1$) drawn bold. Vertices $u, v \in V$ such that $\{u, v\} \notin E$, $1 \in L(u)$, $1 \notin L(v)$, $2 \in L(u) \cap L(v)$ and $\ell \notin L(u) \cup L(v)$.

as shortcut, the conflict graph is exactly the graph induced by the edges of weight $r + 1 > 2$ which includes the vertices $l_1^1, \dots, l_\ell^1$ as clique with connections to the non-isolated vertices in G, connected with the edges in E. Removing any vertex cover for G hence turns the conflict graph into a split graph with clique $l_1^1, \dots, l_\ell^1$ and the remaining independent set in G also as independent set in the conflict graph. This means that $p_s \leq \tau(G)$, so $\mathcal{A}$, and hence the overall described routine to solve List Colouring(τ), runs in $\mathcal{O}^*(f(\tau(G)))$. With the W[1]-hardness of List Colouring(τ) from [34, 37], the existence of the assumed r-approximation algorithm $\mathcal{A}$ for $(\|\cdot\|_\infty, \text{diam})$-$k$-cluster with running time in $\mathcal{O}^*(f(p_s))$ hence implies $\mathsf{FPT} = \mathsf{W}[1]$.

□

Remark 11*:* As the exponential time hypothesis implies $\mathsf{FPT} \neq \mathsf{W}[1]$ by [18], the negative results in this section especially hold assuming ETH.

Both reductions used to show hardness for parameterised approximations here create instances of $(\|\cdot\|, f)$-k-cluster with large values for k. In most applications however, k is a fixed, not too large integer, which raises the question whether an additional parameterisation by k (additional to the number of conflicts, as, like already mentioned in Section 2.3.2, k alone is not helpful) would help overcome the negative results. For the greedy strategies used for the positive results in Sections 5.2 and 5.3, it is not clear how k could be included in a useful way. Better parameterised approximation algorithms with parameterisation by both conflicts and k probably require a different approach and are an interesting open problem.

So far, we did not manage to find these kinds of lower bounds for parameterised approximation for other versions of $(\|\cdot\|, f)$-k-cluster, although also the positive results are weaker. Also, the gap between our positive results and the presented lower bounds leaves room for improvement. Stronger lower bounds seem to require new techniques for reductions which consider both parameterisation and approximation.

One other aspect we did not consider here is the optimality of the asymptotic running times of our positive results. Techniques for such more fine-grained considerations require a more careful analysis. A concrete question in this regard is whether it is possible to improve the parameterised approximations for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER to only require single-exponential time. We did not see a possibility to find such improvements but also did not find lower bounds which suggest that they are unlikely to exist; in this regard it would be very interesting to see if slightly superexponential lower bounds as shown in [47] can be proven for a 2-approximation of $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER with parameter p.

5.4 Parameterisation by Shortcuts

Another option for resolving conflicts with parameterisation is devising special algorithmic strategies for the vertices in X. As already mentioned in Section 5.1, the cardinalities of X and P can differ (almost) arbitrarily which makes the parameterisation by $x = |X|$ a sort of orthogonal approach compared to the parameter p (or c, resp.).

Especially for the radius measure, it seems that shortcuts play a more important role than the vertices involved in a conflict. In particular, these vertices appear to be more suitable to be chosen as central vertices. As the following result shows, it turns out that parameterisation by x indeed yields a better approximation strategy for the radius measure.

Theorem 53

A 2-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(2^x)$.

Proof. We use a simplified version of the algorithm presented for Theorem 36, with shortcut instead of conflict vertices, so let G, d and D be defined as defined in the proof of Theorem 36. We consider, for fixed D, the following subset of shortcut vertices:

$$X_D := \{x \in V \colon \exists\, u, v \in V \colon (d(u,v) > 2D) \wedge (d(u,x), d(v,x) \leq D)\}\,.$$

We guess which of the vertices in X_D are a central vertex in their cluster. For each such $X' \subseteq X_D$ we try to compute a k-cluster for V by successively building clusters until all vertices are partitioned with the following strategy:

(*a*) Pick, while such a vertex exists, a $v \in X'$ that is not assigned to any cluster yet and build a new cluster $P(v)$ with center v by collecting v and all unclustered vertices in $V \setminus X'$ which have distance at most D from v.

(b) If all vertices in X' are clustered, pick any $v \in V$ that is not clustered yet and build a new cluster $P(v)$ with center v by collecting v and all unclustered vertices in V which have distance at most $2D$ from v. [7]

Let $X' = \{x_1, \ldots, x_r\}$ and let $z_1, \ldots, z_q$ be the vertices chosen in step (b) to build clusters $P(z_1), \ldots, P(z_t)$ by the above procedure. Like in Theorem 36, we try to turn this partition into a k-cluster for V by reassigning some vertices, aiming to keep a maximum radius of D for $P(x_1), \ldots, P(x_r)$ and a maximum radius of $2D$ for $P(z_1), \ldots, P(z_q)$. Formally, we allow reassignment of vertices according to the sets:

- $S_x(i,j) := \{v \in P(x_j) \setminus \{x_j\} \colon d(v, x_i) \leq D\}$ for all $1 \leq j < i \leq r$,
- $S(i,j) := \{v \in P(x_j) \setminus \{x_j\} \colon d(v, z_i) \leq 2D\}$ for all $1 \leq j \leq r$ and $1 \leq i \leq q$,
- $S_z(i,j) := \{v \in P(z_j) \setminus \{z_j\} \colon d(v, z_i) \leq 2D\}$ for all $1 \leq j < i \leq q$.

As usual, we try to turn $P(x_1), \ldots, P(x_r), P(z_1), \ldots, P(z_q)$ into a k-cluster by reassigning vertices in the above described sets with the help of a max-flow formulation. Denote in case of a successful reassignment, the resulting k-cluster by $P'(x_1), \ldots, P'(x_r)$, $P'(z_1), \ldots, P'(z_q)$. We claim that the procedure described above behaves very similar to the more complicated algorithm used for the conflict vertices with X taking the place of P, i.e., we claim that for $D = r^*$, there exists a subset $X' \subseteq X_D$ such that the above clustering procedure successfully computes a k-cluster $P'(x_1), \ldots, P'(x_r)$, $P'(z_1), \ldots, P'(z_q)$. Further, a good choice for X' can again be derived from fixed central vertices of a fixed optimal solution, so let $\{S_1, \ldots, S_y\}$ be any optimal solution for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER on input $G = (V, E)$ with distance d. Fix some central vertex s_i for S_i for each $i \in \{1, \ldots, y\}$. Consider running the described greedy procedure for $D = r^*$ and the subset $X' = X_D \cap \{s_1, \ldots, s_y\}$ and let as described in the algorithm $X' = \{x_1, \ldots, x_r\}$.

We first show that any two different vertices in $X' \cup \{z_1, \ldots, z_q\}$ belong to different clusters in the chosen solution $\{S_1, \ldots S_y\}$:

- For two vertices from X' this is true by the choice of X'.
- For a vertex z_i and any $x \in X'$, we know that $d(z_i, x) > D = r^*$, so, since x is central for some cluster S_j which has radius at most r^*, z_i cannot belong to S_j.

[7] Observe that, in contrast to the algorithm for Theorem 36, there is no step (c) for the pre-clustering, as we still allow vertices in $X_D \setminus X'$ to be chosen as central vertices.

