Resolving Conflicts for Lower-Bounded Clustering

Katrin Casel
Hasso Plattner Institute, University of Potsdam, Potsdam, Germany
Universität Trier, Fachbereich IV - Informatikwissenschaften, Germany
casel@informatik.uni-trier.de

Abstract

This paper considers the effect of non-metric distances for lower-bounded clustering, i.e., the problem of computing a partition for a given set of objects with pairwise distance, such that each set has a certain minimum cardinality (as required for anonymisation or balanced facility location problems). We discuss lower-bounded clustering with the objective to minimise the maximum radius or diameter of the clusters. For these problems there exists a 2-approximation but only if the pairwise distance on the objects satisfies the triangle inequality, without this property no polynomial-time constant factor approximation is possible, unless $\mathsf{P} = \mathsf{NP}$. We try to resolve or at least soften this effect of non-metric distances by devising particular strategies to deal with violations of the triangle inequality (*conflicts*). With parameterised algorithmics, we find that if the number of such conflicts is not too large, constant factor approximations can still be computed efficiently.

In particular, we introduce parameterised approximations with respect to not just the number of conflicts but also for the vertex cover number of the *conflict graph* (graph induced by conflicts). Interestingly, we salvage the approximation ratio of 2 for diameter while for radius it is only possible to show a ratio of 3. For the parameter vertex cover number of the conflict graph this worsening in ratio is shown to be unavoidable, unless $\mathsf{FPT} = \mathsf{W}[2]$. We further discuss improvements for diameter by choosing the (induced) $\mathcal{P}_3$-cover number of the conflict graph as parameter and complement these by showing that, unless $\mathsf{FPT} = \mathsf{W}[1]$, there exists no constant factor parameterised approximation with respect to the parameter split vertex deletion set.

2012 ACM Subject Classification Theory of computation: Approximation algorithms analysis, Parameterized complexity and exact algorithms, Facility location and clustering

Keywords and phrases clustering, triangle inequality, parameterised approximation

Digital Object Identifier 10.4230/LIPIcs.IPEC.2018.23

1 Introduction

For most clustering problems, the quality of a solution is usually assessed with respect to a given pairwise distance on the input objects. Approximate solutions for such tasks often rely on this distance to be a metric. But what happens if this property does not hold? Many clustering problems are much more difficult to solve or even approximate if the pairwise distance violates the triangle inequality. The problem Uncapacitated Facility Location, for example, can be approximated with ratio 1.488 if restricted to metric instances, see [20]. For general, possibly non-metric, distances, it is only possible to compute a $\log(n)$-approximation; see [18] for one of many algorithms with this performance. The relation to the Set Cover problem 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 [10], 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.


© Katrin Casel;
licensed under Creative Commons License CC-BY

Such helpful consequences of a restriction to triangle inequality have lead to many approaches which assume that this property holds. Another example of this kind is given in [13], where the properties that come with a restriction to distances which satisfy the triangle inequality are used to speed up the famous heuristic algorithm *k-means*, named after the clustering problem it is designed to approximate efficiently.

For many applications, the requirement that the associated distance is a metric seems to be pretty natural and is not really considered a restriction. For lower-bounded clustering, the problem of computing a partition for a given set of objects with pairwise distance such that each set has a certain minimum cardinality, we also made this assumption in order to enable approximation algorithms with a provable performance ratio, see [1]. In particular, with the objective to minimise the maximum radius or diameter of the clusters, it turned out that, unless $\mathsf{P} = \mathsf{NP}$, there exists no polynomial-time constant factor approximation if the pairwise distance on the objects violates the triangle inequality while a ratio of 2 can be achieved without such violations. With an attempt to use this problem to model a clustering which can be used for recommender systems, we however found that the pairwise distance does not in general satisfy the triangle inequality. The so-called *Pearson distance*, which is usually used for recommendations, does not have this useful property and, as also observed in [23], practical instances actually show this non-metric behaviour. Such unfortunate situations seem to be unavoidable when it comes to human preferences which raises the question of how the resulting negative effects can be avoided, or at least controlled.

One option that comes to mind concerning non-metric instances in general is editing, i.e., a pre-processing step which tries to transform a given instance, with preferably few changes, such that triangle inequality holds and known algorithms for such well-behaved instances can then be applied. This idea however has several drawbacks. Changes to a given instance always come at the price of distortion; altering distances or even deleting objects results in perturbation of the original input. This effect hence raises the task to find alterations which bring as little change to the original instance as possible. Such editing problems are then usually already difficult to solve themselves. In our specific case of lower-bounded clustering, the task to find a minimum number of vertices such that their removal from a given instance deletes all violations of the triangle inequality is closely related to the 3-hitting set problem. But much more troublesome than the complexity of computing such minimal alterations for these types of problems is the danger of accidentally worsening the optimum value with vertex deletion. Observe that by requiring a lower bound on the cardinality of the clusters, the optimum value is not necessarily monotone, in the sense that a larger set of input objects might enable a better solution.

Here, we therefore seek a different approach which employs extra treatment for violations of the triangle inequality within the approximations designed for metric instances. The basic idea is to investigate the consequences of violations and devise strategies to deal with those within moderate exponential time depending on, roughly speaking, how much the given pairwise distance differs from a metric. More precisely, we will, for a pairwise distance d, look at the set of pairs $\{u, v\}$ which directly violate the triangle inequality, i.e., there exists another object x such that $d(u, v) > d(u, x) + d(v, x)$. We call such pairs *conflicts*. 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 for our purposes. Our strategy then is to alter the algorithms for metric instances in such a way that they also yield constant-factor approximations for non-metric instances while only spending exponential effort with respect to the 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 [17]. The idea there is to define some *distance* which specifies how much a given instance differs from some structural property which makes it 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.

We discuss conflicts for the problems of minimising the maximum radius or diameter for lower-bounded clustering. We first develop parameterised approximations with respect to conflicts and then improve those to only require exponential time with respect to the vertex cover number of the *conflict graph* G_c (the graph induced by conflicts interpreted as edges). For diameter, we then consider even smaller parameters given by the size of an (induced) $\mathcal{P}_3$-cover of G_c, but conversely show that the even smaller size of a split vertex deletion set for G_c is not a suitable parameter. Curiously, we find that while the ratio of 2 remains for diameter, it is only possible to prove a ratio of 3 for radius. For the parameter vertex cover of G_c, we prove that this worsening is unavoidable under the assumption that $\mathsf{FPT} \neq \mathsf{W}[2]$.

2 Preliminaries

We mostly use standard notation and refer to textbooks such as [6] for graph theory, [3] for approximation algorithms and [12] for parameterised complexity terminology.

When estimating running times we use $\mathcal{O}^*$-notation which, compared to $\mathcal{O}$-notation, also suppresses factors which are polynomial in the input size. We use B_n to denote the nth *Bell number* which is the number of partitions of a set of size n.

The algorithms discussed here combine parameterisation and approximation and fall into the category of *fpt-approximation algorithms with parameter* κ, as discussed for example in [22], i.e., approximation algorithms with provable ratio and running time in $\mathcal{O}(g(\kappa)\cdot f(n))$, for computable function g and polynomial f, so $\mathcal{O}^*(g(\kappa))$ in $\mathcal{O}^*$-notation. For lower bounds in this context, there does not seem to exist a unified notation, so for these kinds of results, we will not use hardness notions but link the existence of certain parameterised approximations to the (unlikely) equivalence of certain complexity classes.

2.1 Problem Definition

As formal model for lower-bounded clustering we use the abstract problem $(\|\cdot\|, f)$-k-CLUSTER from [1]. An instance of this problem is an undirected graph $G = (V, E)$ with edge-weights $w_E\colon E \to \mathbb{R}_+$ and a lower bound $k \in \mathbb{N}$. A feasible solution is any partition $P_1, \dots, P_s$ of V such that $|P_i| \geq k$ for all $i \in \{1, \dots, s\}$, we refer to such a partition as *k-cluster*.

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 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 the *induced distance* d derived from weights only known for a set E of given edges models missing information about pairwise distances.

For the objectives to minimise maximum radius or diameter we formally define $\mathrm{rad}(P) := \min\{\max\{d(x, y)\colon y \in P\}\colon x \in P\}$ and $\mathrm{diam}(P) := \max\{\max\{d(x, y)\colon y \in P\}\colon x \in P\}$. The resulting problems $(\|\cdot\|_\infty, \mathrm{rad})$-$k$-CLUSTER and $(\|\cdot\|_\infty, \mathrm{diam})$-$k$-CLUSTER ask for a k-cluster $P_1, \dots, P_s$ minimising $\max\{\mathrm{rad}(P_i)\colon 1 \leq i \leq s\}$ and $\max\{\mathrm{diam}(P_i)\colon 1 \leq i \leq s\}$, respectively. A vertex $x \in V$ with $\max\{d(x, y)\colon y \in P\} = \mathrm{rad}(P)$ is called *central* for P.

Figure 1 In the graph on the left, $\{c, d\}$ has a weight larger than the shortest path from c to d but is not in C. Lowering the weights on $\{a, d\}$ and $\{b, c\}$ to 2 to remove these conflicts turns $\{c, d\}$ into a conflict. The graph on the right has a 2-cluster of maximum radius and diameter 1. Removing a to delete conflicts results in a graph for which only a trivial 2-cluster of radius Δ is possible.