- For any two vertices z_i, z_j with $i < j$, we know that both z_i and z_j have distance more than D from all vertices in X' and also that z_j was not clustered in $P(z_i)$ because $d(z_i, z_j) > 2D$. If there was a cluster S_h in the optimal solution such that $\{z_i, z_j\} \subseteq S_h$, then this would imply that $d(s_h, z_i) \leq r^*$ and $d(s_h, z_j) \leq r^*$, while $d(z_i, z_j) > 2D$, which would mean that s_h is a shortcut vertex, so $s_h \in X'$ which contradicts the fact that z_i and z_j were not included in the cluster with center s_h in step (a).

By the choice of X' there exist at least $k - 1$ distinct vertices at distance at most D for each x_i, in particular given by the vertices in the set S_h for which $s_h = x_i$. Since the vertices z_j each belong to a unique cluster S_{i_j} for which the central vertex s_{i_j} is not in X_D (as otherwise X' would have contained s_{i_j} and z_j would have been placed in a cluster with this central vertex), all at least k vertices z from S_{i_j} have distance at most $d(z_j, s_{i_j}) + d(z, s_{i_j}) \leq 2D$ from z_j. Since the sets $S_1, \ldots, S_y$ are pairwise disjoint and of cardinality at least k, the reassignment-procedure described by the max-flow can successfully build a k-cluster by assigning the vertices in S_{i_j} to $P(z_j)$ and the vertices in S_h to $P(x_i)$ for $h \in \{1, \ldots, y\}$ and $i \in \{1, \ldots, r\}$ such that $s_h = x_i$.

If, for some D, the above described procedure successfully builds a partition $P'(x_1), \ldots, P'(x_r), P'(z_1), \ldots, P'(z_q)$, it is again clear that these sets are a k-cluster for V. By the definition of the pre-clustering and the reassignment, it also follows immediately that the sets $P'(x_i)$, $i \in \{1, \ldots, r\}$ have a maximum radius of D and the sets $P'(z_j)$, $j \in \{1, \ldots, t\}$ have a maximum radius of $2D$ which overall makes the described algorithm a parameterised 2-approximation.

At last, the running time of this approximation algorithm is in $\mathcal{O}^*(2^x)$, as it only requires polynomial effort for each set $X' \subseteq X_D$ and $X_D \subseteq X$. □

Although parameterisation by x allows the better approximation factor of 2 for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER, it has the drawback that we do not know if this strategy can be adjusted to also work for a potentially much smaller subset of X; recall that the algorithm for p could with few adjustments be altered to the size of a vertex cover for G_c.

Besides this positive effect for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER, shortcuts do not give an obvious angle to approach other variants of $(\|\cdot\|, f)$-k-CLUSTER. Just like for the above approximation with parameterisation by conflict vertices, i.e., the procedure given for Theorem 36, parameterisation by x as used in Theorem 53 does not translate to a similar result for the weighted infinity norm. For the diameter measure or the approximations derived with tree or path partitioning, it is also not obvious how parameterisation by shortcuts can be efficiently used to resolve conflicts. On the other hand, proving lower bounds for this choice of parameter is also not a simple task, which means that shortcut parameterisation remains wide open for further investigation.

5.5 α-Triangle Inequality

An orthogonal way to approach instances of $(\|\cdot\|, f)$-k-CLUSTER for which the induced distances is not in our sense metric, is considering the severity of the violation of triangle inequality. We will now discuss the restriction to instances for which d satisfies the α-relaxed triangle inequality, as already introduced in Section 5.1; recall that we defined this notion for any $\alpha > 0$ to require that for all $u, v \in V$ the following inequality holds:

$$d(u,v) \leq \alpha \cdot (d(u,w) + d(v,w)) \text{ for all } u,v,w \in V \text{ with } \; w \notin \{u,v\}\,.$$

We will show that the strictly polynomial algorithms introduced for metric instances can in most cases be generalised to α-relaxed triangle inequality with however worsening of the approximation ratio. A parameterisation by α to improve these ratios in the way we parameterised by conflicts, does not seem to be a reasonable option, as we will see in Section 5.5.3 that approximation lower bounds transferred to α-relaxed triangle inequality from the results in Section 2.3.1 make this strategy very unlikely to succeed. We will however briefly discuss the idea to combine conflict parameterisation with the polynomial strategies for α-relaxed triangle inequality in Section 5.5.4.

5.5.1 Application to the Greedy Approximation

For Theorem 20, it is not hard to see that small alterations to the approximation procedure used for this result give a 2α-approximation for instances which satisfy the α-relaxed triangle inequality:

Proposition 54

$(\|\cdot\|_\infty, \mathrm{rad})$- *and* $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER *are 2α-approximable in polynomial time for all $k \geq 2$ and $\alpha \geq 1$, if d satisfies the α-relaxed triangle inequality.*

Proof. Given a graph $G = (V, E)$ with induced distance function d which satisfies the α-relaxed triangle inequality, consider running the procedure described in Theorem 20.

For the diameter-measure, the greedy procedure remains successful for the optimum value $D = \mathrm{opt}(G, d, \mathrm{diam}, \|\cdot\|_\infty, k)$; observe that the crucial property which we used to show this in the proof of Theorem 20 remains true: vertices at distance more than D belong to different clusters in an optimal solution. The computed k-cluster has maximum radius D, which means that by α-relaxed triangle inequality, the pairwise distance of vertices from the same cluster is at most $2\alpha \cdot D$, which gives a maximum diameter of at most 2α times the optimum value.

For the radius, on the other hand, the solution computed by the greedy-procedure still has a maximum radius of the value for D for which it was successful. However, the additional factor of α comes into play when arguing for which D it is possible to find a k-cluster with the described algorithm. Here, α-relaxed triangle inequality for d only implies that vertices can not belong to the same cluster in an optimal solution if their distance is larger than $2\alpha \cdot \mathrm{opt}(G, d, \mathrm{rad}, \|\cdot\|_\infty, k)$. So, we also end up with a 2α-approximation. □

While the change in approximation ratio from standard to α-relaxed triangle inequality is the same for radius and diameter, the arguments used to prove this result are different. When deriving approximations for the weighted infinity norm for this, it turns out that this difference also affects the approximation factor. For the diameter measure, Proposition 24 translates exactly (only exchanging Theorem 20 by Proposition 54) to:

Corollary 55

$(\|\cdot\|_\infty^w, \mathrm{diam})$-$k$-CLUSTER *is 4α-approximable in polynomial time for all $k \geq 2$ and $\alpha \geq 1$, if d satisfies the α-relaxed triangle inequality.*

For radius however, we encounter the problem that bounding the cardinality of the clusters built by Proposition 54 might cause an increase of the radius; just like in Section 5.2.2. Since now the conflicts are bounded in the sense that α-relaxed triangle inequality limits their severity, we at least arrive at the following result:

Proposition 56

$(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER *is $4\alpha^2$-approximable in polynomial time for all $k \geq 2$ and $\alpha \geq 1$, if d satisfies the α-relaxed triangle inequality.*

Proof. For a given graph $G = (V, E)$ with distance d satisfying the α-relaxed triangle inequality, start with the approximation from Theorem 20, which gives a 2α-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER on G by Proposition 54. Just like in Proposition 24, we cut clusters of cardinality k from large clusters to arrive at a partition which only contains clusters of cardinality at most $2k - 1$. This splitting procedure however might result in an increase of the radius by a factor of 2α for the clusters of cardinality k. Combined with Equation 2, which, as already observed in Section 5.2.2, holds for even non-metric distances, and the factor 2α from Proposition 54 this shows that the resulting partition is a $4\alpha^2$-approximation for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER. □

A generalisation for Proposition 25 for α-relaxed triangle inequality, also brings an increase of approximation ratio by a factor of α^2:

Proposition 57

$(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER *is $(4k-2)\alpha^2$-approximable in polynomial time for all $k \geq 2$ and $\alpha \geq 1$, if d satisfies the α-relaxed triangle inequality.*