2.2 Conflicts

A function $d\colon V \times V \to \mathbb{R}^+$ satisfies the *triangle inequality* if $d(u, v) \leq d(u, w) + d(w, v)$ for all $u, v, w \in V$. We will call an instance of $(\|\cdot\|, f)$-k-CLUSTER *metric* if the induced distance satisfies the triangle inequality although 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, d derived from edge-weights is just non-negative, symmetric and reflexive.

Observe that our definition allows non-metric instances, see Figure 1. We define the set of *conflicts* for an instance (G, k) with $G = (V, E)$ and induced distance d as the collection C of vertex pairs $\{u, v\}$ such that the triangle inequality is violated for u and v, formally:

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

One curious property is that the 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 1), it however is always a subset of E. Considering the option of weight reduction to achieve triangle inequality, C might be smaller than the set of edge-weights which have to be adjusted in order to arrive at a graph without conflicts. Figure 1 also gives a small example which illustrates why vertex removal can be a dangerous editing step for problems with non-monotone objective function such as lower bounded clustering. Observe how the optimum value may increase arbitrarily from the original graph to a graph created by deleting vertices to avoid conflicts. This effect is another reason to favour parameterisation over editing.

Actually, we will mostly consider parameterisation by the cardinality of the set P of *conflict vertices*, which simply are the vertices involved in a conflict, formally defined by:

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

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 parameters are related by the inequalities $p \leq 2c \leq (p(p-1))$. For the concrete running times, it is however still relevant to distinguish between p and c as the bounds given by this inequality are sharp.

We further refer to the graph $G_c = (P, C)$ as *conflict graph*; observe that G_c is always a subgraph of the input graph. The structure of the conflict graph reflects the entanglement of conflicts in a given instance. When designing algorithms which devise specific strategies to resolve conflicts, it is not surprising that the relations between these can be exploited for improvement. The structure of the conflict graph hence yields further possibilities for a parameterisation which measures the distance to a metric instance.

3 Parameterisation by Conflict Vertices

Without restriction to metric instances, there exists no constant factor approximation in polynomial time for $(\|\cdot\|_\infty, \text{rad})$- or $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER, unless P = NP (see Proposition 6 from [1]). The 2-approximation presented for these problems in [1], Theorem 9 hence highly relies on the assumption that the input instance is metric. In short, on input $((V, E), w_E, k)$ with induced distance d, the approximation algorithm with the performance ratio of 2 for both $(\|\cdot\|_\infty, \text{rad})$- and $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER simply performs a binary search for the smallest value D for which the following 2-step greedy procedure is successful:

step (1) While V is not empty, pick some $c \in V$, build the set $P(c) := \{w \in V : d(c, w) \leq D\}$ and set $V = V \setminus P(c)$.

step (2) Let $P(c_1), \ldots, P(c_s)$ be the partition built in step (1). Try to create from this a k-cluster $P'(c_1), \ldots, P'(c_s)$ such that $c_i \in P'(c_i)$ and $d(v, c_i) \leq D$ for all $v \in P'(c_i)$, $i \in \{1, \ldots, s\}$. (Such a partition can be efficiently computed with the help of a network flow formulation over vertices $V \cup \{s, t\}$. With arcs of capacity 1 from s to every $v \in V$, arcs of capacity k from c_i to t and arcs of capacity 1 from $v \in V$ to c_i if $d(v, c_i) \leq D$ for each $i \in \{1, \ldots, s\}$. A flow of value sk in this network interpreted as moving a vertex $v \in P(c_i)$ into the set $P(c_j)$ if and only if the flow uses the arc from v to c_j, gives a polynomial procedure to create a k-cluster.)

The ratio of 2 for $(\|\cdot\|_\infty, \text{rad})$- and $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER can be shown by proving that the greedy procedure is successful if D is fixed to be twice the optimum radius or the optimum diameter, respectively, which is due to the following two properties for this choice of D:

(i) By the choice of the vertices c_i in step (1), it follows that $d(c_i, c_j) > D$ for all $i \neq j$, which means that c_i and c_j belong to different sets in an optimum solution. So, for each i there exist enough vertices at distance at most D from c_i to be put into $P'(c_i)$ in step (2).

(ii) If the greedy procedure is successful, the construction immediately yields $d(v, c_i) \leq D$ for each $v \in P'(c_i)$, so the resulting k-cluster $P'(c_1), \ldots, P'(c_s)$ has maximum radius D and maximum diameter $2D$.

The above properties however do not hold in case the input instance is not metric. More precisely, for each objective function (radius, diameter), one of them is no longer true. For radius, property (i) fails, as without triangle inequality, vertices at distance more than twice the optimum can still be contained in the same cluster in an optimum solution. For diameter, property (ii) fails, as a radius of D does no longer guarantee a diameter of $2D$.

We will now try to salvage these properties for non-metric instances by adding exponential effort with respect to conflicts to the above approximation procedure. Given that different properties are lost for the two objective functions, it is not surprising that this approach yields two different parameterised approximations, the basic algorithmic idea of fixing a maximum radius D, building a preliminary clustering with clusters of radius D and then balancing the cardinalities with a network however always remains and we will only sketch the crucial points which have to be adjusted in each case.

At first, we observe that starting from the approximation algorithm for metric instances, it is not too hard to see that a constant number of conflicts does not yield too much trouble. More precisely, we simply guess a suitable central vertex for each conflict vertex and fix the resulting partition of P and centres for step (1). This optimal guessing for the problematic vertices resolves the problems for both objective functions, as the fixed centres are optimal by choice and the greedy algorithm is only responsible for partitioning remaining vertices in $V \setminus P$ for which the triangle inequality holds. This kind of guessing yields:

▶ **Theorem 1.** *A 2-approximation for* $(\|\cdot\|_\infty, \text{rad})$- *and* $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(n^p)$.

This result raises the question whether an improvement to a more efficient running time is possible, i.e., some constant factor parameterised approximation. To this end, we want to mention that an improvement of the approximation ratio of 2 is unlikely, as this is already shown to imply $\mathsf{P} = \mathsf{NP}$ for metric instances in [1], Corollaries 1 and 3.

For diameter, we have to only be careful with property (ii) which means that in the above guessing, we did not really require the knowledge of centres but only the partition they imply on the vertices in P. In fact, it can be shown that knowing this partition is enough, and simply enforcing it in the metric approximation algorithm still yields a 2-approximation for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER. The cost of trying all partitions of P can be estimated with the Bell number (which can be bounded by $B_n < \left(\frac{0.792n}{\log(n+1)}\right)^n$, see [5]) and yields:

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

For radius, the fixed centres are important, but by compromising a little on the approximation ratio it is still possible to design a parameterised approximation with parameter p. This algorithm only requires guessing which vertices in P are central in an optimal solution and accordingly forcing step (1) to build clusters around those first; observe that this knowledge salvages property (i). Picking a suitable cluster for the vertices in P which are not chosen to be central however blows up the approximation ratio to 3 (an effect which is explained later in connection to the lower bounds), which yields:

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

4 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 this section, we want to focus on strategies to only spend exponential time for vertices in a subset of P. More precisely, as advertised in the section-title, we will consider parameterisation by structural parameters of the conflict graph.

4.1 Vertex Cover

Looking closer at the problems caused by the conflicts in C, it is not necessary to consider all vertices in P but it appears to be sufficient to pick a subset which covers all conflicts. Formally, this idea translates into parameterisation by a vertex cover for the conflict graph $G_c = (P, C)$. 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, a first easy observation is that, at least for $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER, instances for which p_c is a constant can be approximated efficiently. In fact, we can simply switch from the set P to a minimum vertex cover for G_c in the procedure discussed for Theorem 1, as property (i) can only be violated if the algorithm can choose two vertices of a conflict as centres. The additionally required computing of a constant size vertex cover for G_c is not an expensive task, so this idea immediately yields:

▶ **Theorem 4.** *A 2-approximation for* $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(n^{p_c})$.

The parameterised approximation for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER from Section 3 requires little algorithmic adjustment to switch from parameter p to parameter p_c. Proving correctness of the given procedure, i.e., guaranteeing a performance ratio, is however more complicated.

Consider only guessing a partition of a vertex cover $\mathcal{V}$ of the conflict graph and fixing this partition for step (1). The only further change required to make sure that property (ii) remains true, is to adjust the algorithm when building clusters including the guessed sets $V_1, \ldots, V_t$ in the partition of $\mathcal{V}$. Now, both when building the preliminary clusters including some V_i in step (1) and moving vertices in step (2) into such a cluster, we not just require a distance of at most D to one chosen centre but to all vertices in V_i. Then all distances in a set of the resulting k-cluster involving vertices in the cover $\mathcal{V}$ are bounded by D. As $\mathcal{V}$ is a vertex cover of G_c, distances not involving a vertex in $\mathcal{V}$ are not a conflict which means triangle inequality can be used for all remaining cases to prove that property (ii) still holds.

This approach requires computing a vertex cover for the conflict graph, a problem which can be solved by [9] with a running time in $\mathcal{O}^*(1.2738^{p_c})$. As this single-exponential effort is only performed once in the beginning and dominated by the Bell number, we can conclude:

▶ **Theorem 5.** *A 2-approximation for* $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(B_{p_c})$.