Proof. Given a graph $G = (V, E)$ with distance d satisfying the α-relaxed triangle inequality, first observe that the inequality used in the proof of Proposition 25 generalises to $\alpha \cdot \mathrm{opt}(G, d, \mathrm{avg}, \|\cdot\|_\infty^w, k) \geq \mathrm{opt}(G, d, \mathrm{diam}, \|\cdot\|_\infty, k)$, as for any set P in an optimal solution for $(\|\cdot\|_\infty, \mathrm{avg})$-$k$-CLUSTER α-relaxed triangle inequality and $k \geq 2$ yields:

$$\begin{aligned} |P| \cdot \mathrm{avg}(P) = \ & \min\left\{\sum_{p \in P} d(c, p) \colon c \in P\right\} \\ \geq \ & \min\{\max\{d(u, c) + d(v, c) \colon u, v \in P, u \neq v\} \colon c \in P\} \\ \geq \ & \max\{\tfrac{1}{\alpha} \cdot d(u, v) \colon u, v \in P\} = \tfrac{1}{\alpha} \cdot \mathrm{diam}(P)\,. \end{aligned}$$

Running Theorem 54 followed by simply splitting up large clusters of cardinality more than $2k - 1$ (recall that Corollary 1 also holds if d violates the triangle inequality) produces a 2α-approximation for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER on G for which each set contains at most $2k - 1$ vertices. The global cost of the resulting partition with respect to the weighted infinity norm and average distortion is at most $(2k - 1) \cdot 2\alpha \cdot \mathrm{opt}(G, d, \mathrm{diam}, \|\cdot\|_\infty, k)$, and hence yields an approximation of ratio $(4k - 2)\alpha^2$ for $(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER. □

Remark 12*:* If α-relaxed triangle inequality holds for some $\frac{1}{2} \leq \alpha < 1$, d is not just what we referred to as *metric* but satisfies an even stricter requirement than simple triangle inequality; so quite the opposite of *relaxed* (observe that $\alpha < \frac{1}{2}$ is uninteresting as this can only hold if all distances are zero). In this case we can improve on the results given in Section 3. In fact, all arguments used to prove Proposition 54 still hold in this case and yield an approximation factor of $2\alpha < 2$ for $(\|\cdot\|_\infty, \mathrm{rad})$- and $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER. Observe that the lower bound from Corollaries 9 and 12 do not apply to the restriction to α-relaxed triangle inequality with $\alpha < 1$. Similarly, Corollary 55 and Proposition 56 yield a 4α-approximation for $(\|\cdot\|_\infty^w, \mathrm{diam})$- and $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER, respectively. For $k \geq 3$, we can also use the argumentation from Proposition 57, which gives a $(4k - 2)\alpha^2$-approximation for $(\|\cdot\|_\infty^w, \mathrm{avg})$-$k$-CLUSTER even for $\frac{1}{2} \leq \alpha < 1$.

5.5.2 Application to Constraint Forest Approximations

The effect of a worsening in approximation ratio by multiple factors of α as already observed for the greedy approximation, appears to occur even stronger for the approximations derived from tree (or path) partitioning. As illustrated in Figure 12, α-relaxed triangle inequality allows for a situation where the average distortion (or radius) of a set given by a connected component of a tree (or path) partition increases by a factor of $\alpha^{\lceil \log k \rceil}$ with respect to the sum of the cost of the connecting edges. For a generalisation of Proposition 21, this effect is the worst that can happen and we arrive at the following result:

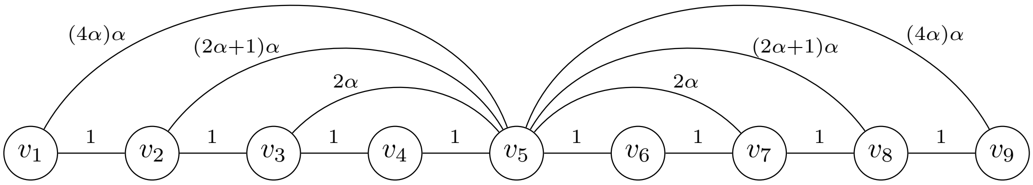


Figure 12: Illustration of the effect of α-relaxed triangle inequality for the approximations based on constraint forest problems.

Proposition 58
*$(\|\cdot\|_1^w, \mathrm{avg})$-$k$-*CLUSTER *is $2k\alpha^{\lceil \log k\rceil}$-approximable in polynomial time for all $k \geq 2$ and $\alpha > 1$, if d satisfies the α-relaxed triangle inequality.*

Proof. Consider the approximation procedure discussed in the proof of Proposition 21 for an instance (G, d) of $(\|\cdot\|_1^w, \mathrm{avg})$-$k$-CLUSTER where d only satisfies the α-relaxed triangle inequality for some $\alpha > 1$. The cost of a minimum tree partitioning of capacity k for G' (the complete graph on the vertices of G with edge-cost given by d) is still at most the global cost of an optimal solution for $(\|\cdot\|_1^w, \mathrm{avg})$-$k$-CLUSTER on G.

Any tree partitioning $\bar{E}$ of capacity k for G' of cost L with connected components $\mathfrak{C}$ can again be interpreted as a k-cluster $\mathfrak{P}$ for G by taking the vertex sets associated to the components in $\mathfrak{C}$. For each $C \in \mathfrak{C}$, we again fix a vertex $c \in C$ for which $C \setminus \{c\}$ is a forest of trees each of maximum cardinality k; denote again by $T_1^c, \ldots, T_{s_c}^c$ the connected components of $C \setminus \{c\}$. For this vertex c, the length of a longest path to any vertex in C along the edges in C is at most k. So, for each vertex $v \in C$ in component T_i^c α-relaxed triangle inequality for d yields $d(v, c) \leq \alpha^{\lceil \log k\rceil} c(E[T_i^c])$; this can easily be seen by induction on k, as an additional factor of α can only be introduced for every time k is doubled. A computation analogous to the one conducted for Proposition 21 with the additional factor of $\alpha^{\lceil \log k\rceil}$ yields $\mathrm{avg}(\mathfrak{P}) \leq \alpha^{\lceil \log k\rceil} k \cdot L$, which makes an application of the polynomial 2-approximation for tree-partitioning interpreted as a partition a $2k\alpha^{\lceil \log k\rceil}$-approximation for $(\|\cdot\|_1^w, \mathrm{avg})$-$k$-CLUSTER. □

For the results based on path partitioning (Proposition 22, Corollary 23), we already have the problem, that the 4-approximation for path partitioning requires a distance which satisfies the triangle inequality. In our case of a complete graph this algorithm simply computes a 2-approximation for tree-partitioning and then flattens the tree with increasing the cost by at most a factor of 2. For the generalisation to α-relaxed triangle inequality, this flattening already yields a much larger increase of cost which is pretty difficult to bound. The strategy sketched in Remark 4 is probably more suitable for designing approximations for $(\|\cdot\|_1^w, \mathrm{diam})$-$k$-CLUSTER restricted to α-relaxed triangle inequality. For a generalisation of Corollary 23, we further run into the problem that Equation 1, which we used for this result to relate radius to diameter, is also affected by the relaxation of the triangle inequality. The problem variant $(\|\cdot\|_1^w, \mathrm{rad})$-$k$-CLUSTER hence also requires more sophisticated strategies to make use of the α-relaxed triangle inequality.

5.5.3 Lower Bounds

The lower bounds for the approximation ratios derived from the reductions in Section 2.3.1 quite easily generalise to α-relaxed triangle inequality. More precisely, if we change the constructions given in Theorems 8, 11 and 13 by adding for all $u, v \in V$ with $\{u, v\} \notin E$ and edge of weight 2α, we can translate pretty much all lower bounds from Section 2.3.1.