For $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER, it is not obvious how to reduce p in Theorem 3 to p_c. While knowing which vertices in a vertex cover are central is sufficient to avoid picking centres from the same cluster in step (1), the problem is finding a suitable cluster for the vertices from the vertex cover which are not central. In particular, this is a problem when two vertices from the vertex cover which are not central are wrongfully put in the same cluster; more precisely, if two vertices $u, v \in \mathcal{V}$ are not central in any optimal solution but belong to two different clusters, with centres $c_u, c_v \notin \mathcal{V}$, while these two correct centres are put into $P(c)$ with $\{u, c\}, \{v, c\} \in C$ in step (1) of our algorithm. In such a case there is no general clean way to identify how to split up $P(c)$ into sets of cardinality at least k and such that u and v can be assigned at a radius which can in any way be bounded by the optimum.

With a more involved algorithm which additionally guesses a partition, like for the diameter measure, it is possible to find an approximation for $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER parameterised by p_c. After guessing a partition and a set of centres for the vertex cover, this approach further requires computing a suitable centre for each set in the partition before running steps (1) and (2), which are altered to respect the fixed partition and central vertices. This just means finding for each set V_i in the partition of the vertex cover $\mathcal{V}$ for which no vertex in V_i is fixed as center, a vertex $v \in V \setminus \mathcal{V}$ such that $d(v, w) \leq \frac{D}{2}$ for all $w \in V_i$ (where $\frac{D}{2}$ relates to the optimum radius). As two sets V_i and V_j might compete over such candidates for centres, we use maximum matching to enable finding, in case our fixed guesses are correct, a centre for each set V_i. These adjustments are sufficient to salvage property (i). Although now there is no longer the problem of finding a suitable cluster for vertices which are not allowed to be central, the performance ratio is still only 3, as for this procedure the worst-case now comes from choosing a wrong center $c' \in V \setminus \mathcal{V}$ for some V_i: If $P \subseteq V$ is the cluster containing V_i in an optimum solution and c is its correct center, it follows that the distance of $v \in P$ to the wrong center c' can only be bounded by $d(v, c') \leq d(v, c) + d(c, c') \leq \frac{D}{2} + d(c, v_i) + d(c', v_i) \leq 3\frac{D}{2}$ (using some $v_i \in V_i$ for this equation). The asymptotic running time of this approach is dominated by guessing the partition and centre-choice for a minimum vertex cover of G_c, so:

▶ **Theorem 6.** *A 3-approximation for* $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(2^{p_c} \cdot B_{p_c})$.

4.2 $\mathcal{P}_3$-Covers

With more changes to the algorithms discussed for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER, it is possible to further reduce the size of the subset of P which requires the expensive guessing of the partition. When building the first partition in step (1), 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. The network used to model the reassignment of vertices according to the fixed centres 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 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 edges, or equivalently a graph which does not contain any path of length 2 usually denoted $\mathcal{P}_3$, is smaller than the vertex cover of the conflict graph (unless the instance is initially metric). Formally, such a set is called a *$\mathcal{P}_3$-cover*. In the following, we use p_{3c} to denote the cardinality of a smallest $\mathcal{P}_3$-cover for G_c and with the parameterised algorithm from [24], such a set can be computed with a running time in $\mathcal{O}^*(1.7485^{p_{3c}})$. Using the more expensive strategy of guessing the correct partition only for a minimum $\mathcal{P}_3$-cover of the conflict graph computed with the algorithm in [24], branching on the remaining isolated conflicts for the pre-clustering and modifying the network to avoid conflicts as described above gives the following result; observe that the number of remaining conflicts is bounded by $\frac{p}{2}$:

▶ **Theorem 7.** *A 2-approximation for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER can be computed with a running time in $\mathcal{O}^*(\sqrt{2}^p \cdot B_{p_{3c}})$.*

As already mentioned, branching on conflicts in step (1) 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 if 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 called Induced $\mathcal{P}_3$-Cover or Cluster Vertex Deletion.

It is possible to, in a sense, generalise the algorithm for Theorem 7 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, size of a 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 $\frac{1}{2}|P|$. A minimum cluster vertex deletion set for G_c can be computed in time $\mathcal{O}^*(1.9102^{p_{3d}})$ with the help of the parameterised algorithm in [7], so again a negligible effort compared to checking all partitions, and this idea hence yields the following result:

p_c	p_{3c}	p_{3d}
$\mathcal{O}^*(B_{p_c})$ (Theorem 5)	$\mathcal{O}^*(\sqrt{2}^p B_{p_{3c}})$ (Theorem 7)	$\mathcal{O}^*(2^c B_{p_{3d}})$ (Theorem 8)

Table 1 Summary of the running time of the parameterised 2-approximation for $(\|\cdot\|_\infty, \text{diam})$-$k$-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.

▶ **Theorem 8.** *A 2-approximation for* $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER *can be computed with a running time in* $\mathcal{O}^*(2^c \cdot B_{p_{3d}})$.

The results to improve the parameterised approximation from Theorem 5 are presented here 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 7 and 8 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 2-approximations for $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER with respect to the structural parameters of the conflict graph discussed in this section is given in Table 1.

5 Lower Bounds

In this section, we investigate the limitations of parameterised approximation for lower bounded clustering with structural parameters of the conflict graph. Especially the increase from ratio 2 for metric instances to ratio 3 for non-metric instances for $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER appears strange. We will however see that ratio 3 is, under certain complexity theoretic assumptions, optimal for parameter p_c and that the approach we use to design parameterised approximations is generally limited to the performance ratio of 3 for radius. Further, we will discuss the limits of choosing structural parameters for $(\|\cdot\|_\infty, \text{diam})$-$k$-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 does not seem to allow for a constant factor parameterised approximation.

For the negative results of this section, we use a kind of reduction which links the existence of a parameterised approximation with certain ratio to a parameterised algorithm for a problem which is believed not to be in FPT. Such an algorithm can be seen as *fpt gap-reduction* as introduced in [4]. As the problem to reduce from for $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER , we use MULTICOLOURED DOMINATING SET, which asks for input graph $G = (V, E)$ with vertex partition $V = V_1, \ldots, V_\ell$ for the existence of 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 [2] shows that the W[2]-hardness of classical MINIMUM DOMINATING SET, which is shown in [11], transfers to this restricted version we called *multicoloured* in reference to MULTICOLOURED CLIQUE and the corresponding reduction technique introduced in [15].

We will in the following sketch a reduction from MULTICOLOURED DOMINATING SET to $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER which shows that a parameterised approximation with parameter p_c for the clustering problem could be used to show fixed parameter tractability of the W[2]-hard domination problem. This kind of reduction yields a lower bound for parameterised approximation of $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER under the assumption FPT $\neq$ W[2].

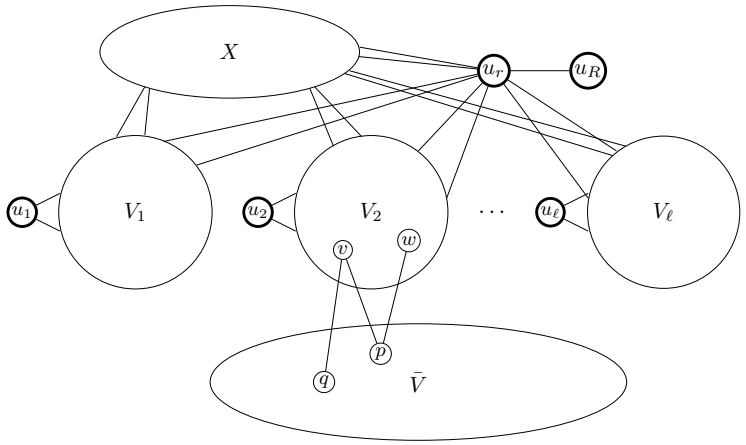

Figure 2 Illustration of the reduction used for Theorem 9, vertices in the vertex cover for the conflict graph drawn with thick border.

For an instance $G = (V, E)$ with $V = V_1, \ldots, V_\ell$ and $|V| = n$ of Multicoloured Dominating Set, we construct an instance $((V', E'), w_{E'}, n+2)$ of $(\|\cdot\|_\infty, \text{rad})$-$k$-cluster with vertex set V' containing V, 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 $u_1, \ldots, u_\ell$ and u_r, u_R and an additional set X of $(\ell-1)n + \ell$ vertices. These vertices are connected with the following edges of weight 1 (see also the illustrated in Figure 9):

- $\{v, \bar{w}\}, \{\bar{w}, v\} \in E'$ iff $\{v, w\} \in E$ (these model the structure of the original graph),
- $\{u_i, v\} \in E'$ for all $v \in V_i$, $i \in \{1, \ldots, \ell\}$ (these force to pick one center from each V_i),
- $\{v, x\} \in E'$ for all $x \in X$, $v \in V \cup \{u_r\}$ (enables balancing cardinalities with the set X),
- $\{u_r, v\} \in E'$ for all $v \in V$ and $\{u_r, u_R\} \in E'$ (forces u_r to be central and allows to assign $v \in V$ not picked for the dominating set at radius 1 to the corresponding cluster).

Further E' contains all other edges involving the vertices $u_1, \ldots, u_\ell$ and u_r, u_R with weight 3 which clearly makes this set of $\ell + 2$ vertices a vertex cover of the conflict graph for the resulting $(\|\cdot\|_\infty, \text{rad})$-$k$-cluster instance $((V', E'), w_{E'}, n + 2)$.