In the proof of Theorem 8 there is now a gap of 2α for the maximum radius between "yes"- and "no"-instance for EXACT-k-COVER, as the maximum radius is either equal to 1, meaning that all vertices in a cluster are adjacent to the central vertex in the original construction, or at least the weight of a newly introduced edge, which implies:

Corollary 59
There is no $(2\alpha - \varepsilon)$-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER in polynomial time for any $k \geq 3$, $\alpha \geq 1$ and any $\varepsilon > 0$, unless $\mathsf{P} = \mathsf{NP}$, even if d satisfies α-relaxed triangle inequality.

Altering further the reduction used for Theorem 8 for $k \geq 4$ to reduce to EXACT-$(k-1)$-COVER, like used for Corollary 10, we can conclude that in case of a "yes"-instance for EXACT-$(k-1)$-COVER all clusters in a k-cluster of maximum radius 1 for the corresponding graph G have to contain exactly k vertices. This yields a gap of 2α also for the maximum weighted radius between "yes"- and "no"-instance for EXACT-$(k-1)$-COVER, which implies:

Corollary 60
There is no $(2\alpha - \varepsilon)$-approximation for $(\|\cdot\|_\infty^w, \mathrm{rad})$-$k$-CLUSTER in polynomial time for any $k \geq 4$, $\alpha \geq 1$ and any $\varepsilon > 0$, unless $\mathsf{P} = \mathsf{NP}$, even if d satisfies the α-relaxed triangle inequality.

The reduction given in Theorem 11 also turns into a gap-reduction with a gap of 2α for both weighted and unweighted infinity norm, which gives:

Corollary 61
There is no $(2\alpha - \varepsilon)$-approximation in polynomial time for $(\|\cdot\|_\infty, \mathrm{diam})$- or $(\|\cdot\|_\infty^w, \mathrm{diam})$-$k$-CLUSTER for any $k \geq 3$, $\alpha \geq 1$ and any $\varepsilon > 0$, unless $\mathsf{P} = \mathsf{NP}$, even if d satisfies the α-relaxed triangle inequality.

Only the inapproximability for average distortion can not be translated to a stronger result for α-relaxed triangle inequality, which concludes this section on lower bounds.

5.5.4 Summary and Possible Further Applications

Table 6 summarises the approximations for restriction to α-relaxed triangle inequality. Unfortunately, for many variants the ratios differ more than a single factor of α from the ratios for metric instances while the lower bounds do not seem to support such a behaviour. It is hence quite likely, that a more careful adjustment to α-relaxed triangle inequality and a more thorough analysis can yield better results.

	rad	diam	avg
$\|\cdot\|_\infty$	$\boldsymbol{2\alpha}$ (Proposition 54)	$\boldsymbol{2\alpha}$ (Proposition 54)	?
$\|\cdot\|_\infty^w$	$4\alpha^2$ (Proposition 56)	4α (Corollary 55)	$(4k-2)\alpha^2$ (Proposition 57)
$\|\cdot\|_1^w$	?	?	$2k\alpha^{\lceil \log k \rceil}$ (Proposition 58)

Table 6: Summary of the approximation ratios if d only satisfies α-relaxed triangle inequality for some $\alpha \geq 1$; bold ratios are optimal assuming $\mathsf{P} \neq \mathsf{NP}$.

One possible further degree of freedom for the above discussed approaches to deal with non-metric instances can be gained by combining the results for α-relaxed triangle inequality and the parameterised approximations. It appears to be possible to have a flexible trade between running time and approximation ratio by fixing a desired $\alpha \geq 1$ and only considering the conflict set

$$C_D^\alpha := \{\{u,v\} \colon \exists\, w \in V \colon d(u,v) > \alpha \cdot (d(u,w) + d(v,w))\}\,.$$

Considering the improvement of the approximation ratio for α-relaxed triangle inequality with $\alpha < 1$ as mentioned in Remark 12, it might even be possible to design a parameterised approximation with arbitrarily small approximation ratio. This idea might yield the same kind of flexible family of parameterised approximation algorithms, usually referred to as *efficient polynomial time approximation scheme*, short *EPTAS*, as defined for example in [38].

6 Conclusions

This thesis introduced the general problem $(\|\cdot\|, f)$-k-CLUSTER to model clustering tasks which do not fix the number of clusters but require each cluster to contain at least k objects. The nine specific variants of this problem chosen here generalise many previous models but, of course, do not capture every possible way to measure the quality of such a clustering. We however tried to cover many previous models while maintaining a clear framework. The unified abstract formulation of $(\|\cdot\|, f)$-k-CLUSTER proved to be very useful in exploiting similarities between the specific problem variants, but also in pointing out significant differences caused by the choice for local and global cost function.

In our attempt to find efficient ways to solve variants of $(\|\cdot\|, f)$-k-CLUSTER we considered structural properties of the problem with respect to the role of the lower bound k as well as several restrictions of the distance d. From the methodological side, we discussed polynomial-time solvability, approximability and parameterised complexity with respect to different parameterisations. Generally, most results in this thesis come from a combination of insights from different viewpoints such as approximations which translate from one problem variant to another, algorithmically useful structural observations with respect to k which are only possible with triangle inequality for d, approximation strategies based on a combination of polynomial algorithms for restricted problem cases, and, most prominently, parameterised approximation algorithms.

Considering the complexity of the nine variants of $(\|\cdot\|, f)$-k-CLUSTER discussed here, it turned out that the lower bound k plays a minor role compared to the effect of the properties of the distance d. In particular, we did not find positive results based on a restriction of the lower bound k other than the very specific restriction to $k = 2$. The properties of d however play such an important role that they are reflected in the general organisation of the whole thesis. Restriction to triangle inequality for d enabled polynomial time approximations for eight of the nine variants of $(\|\cdot\|, f)$-k-CLUSTER, while such results are highly unlikely without such a restriction. Our investigation of instances in fixed dimension however showed that the impact of restrictions of d seems to be limited in the sense that it can not break the general NP-hardness; at least not with the most obvious idea to fix d to the Euclidean norm.

The specific algorithmic approaches to solve variants of $(\|\cdot\|, f)$-k-CLUSTER for general k discussed in this thesis all rely on certain properties of d. From restriction to (α-relaxed) triangle inequality to parameterisation by conflicts, shortcuts and related parameters, the results always give a better approximation ratio or are most efficient if d is in some sense close to a metric. Nevertheless, the concept of parameterised approximation yields efficient ways to approach non-metric instances. Our lower bounds, especially for these results, do not match the guarantees of the presented algorithms which suggests that there is still room for improvement.

Aside from $(\|\cdot\|, f)$-k-CLUSTER, there are many other related clustering-type problems which exhibit similar difficulties with violations of the triangle inequality. The parameterisation by conflicts and related parameters discussed in this thesis might provide a useful way to approach these problems as well. Further, we believe that structural parameterisation for approximations in general is an underestimated and hence still very rarely considered research direction from which many other problems could benefit greatly.

At last, we will give a more detailed summary of the results presented in this thesis, list specific open problems related to those which also mention some broader ideas for further research in this direction.