Assuming the existence of a parameterised approximation with parameter p_c and ratio better than 3 for $(\|\cdot\|_\infty, \text{rad})$-$k$-cluster, this algorithm would be able to build clusters with maximum radius less than 3 for $((V', E'), w_{E'}, n+2)$ if and only if there exists a multicoloured dominating set for $G = (V, E)$; more precisely, the centres of the clusters containing the vertices $u_1, \ldots, u_\ell$ in a k-cluster of maximum radius less than 3 for $((V', E'), w_{E'}, n + 2)$ correspond to a multicoloured dominating set for $G = (V, E)$. To understand why this is true, observe that the lower bound $n + 2$ only allows to build at most $\ell + 1$ clusters, while a maximum radius of less than 3 requires at least one cluster with u_r and one cluster with a vertex from $V_i \cup \{u_i\}$ for each $i \in \{1, \ldots, \ell\}$ as centre. A vertex $\bar{V}$ can then only be at distance less than 3 from a centre if the corresponding vertex in the original graph G is adjacent to this centre. As this parameterised approximation has a running-time in $\mathcal{O}^*(g(p_c))$ for some computable function g, and p_c is bounded by $\ell + 2$, the given polynomial construction of $((V', E'), w_{E'}, n + 2)$ combined with this assumed parameterised approximation could be used to solve Multicoloured Dominating Set in $\mathcal{O}^*(g(\ell))$, which formally yields:

▶ **Theorem 9.** *There exists no $(3 - \varepsilon)$-approximation for $(\|\cdot\|_\infty, \text{rad})$-$k$-cluster with a running time in $\mathcal{O}^*(g(p_c))$ for any $\varepsilon > 0$ and computable function g, unless* FPT = W[2].

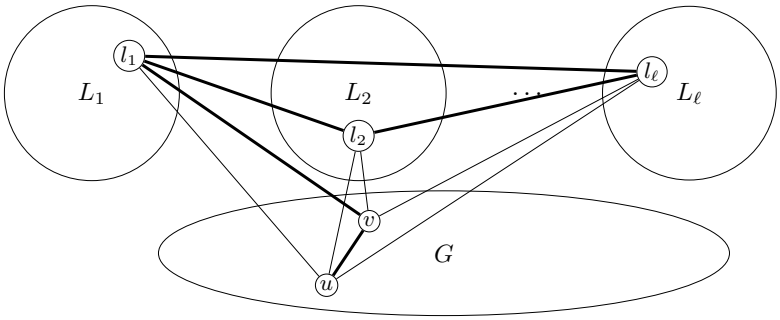

Figure 3 Illustration of the reduction used for Theorem 10, bold lines represent conflict edges with weight $r+1$, all other edges in the complete graph have weight 1. Vertices $u, v \in V$ are such that $\{u,v\} \in E$, $1 \in L(u)$, $1 \notin L(v)$, $2 \in L(u) \cap L(v)$ and $\ell \notin L(u) \cup L(v)$.

The reduction used to prove Theorem 9 also illustrates why our parameterised approximation for $(\|\cdot\|_\infty, \text{rad})$-$k$-CLUSTER with parameter p does not have a performance ratio better than 3. The situation illustrated in Figure 2 is also a case where our algorithmic strategy fails; if the algorithm chooses w instead of v as central vertex in step (1) (observe that these two are both not in the set P), then q is one of the vertices in $P \setminus P'$ (in fact $P' = \emptyset$ is the only correct choice) which has to be put into a suitable cluster without choosing it as centre, which places it at the worst possible distance (3 times the optimum) from the central vertex.

For diameter, we want to investigate the limits of choosing smaller sets of vertices which require partitioning in our parameterised approximations. 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 from the problem LIST COLOURING, which for a given graph $G = (V, E)$, colours $\{1, \ldots, \ell\}$ and a set of possible colours for each vertex $v \in V$, given by a list $L(v) \subseteq \{1, \ldots, \ell\}$ for each $v \in V$ asks if 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 $\{v, w\} \in E$. LIST COLOURING with parameter $\tau(G)$ (vertex cover number of G) is W[1]-hard, see [14, 16].

Our reduction from LIST COLOURING to $(\|\cdot\|_\infty, \text{diam})$-$k$-CLUSTER constructs a clustering instance $((V', E'), w_{E'}, n+2)$ as sketched in Figure 3, where G is the input graph for LIST COLOURING and each L_i, is a set of $n+2$ new vertices. Observe that a vertex cover for G is a split vertex deletion set of the conflict graph for this instance. It can be shown that a "yes"-instance for LIST COLOURING corresponds to a k-cluster of maximum diameter 1 while a "no"-instance yields a diameter of at least $r+1$.

This would enable solving List Colouring in $\mathcal{O}^*(g(\tau))$ if there existed a parameterised approximation with ratio r and cluster vertex deletion set as parameter for $(\|\cdot\|_\infty, \text{diam})$-$k$-cluster, hence:

▶ **Theorem 10.** *There exists no constant factor approximation for* $(\|\cdot\|_\infty, \text{diam})$-$k$-cluster *with a running time in* $\mathcal{O}^*(f(p_s))$*, for any computable function* f*, unless* FPT = W[1].

6 Conclusions

As the exponential time hypothesis implies FPT $\neq$ W[1] by [8], the negative results in this paper especially hold assuming the *exponential time hypothesis* [19]. Both reductions used to show these 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) would help overcome the negative results. For the greedy strategies used for the parameterised approximations in this paper, it is not clear how k could be included in a useful way. An improvement with parameterisation by both conflicts and k probably means using a very different algorithmic approach. The gap between our positive results and the presented lower bounds further suggests room for improvement. Stronger lower bounds seem to require new techniques for reductions which consider both parameterisation and approximation.

We would like to mention that a relaxation of the cardinality constraint, i.e., asking for an approximate solution in the sense that this partition is allowed to contain clusters with only αk vertices, for some factor $0 < \alpha \leq 1$, does not help with the problems caused by conflicts. In fact, the inapproximability for general non-metric instances from [1] holds for fixed values of k with $k \geq 3$, which means that this kind of additional cardinality relaxation either yields a polynomial time solvable problem, in case $\alpha k \leq 2$, or a problem with the same approximation hardness.

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 need a more careful analysis. A concrete question in this regard is whether it is possible to improve the parameterised approximations for $(\|\cdot\|_\infty, \text{diam})$-$k$-cluster to only require single-exponential time. In this regard it would be very interesting to see if slightly superexponential lower bounds as shown in [21] can be proven for a 2-approximation of $(\|\cdot\|_\infty, \text{diam})$-$k$-cluster with parameter p. For improvement on the running time on the other hand, it might be interesting to analyse the algorithms with a closer look at the enumerations of the partitions of P. We always estimated this with the Bell number although we only consider partitions with specific properties (those which are possible in a clustering of maximum diameter D) which in a sense relate to colourings of the conflict graph. It might be possible to enumerate the relevant partitions of P more efficiently with the help of colouring strategies.

Aside from the problems discussed here, 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 might provide a useful way to approach these problems as well.

Acknowledgements

I would like to thank Lorik Dumani for his invaluable help with implementing and testing the algorithms in this paper. This work was supported by the DFG, grant FE 560/6-1.

References

1 F. N. Abu-Khzam, C. Bazgan, K. Casel, and H. Fernau. Clustering with lower-bounded sizes - A general graph-theoretic framework. *Algorithmica*, 80(9):2517–2550, 2018.

2 N. Alon, R. Yuster, and U. Zwick. Color coding. In M.-Y. Kao, editor, *Encyclopedia of Algorithms*. Springer, 2008.

3 G. Ausiello. *Complexity and approximation: combinatorial optimization problems and their approximability properties*. Springer, 1999.

4 C. Bazgan, M. Chopin, A. Nichterlein, and F. Sikora. Parameterized inapproximability of target set selection and generalizations. *Computability*, 3(2):135–145, 2014.

5 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.

6 B. Bollobás. *Modern Graph Theory*, volume 184 of *Graduate texts in mathematics*. Springer, 1998.

7 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.

8 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.

9 J. Chen, I. A. Kanj, and G. Xia. Improved upper bounds for vertex cover. *Theoretical Computer Science*, 411(40–42):3736–3756, 2010.

10 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.

11 R. G. Downey and M. Fellows. Fixed parameter tractability and completeness. *Congressus Numerantium*, 87:161–187, 1992.

12 R. G. Downey and M. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

13 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.

14 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.

15 M. 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.

16 J. Fiala, P. Golovach, and J. Kratochvíl. Parameterized complexity of coloring problems: Treewidth versus vertex cover. *Theoretical Computer Science*, 412(23):2513–2523, 2011.

17 J. Guo, F. Hüffner, and R. Niedermeier. A structural view on parameterizing problems: Distance from triviality. In R. G. Downey, M. 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.

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

19 R. Impagliazzo and R. Paturi. On the complexity of k-sat. *Journal of Computer and System Sciences*, 62(2):367–375, 2001.

20 S. Li. A 1.488 approximation algorithm for the uncapacitated facility location problem. *Information and Computation*, 222:45–58, 2013.

21 D. Lokshtanov, D. Marx, and S. Saurabh. Slightly superexponential parameterized problems. *SIAM J. Comput.*, 47(3):675–702, 2018.

22 D. Marx. Parameterized complexity and approximation algorithms. *The Computer Journal*, 51(1):60–78, 2008.

23 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.

24 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.