6.1 Summary of Results

As a first step in our investigation of the nine variants of $(\|\cdot\|, f)$-k-CLUSTER, we considered the role of the lower bound k. It turned out that as soon as k is larger than 2, all variants of $(\|\cdot\|, f)$-k-CLUSTER are already NP-hard; shown for the local cost radius in Theorem 8, for diameter in Theorem 11 and for average distortion in Theorem 13. These results harshly show that a parameterisation by the lower bound k can not (at least not alone) help in designing efficient algorithms for $(\|\cdot\|, f)$-k-CLUSTER. For the remaining case, k fixed to 2, the choice of local and global measure yields a quite peculiar diverse behaviour. While some variants can be solved efficiently by a reduction to matching-type problems, others are already intractable; recall here the summary of these results (restatement of Table 1):

$k = 2$	rad	diam	avg
$\|\cdot\|_\infty$	in P *(Edge Cover)* (Proposition 5)	in P *(Simplex Cover)* (Proposition 7)	NP-complete (Theorem 19)
$\|\cdot\|_\infty^w$	NP-complete (Theorem 19)	in P *(Simplex Cover)* (Proposition 7)	NP-complete (Theorem 19)
$\|\cdot\|_1^w$	APX-hard (Theorem 18)	in P *(Simplex Matching)* (Proposition 6)	in P *(Weighted Edge Cover)* (Theorem 4)

An interesting further result from Section 2 is that the restriction to distances d which satisfy the triangle inequality turns the generally NP-hard problem $(\|\cdot\|_\infty^w, \mathrm{avg})$-2-CLUSTER into a problem that can be solved in polynomial time. We further showed that the reductions used to prove the NP-hardness results also prove that this restriction to metric distances is a necessary requirement (assuming P $\neq$ NP) for the existence of polynomial time approximations, see Proposition 17.

In Section 3 we therefore considered exactly this restriction of d to find polynomial time approximations. Especially the relation between the different problem variants proved to be very useful to translate results here. A summary of the approximation ratios for the different problem variants with the restriction to instances for which d satisfies triangle inequality is given in the table below with bold values being optimal assuming $\mathsf{P} \neq \mathsf{NP}$ (restatement of Table 2):

	rad	diam	avg
$\|\cdot\|_\infty$	**2** (Theorem 20)	**2** (Theorem 20)	?
$\|\cdot\|_\infty^w$	4 (Proposition 24)	4 (Proposition 24)	$4k-2$ (Proposition 25)
$\|\cdot\|_1^w$	$16(k-1)$ (Corollary 23)	$8(k-1)$ (Proposition 22)	$2k$ (Proposition 21)

Based on the success of restriction to triangle inequality, we considered in Section 4 the problem family EUCLIDEAN $(\|\cdot\|, f)$-k-CLUSTER, a more specific version of $(\|\cdot\|, f)$-k-CLUSTER, where the objects represented by V are points in $\mathbb{R}^\delta$ and d is the Euclidean norm. Considering the *curse of dimensionality*, we investigated if the dimension w is the source of computational hardness for these problems, but found that NP-hardness remains already for small fixed values for δ (also in combination with a fixed value for k). A summary of the NP-hardness results for EUCLIDEAN $(\|\cdot\|, f)$-CLUSTER with the concrete fixed values for δ and k is given in the table below (restatement of Table 3):

	rad	diam
$\|\cdot\|_\infty$	$k=7$ $\delta=3$ (Theorem 32)	$k=4$ $\delta=3$ (Theorem 29)
$\|\cdot\|_\infty^w$	$k=7$ $\delta=3$ (Proposition 33)	$k=4$ $\delta=3$ (Proposition 30)
$\|\cdot\|_1^w$	$k=12$ $\delta=2$ (Theorem 31)	$k=6$ $\delta=2$ (Theorem 28)

In Section 5, we tried different approaches to generalise the approximations for metric to general instances. We first considered parameterisation by conflicts (edges which violate the triangle inequality) and conflict vertices (vertices included in conflicts) in Section 5.2. A summary of the approximation ratios and asymptotic running times with respect to the number of conflict vertices p of the resulting parameterised approximations is given in the table below (restatement of Table 4):

	rad		diam	avg
$\|\cdot\|_\infty$	2 $\mathcal{O}^*(n^p)$ (Theorem 34)	3 $\mathcal{O}^*(2^p)$ (Theorem 36)	2 $\mathcal{O}^*(B_p)$ (Theorem 35)	?
$\|\cdot\|_\infty^w$	4 $\mathcal{O}^*(n^p)$ (Proposition 38)		4 $\mathcal{O}^*(B_p)$ (Proposition 37)	$4k$ $\mathcal{O}^*(n^p)$ (Proposition 39)
$\|\cdot\|_1^w$	$16(k-1)$ $\mathcal{O}^*(n^p)$ (Corollary 40)		$8(k-1)$ $\mathcal{O}^*(n^p)$ (Corollary 40)	$2k$ $\mathcal{O}^*(n^p)$ (Corollary 40)

In Section 5.3 we found that the parameterised approximations for the problem variants $(\|\cdot\|_\infty, \text{rad})$- and $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER could, with little additional algorithmic effort, be altered to also work for a parameterisation by the vertex cover number of the conflict graph, denoted by p_c. For the diameter measure, it turned out that this parameter could be reduced even further. A Summary of the running time of the parameterised 2-approximation for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER (which is also a 4-approximation for $(\|\cdot\|_\infty^w, \text{diam})$-$k$-CLUSTER) is given in the table below (restatement of Table 5), where p_{3c} and p_{3d} denote the size of a minimum $\mathcal{P}_3$ and induced $\mathcal{P}_3$-cover for the conflict graph, respectively:

p_c	p_{3c}	p_{3d}
$\mathcal{O}^*(B_{p_c} + 1.1996^p)$ (Theorem 42)	$\mathcal{O}^*(\sqrt{2}^p B_{p_{3c}} + 1.4656^p)$ (Theorem 46)	$\mathcal{O}^*(2^c B_{p_{3d}} + 1.6538^p)$ (Theorem 48)
$\mathcal{O}^*(B_{p_c})$ (Corollary 43)	$\mathcal{O}^*(\sqrt{2}^p B_{p_{3c}})$ (Corollary 47)	$\mathcal{O}^*(2^c B_{p_{3d}})$ (Corollary 49)

For $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER, we further found that parameterisation by shortcuts can also be used to efficiently compute approximate solutions. Specifically it turned out that for this parameter it is possible to compute a 2-approximation in fpt-time, while the lower bounds in Section 5.3.3 suggest that the ratio of 3 is optimal for fpt-time parameterised approximation with respect to the parameter p_c.

In Section 5.5 we considered as a different approach to find approximations for non-metric instances of $(\|\cdot\|, f)$-k-CLUSTER the restriction to instances for which d satisfies α-relaxed triangle inequality for some $\alpha \geq 1$. The performance ratios of the approximations discussed there are summarised in the table below (restatement of Table 6), where again bold ratios are optimal assuming $\mathsf{P} \neq \mathsf{NP}$:

	rad	diam	avg
$\|\cdot\|_\infty$	$\boldsymbol{2\alpha}$ (Proposition 54)	$\boldsymbol{2\alpha}$ (Proposition 54)	?
$\|\cdot\|^w_\infty$	$4\alpha^2$ (Proposition 56)	4α (Corollary 55)	$(4k-2)\alpha^2$ (Proposition 57)
$\|\cdot\|^w_1$	?	?	$2k\alpha^{\lceil \log k \rceil}$ (Proposition 58)

Another interesting result from Section 5.5 is the observation stated in Remark 12, which showed that restriction to α-relaxed triangle inequality with $\alpha < 1$ (not a relaxation but rather a further restriction) can even yield an improvement on some of the approximation ratios for general metric instances.

6.2 Summary of Open Problems

Throughout this thesis, we already mentioned questions which remained unanswered and ideas for further interesting research directions. Here, we want to briefly summarise a list of concrete open problems:

- $(\|\cdot\|_\infty, \text{avg})$-$k$-CLUSTER is the only variant of $(\|\cdot\|, f)$-k-CLUSTER for which we did not find any approximation strategy with provable performance guarantee. The lack of monotonicity of average distortion appears to be the main problem in this regard. This behaviour led us to believe that $(\|\cdot\|_\infty, \text{avg})$-$k$-CLUSTER is indeed a problem which is likely to be hard to approximate but we also did not succeed in finding an appropriate reduction to prove this. The approximability of $(\|\cdot\|_\infty, \text{avg})$-$k$-CLUSTER hence remains open.

- Section 3.4 derives an approximation for $(\|\cdot\|_1^w, \text{diam})$-4-CLUSTER by combining the polynomial strategies for the case $k = 2$ for $(\|\cdot\|_1^w, \text{diam})$- and $(\|\cdot\|_1^w, \text{avg})$-2-CLUSTER. We believe that it is possible to expand this result for larger values of k (powers of 2, to be precise) by nesting more than 2 applications of the polynomial procedures for $k = 2$. While very difficult to analyse, the implementation of this generalisation is fairly simple, and our experimental results in which we tried exactly this indicate that this strategy yields reasonable results; it also seemed that the approximation ratio for $k = 4$ is in fact much better than $\frac{35}{6}$. These observations suggest that a clever combination of the polynomial procedures for $k = 2$ might give much better approximation algorithm for $(\|\cdot\|_1^w, \text{diam})$-$k$-CLUSTER (and possibly also $(\|\cdot\|_1^w, \text{rad})$-$k$-CLUSTER) than Proposition 22.

- Section 4 showed that the restriction to constant dimensionality for instances of EUCLIDEAN $(\|\cdot\|, f)$-k-CLUSTER remains NP-complete if the local cost f is radius or diameter. The case $f = \text{avg}$ was left open, with strong indication that the construction used for radius might already provide a reduction. Further, the given results always fix some value for k and some dimensionality; while it is quite obvious that the results also hold for larger k and larger dimensionality, the given lower bounds are not necessarily minimal, which raises the question of finding for each problem variant the minimum k and dimension d for which NP-completeness holds.

- Also concerning the restriction to constant dimensionality, the given NP-completeness results do not give (good) lower bounds with respect to approximation hardness. It is in fact quite likely that specific strategies for geometric instances of $(\|\cdot\|, f)$-k-CLUSTER allow for much better approximations than the ones given in Section 3 for general metric instances.

- The parameterised approximations with parameter p based on constraint forest problems only gave xp-time algorithms. It is not clear if such approximations for the corresponding problem variants can be designed to only require fpt-time. It appears that this requires already different strategies for metric instances.

- For parameterisation by shortcuts, we only developed an approximation algorithm for $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER. For all other variants it is completely open whether this parameter is useful to design efficient algorithms.

- The lower bounds for conflict parameterisation given in Section 5.3.3 only give a vague intuition about the optimality of the parameterised approximations given in this thesis. In particular, it remains open if:

  - a parameterised 2-approximation for $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER which only requires exponential running time in p is possible.
  - the parameterised 2-approximation introduced for $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER can be improved to run in single-exponential time in p (or even for the smaller parameters p_c, p_{3c} or p_{3d}). In this regard, it might be interesting to analyse the algorithms presented in this thesis with a closer look at the enumerations of the partitions of P_D. We always estimated this with the Bell number although we only consider partitions with specific properties which in a sense relate to colourings of the conflict graph. It might be possible to enumerate the relevant partitions of P_D more efficiently with the help of colouring strategies.

- Section 5.5 only discussed very simple generalisations of the algorithms from Section 3 for α-relaxed triangle inequality; only the approximation for $(\|\cdot\|_\infty, \mathrm{rad})$- and $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER derived this way is provably optimal. In particular we do not give any strategy for $(\|\cdot\|_1^w, \mathrm{diam})$- or $(\|\cdot\|_1^w, \mathrm{rad})$-$k$-CLUSTER. Further investigations for α-relaxed triangle inequality are quite likely to yield stronger results.

Acknowledgements

This thesis would not have been possible without funding by the PhD scholarships of the German Academic Exchange Service (DAAD) and the Stipendienstiftung Rheinland-Pfalz, and especially the corresponding grant (number FE 560/6-1) provided by the German Science Foundation (DFG). As for the exceptional people who have supported me not just for this work but in general, I dearly hope that you do not need some written acknowledgement which could never reflect my gratitude for the many ways you have changed my life.

References

[1] F. N. Abu-Khzam, C. Bazgan, K. Casel, and H. Fernau. Building clusters with lower-bounded sizes. In S.-H. Hong, editor, *27th International Symposium on Algorithms and Computation, ISAAC*, volume 64 of *LIPIcs*, pages 4:1–4:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016.

[2] F. N. Abu-Khzam, C. Bazgan, K. Casel, and H. Fernau. Clustering with lower-bounded sizes. *Algorithmica*, Sep 2017.

[3] C. C. Aggarwal. On k-anonymity and the curse of dimensionality. In K. Böhm, C. S. Jensen, L. M. Haas, M. L. Kersten, P.-Å. Larson, and B. C. Ooi, editors, *Proceedings of the 31st International Conference on Very Large Data Bases, VLDB'05*, pages 901–909. ACM, 2005.

[4] G. Aggarwal, R. Panigrahy, T. Feder, D. Thomas, K. Kenthapadi, S. Khuller, and An Zhu. Achieving anonymity via clustering. *ACM Transactions on Algorithms*, 6(3), 2010.

[5] P. Alimonti and V. Kann. Some APX-completeness results for cubic graphs. *Theoretical Computer Science*, 237(1-2):123–134, 2000.

[6] N. Alon, R. Yuster, and U. Zwick. Color coding. In M.-Y. Kao, editor, *Encyclopedia of Algorithms*. Springer, 2008.

[7] E. Anshelevich and A. Karagiozova. Terminal Backup, 3D Matching, and Covering Cubic Graphs. *SIAM Journal on Computing*, 40(3):678–708, 2011.

[8] A. Armon. On min-max r-gatherings. *Theoretical Computer Science*, 412(7):573–582, 2011.

[9] G. Ausiello. *Complexity and approximation: combinatorial optimization problems and their approximability properties.* Springer, 1999.

[10] C. Bazgan, M. Chopin, A. Nichterlein, and F. Sikora. Parameterized inapproximability of target set selection and generalizations. *Computability*, 3(2):135–145, 2014.

[11] R. Bellman. *Dynamic Programming.* Princeton University Press, 1957.

[12] D. Berend and T. Tassa. Improved bounds on bell numbers and on moments of sums of random variables. *Probability and Mathematical Statistics*, 30(2):185–205, 2010.

[13] J. Blocki and R. Williams. Resolving the Complexity of Some Data Privacy Problems. In S. Abramsky, C. Gavoille, C. Kirchner, F. Meyer

auf der Heide, and P. G. Spirakis, editors, *Proceedings of the 37th International Colloquium Conference on Automata, Languages and Programming, ICALP'10: Part II*, volume 6199 of *LNCS*, pages 393–404. Springer, 2010.

[14] B. Bollobás. *Modern Graph Theory*, volume 184 of *Graduate texts in mathematics*. Springer, 1998.

[15] A. Boral, M. Cygan, T. Kociumaka, and M. Pilipczuk. A fast branching algorithm for cluster vertex deletion. *ACM Transactions on Computer Systems*, 58(2):357–376, 2016.

[16] J.-W. Byun, A. Kamra, E. Bertino, and N. Li. Efficient k-anonymization using clustering techniques. In R. Kotagiri, P. R. Krishna, M. Mohania, and E. Nantajeewarawat, editors, *Advances in Databases: Concepts, Systems and Applications*, volume 4443 of *LNCS*, pages 188–200. Springer, 2007.

[17] L. Cai and X. Huang. Fixed-parameter approximation: Conceptual framework and approximability results. *Algorithmica*, 57(2):398–412, 2010.

[18] J. Chen, X. Huang, I. A. Kanj, and G. Xia. Strong computational lower bounds via parameterized complexity. *Journal of Computer and System Sciences*, 72(8):1346–1367, 2006.

[19] J. Chen, I. A. Kanj, and G. Xia. Improved upper bounds for vertex cover. *Theoretical Computer Science*, 411(40–42):3736–3756, 2010.

[20] Y. Chen, M. Grohe, and M. Grüber. On parameterized approximability. In H. L. Bodlaender and M. A. Langston, editors, *Parameterized and Exact Computation, Second International Workshop, IWPEC 2006, Zürich, Switzerland, September 13-15, 2006, Proceedings*, volume 4169 of *LNCS*, pages 109–120. Springer, 2006.

[21] G. Cornuéjols, D. Hartvigsen, and W. Pulleyblank. Packing subgraphs in a graph. *Operations Research Letters*, 1(4):139–143, 1982.

[22] M. Cygan, F. Fomin, Ł. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.

[23] J. Darlay, N. Brauner, and J. Moncel. Dense and sparse graph partition. *Discrete Applied Mathematics*, 160(16-17):2389–2396, 2012.

[24] M. Deza and E. Deza. *Encyclopedia of Distances*. Springer, 2013.

[25] R. Diestel. *Graph Theory, 5th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2017.

[26] I. Dinur and D. Steurer. Analytical approach to parallel repetition. In D. B. Shmoys, editor, *Symposium on Theory of Computing, STOC*, pages 624–633. ACM, 2014.

[27] J. Domingo-Ferrer and J. M. Mateo-Sanz. Practical Data-Oriented Microaggregation for Statistical Disclosure Control. *IEEE Transactions on Knowledge and Data Engineering*, 14(1):189–201, 2002.

[28] J. Domingo-Ferrer and F. Sebé. Optimal Multivariate 2-Microaggregation for Microdata Protection: A 2-Approximation. In J. Domingo-Ferrer and L. Franconi, editors, *Privacy in Statistical Databases, PSD'06*, volume 4302 of *LNCS*, pages 129–138. Springer, 2006.

[29] R. G. Downey and M. R. Fellows. Fixed parameter tractability and completeness. *Congressus Numerantium*, 87:161–187, 1992.

[30] R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

[31] J. Edmonds and E. L. Johnson. Matching, Euler tours and the Chinese postman. *Mathematical Programming*, 5:88–124, 1973.

[32] C. Elkan. Using the Triangle Inequality to Accelerate k-Means. In T. Fawcett and N. Mishra, editors, *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21-24, 2003, Washington, DC, USA*, pages 147–153. AAAI Press, 2003.

[33] F. Ergün, R. Kumar, and R. Rubinfeld. Fast Approximate PCPs. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing, May 1-4, 1999, Atlanta, Georgia, USA*, pages 41–50, 1999.

[34] M. Fellows, F. Fomin, D. Lokshtanov, F. Rosamond, S. Saurabh, S. Szeider, and C. Thomassen. On the complexity of some colorful problems parameterized by treewidth. *Information and Computation*, 209(2):143–153, 2011.

[35] M. R. Fellows, S. Gaspers, and F. A. Rosamond. Parameterizing by the number of numbers. *Theoretical Computer Science*, 50(4):675–693, 2012.

[36] M. R. Fellows, D. Hermelin, F. A. Rosamond, and S. Vialette. On the parameterized complexity of multiple-interval graph problems. *Theoretical Computer Science*, 410(1):53–61, 2009.

[37] J. Fiala, P. Golovach, and J. Kratochvíl. Parameterized complexity of coloring problems: Treewidth versus vertex cover. *Theoretical Computer Science*, 412(23):2513–2523, 2011.

[38] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, 2006.

[39] M. Goemans and D. Williamson. A general approximation technique for constrained forest problems. *SIAM Journal on Computing*, 24(2):296–317, 1995.

[40] S. Guha, A. Meyerson, and K. Munagala. Hierarchical placement and network design problems. In *In Proceedings of the 41th Annual IEEE Symposium on Foundations of Computer Science, FOCS'00*, pages 603–612. IEEE Computer Society, 2000.

[41] J. Guo, F. Hüffner, and R. Niedermeier. A structural view on parameterizing problems: Distance from triviality. In R. G. Downey, M. R. Fellows, and F. K. H. A. Dehne, editors, *Parameterized and Exact Computation, First International Workshop, IWPEC 2004, Bergen, Norway, September 14-17, 2004, Proceedings*, volume 3162 of *LNCS*. Springer, 2004.

[42] D. S. Hochbaum. Heuristics for the fixed cost median problem. *Mathematical Programming*, 22(1):148–162, 1982.

[43] C. Imielinska, B. Kalantari, and L. Khachiyan. A greedy heuristic for a minimum-weight forest problem. *Operations Research Letters*, 14(2):65–71, 1993.

[44] A. K. Jain. Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, 31(8):651–666, 2010.

[45] V. King, S. Rao, and R. Tarjan. A faster deterministic maximum flow algorithm. *Journal of Algorithms*, 17(3):447–474, 1994.

[46] S. Li. A 1.488 approximation algorithm for the uncapacitated facility location problem. *Information and Computation*, 222:45–58, 2013.

[47] D. Lokshtanov, D. Marx, and S. Saurabh. Slightly superexponential parameterized problems. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pages 760–776, 2011.

[48] M. Mahajan, P. Nimbhorkar, and K. R. Varadarajan. The planar k-means problem is NP-hard. *Theoretical Computer Science*, 442:13–21, 2012.

[49] D. Marx. Parameterized complexity and approximation algorithms. *The Computer Journal*, 51(1):60–78, 2008.

[50] N. Megiddo and K. J. Supowit. On the Complexity of Some Common Geometric Location Problems. *SIAM Journal on Computing*, 13(1):182–196, 1984.

[51] M.Xiao and S. Kou. Exact algorithms for the maximum dissociation set and minimum 3-path vertex cover problems. *Theoretical Computer Science*, 657, Part A:86–97, 2017.

[52] J. B. Orlin. Max flows in $O(nm)$ time, or better. In *Proceedings of the forty-fifth annual ACM Symposium on Theory of Computing, STOC*, pages 765–774. ACM, 2013.

[53] C. H. Papadimitriou. Worst-case and probabilistic analysis of a geometric location problem. *SIAM Journal on Computing*, 10(3):542–557, 1981.

[54] C. H. Papadimitriou and M. Yannakakis. Optimization, approximation, and complexity classes. *Journal of Computer and System Sciences*, 43:425–440, 1991.

[55] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor. *Recommender Systems Handbook*. Springer, 2nd edition, 2015.

[56] P. Samarati. Protecting respondents' identities in microdata release. *IEEE Transactions on Knowledge and Data Engineering*, 13(6):1010–1027, November 2001.

[57] J. B. Schafer, D. Frankowski, J. L. Herlocker, and S. Sen. Collaborative filtering recommender systems. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, *The Adaptive Web, Methods and Strategies of Web Personalization*, volume 4321 of *LNCS*, pages 291–324. Springer, 2007.

[58] A. Schrijver. *Combinatorial Optimization*. Springer, 2003.

[59] A. Shalita and U. Zwick. Efficient algorithms for the 2-gathering problem. *ACM Transactions on Algorithms*, 6(2), 2010.

[60] Kumar V. Steinbach M., Ertöz L. The challenges of clustering high dimensional data. In Wille L. T., editor, *New Directions in Statistical Physics*. Springer, 2004.

[61] K. Stokes. On computational anonymity. In *Privacy in Statistical Databases - UNESCO Chair in Data Privacy, International Conference, PSD 2012, Palermo, Italy, September 26-28, 2012. Proceedings*, pages 336–347, 2012.

[62] C. Tovey. A Simplified NP-complete Satisfiability Problem. *Discrete Applied Mathematics*, 8(1):85–89, 1984.

[63] M. Wahlström. Exact algorithms for finding minimum transversals in rank-3 hypergraphs. *Journal of Algorithms*, 51(2):107–121, 2004.

[64] M. Xiao and S. Kou. Kernelization and parameterized algorithms for 3-path vertex cover. In *Theory and Applications of Models of Computation - 14th Annual Conference, TAMC 2017, Bern, Switzerland, April 20-22, 2017, Proceedings*, pages 654–668, 2017.

[65] M. Xiao and H. Nagamochi. Exact algorithms for maximum independent set. volume 255, pages 126–146, 2017.

[66] D. Xu, E. Anshelevich, and M. Chiang. On survivable access network design: Complexity and algorithms. In *INFOCOM 2008. 27th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 13-18 April 2008, Phoenix, AZ, USA*, pages 186–190, 2008.

List of Problems

Family of Problems Discussed in the Thesis

For the choices $f \in \{\text{rad,diam,avg}\}$ with the definitions:

$$\text{rad}(P) := \min\{\max\{d(x,y)\colon y \in P\}\colon x \in P\},$$

$$\text{diam}(P) := \max\{\max\{d(x,y)\colon y \in P\}\colon x \in P\},$$

$$\text{avg}(P) := |P|^{-1} \cdot \min\{\textstyle\sum_{y \in P} d(x,y)\colon x \in P\},$$

and $\|\cdot\| \in \{\|\cdot\|_1^w, \|\cdot\|_\infty^w, \|\cdot\|_\infty\}$ with the definitions:

$$\|(f(P_1),\ldots,f(P_s))\|_\infty := \max\{f(P_i)\colon 1 \leq i \leq s\},$$

$$\|(f(P_1),\ldots,f(P_s))\|_\infty^w := \max\{|P_i|f(P_i)\colon 1 \leq i \leq s\},$$

$$\|\cdot(f(P_1),\ldots,f(P_s))\|_1^w := \textstyle\sum_{i=1}^s |P_i|f(P_i),$$

we considered the following problems:

($\|\cdot\|, f$)-k-CLUSTER

Input: Graph $G = (V,E)$ with edge-weight function $w_E\colon E \to \mathbb{Q}_+$, $k \in \mathbb{N}$.

Output: A k-cluster $P_1,\ldots,P_s$ of V for some $s \in \mathbb{N}$, which minimises $\|(f(P_1),\ldots,f(P_s))\|$.

EUCLIDEAN ($\|\cdot\|, f$)-k-CLUSTER

Input: $P \subset \mathbb{R}^w$ finite, $k \in \mathbb{N}$, $D \in \mathbb{R}$.

Question: Is there a k-cluster $P_1,\ldots,P_s$ of P for some $s \in \mathbb{N}$, such that $\|(f(P_1),\ldots,f(P_s))\| \leq D$, where the pairwise distances to compute this objective function is computed by d_w.

Other Problems

P_3-COVER

Input: Graph $G = (V,E)$, $\ell \in \mathbb{N}$.

Parameter: ℓ

Question: Does there exists a subset $\mathcal{F} \subseteq V$ such that the degree of each vertex in $G[V \setminus \mathcal{F}]$ is at most 1?

Cluster Vertex Deletion

Input: Graph $G = (V, E)$, $\ell \in \mathbb{N}$.

Parameter: ℓ

Question: Does there exists a subset $\mathcal{F} \subseteq V$ such that $G[V \setminus \mathcal{F}]$ does not contain P_3 as an induced subgraph?

3-Hitting Set

Input: Hypergraph $H = (V, F)$ such that $|f| = 3$ for all hyperedges $f \in F$, $\ell \in \mathbb{N}$.

Question: Does there exists a subset $\mathcal{C} \subseteq V$ such that $f \cap \mathcal{C} \neq \emptyset$ for all $f \in F$?

Multicoloured Dominating Set

Input: Graph $G = (V, E)$, with vertex partition $V = V_1 \cup \cdots \cup V_\ell$.

Parameter: ℓ

Question: Does there exists a subset $\mathcal{D} \subseteq V$ such that $N[\mathcal{D}] = V$ ($\mathcal{D}$ is a dominating set for G) and $|\mathcal{D} \cap V_i| = 1$ for all $i \in \{1, \ldots, \ell\}$?

List Colouring(τ)

Input: Graph $G = (V, E)$, colours $\{1, \ldots, \ell\}$ and a list of possible colours for each vertex $v \in V$, given by a list $L(v) \subseteq \{1, \ldots, \ell\}$ for each $v \in V$.

Parameter: $\tau(G)$ (cardinality of a minimum vertex cover for G).

Question: Does there exists a colouring $f \colon V \to \{1, \ldots, r\}$ such that $f(v) \in L(v)$ for all $v \in V$ and $f(v) \neq f(w)$ for all $\{v, w\} \in E$?

Simplex Matching

Input: Hypergraph $H = (V, F)$ with $F \subseteq (V^2 \cup V^3)$ and cost-function $c \colon F \to \mathbb{Q}$ satisfying:

1. $\{\{u, v\}, \{v, w\}, \{u, w\}\} \subseteq F$ for all $\{u, v, w\} \in F$. (*subset condition*)
2. $c(\{u, v\}) + c(\{v, w\}) + c(\{u, w\}) \leq 2c(\{u, v, w\})$ for all $\{u, v, w\} \in F$. (*simplex condition*)

Output: A perfect matching of H (that is a set $S \subseteq F$ such that every vertex in V appears in exactly one hyperedge of S) of minimal cost.

Exact-t-Cover

Input: A universe $X = \{x_1, \ldots, x_n\}$ and a collection $C = \{S_1, \ldots, S_r\}$ of subsets of X, such that each S_i, $i \in \{1, \ldots, r\}$, has cardinality t.

Question: Does there exist a subset $C' \subseteq C$ (*exact cover*) that is a partition of X?

Cubic Vertex Cover

Input: Graph $G = (V, E)$ such that all vertices $v \in V$ have degree 3.

Output: A set $C \subseteq V$ (*vertex cover*) of minimum cardinality such that $e \cap C \neq \emptyset$ for all $e \in E$.

$(3,3)$-Satisfiability (or $(3,3)$-SAT)

Input: Boolean formula F in conjunctive normal form such that each clause contains at most 3 literals and each variable occurs both positively and negatively in F and overall at most 3 times.

Question: Does there exist a satisfying assignment for F?

Lower Capacitated Tree Partitioning

Input: Graph $G = (V, E)$, edge-weights $w_E \colon E \to \mathbb{Q}_+$, capacity $k \in \mathbb{N}$.

Output: A set $E' \subseteq E$ minimising $\sum_{e \in E'} w_E(e)$ such that each $v \in V$ occurs in at least one $e \in E'$ and each component in the graph induced by E' is a tree with at least k vertices.

Simplex Cover

Input: Hypergraph $H = (V, F)$ with $F \subseteq (V^2 \cup V^3)$ satisfying the subset condition, i.e., $\{\{u, v\}, \{v, w\}, \{u, w\}\} \subseteq F$ for all $\{u, v, w\} \in F$.

Output: A perfect matching of H.