



On the k -colored Rainbow Sets in Fixed Dimensions

Vahideh Keikha¹ [✉], Hamidreza Keikha², and Ali Mohades³

¹ Institute of Computer Science, The Czech Academy of Sciences,
Pod Vodárenskou věží 2, 182 07 Prague, Czech Republic
keikha@cs.cas.cz

² Department of Computer Engineering, Amirkabir University of Technology,
Tehran, Iran
keikha@aut.ac.ir

³ Department of Mathematics and Computer Science, Amirkabir University
of Technology, Tehran, Iran
mohades@aut.ac.ir

Abstract. In this paper, we introduce a variant of the minimum diameter color spanning set (MDCSS) problem. Let P be a set of n points of m colors in \mathbb{R}^d . For a given k , our objective is to find a set with k points of different colors that admits the minimum possible diameter. Such a set is called a k -rainbow set. This problem has applications in database queries, mostly composed by weighted points (i.e., a positive value is assigned to each point as its weight), and seeking a maximum weight k -rainbow set. We first assume the points have equal weight and design an FPT algorithm, which we generalize to the weighted version. We also solve the decision and the enumeration version of the problem by introducing a reduction to all maximal independent sets of a bipartite graph. We also introduce a 1.154-approximation algorithm for this problem and a 2.236-approximation for the enumeration version, and we perform some experimental studies on a real data-set, as well as providing several analyses of the data-set based on the outputs of our algorithm. Our exact algorithms and the approximation algorithm for the enumeration problem have a complexity being near-linear to n in \mathbb{R}^2 .

Keywords: Minimum diameter color spanning set · FPT algorithms · Colored points

1 Introduction

Let $P = \{p_1, \dots, p_n\}$ be a set of n points in \mathbb{R}^d , the *diameter* of P is defined as $\text{diameter}(P) = \max_{p_i, p_j \in P} d(p_i, p_j)$, and can be computed in $O(dn^2)$ time [26]. In \mathbb{R}^2 , computing the diameter takes $O(n \log n)$ time [28]. Now, suppose each $p_i \in P$ is assigned a color. The objective of the minimum diameter color spanning set (MDCSS) problem is to find a subset $P^* \subseteq P$ that contains one point from each color, and P^* has the smallest possible diameter among all choices of P^* , where

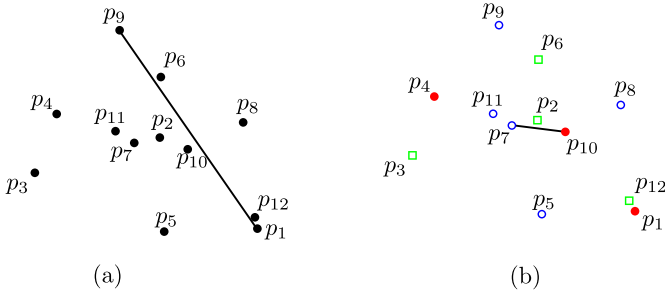


Fig. 1. (a) The diameter of a set P of points in \mathbb{R}^2 . (b) For a set P with $m = 3$ colors, the rainbow set $P^* = \{p_2, p_7, p_{10}\}$.

the diameter is the maximum distance between any two points in P^* . P^* is called the *color spanning set* or the *rainbow set*; see Fig. 1.

The MDCSS problem can be considered as a database query; consider a spatial database where each tuple is associated with a *keyword* or, equivalently, a *color code* in our setting. The m -closest keywords query is the problem of finding the m tuples that match all the keywords chosen by the customer [29]. In our problem, the closeness is measured by the diameter. Now suppose a customer aims at finding some closest keywords of the desired number and his/her maximum willingness. The motivation behind this study is efficiently answering such queries. We note that such queries are introduced in the database literature as *reverse top- k queries* [13], without theoretic analysis, but have recently received considerable attention from the database community.

Related Work. Fleischer and Xu [12] showed that for a large number of colors, the MDCSS problem is NP-hard even in two dimensions but is solvable in polynomial-time for a small number of colors. The fixed-parameter tractability of MDCSS is posed as an open problem in [12], in which they assume that the dimension d is fixed. Recently, Prunte [27] answered this question by proving that MDCSS is W[1]-hard by using a complicated reduction from multi-colored clique graph problems [11], where the dimension d is not fixed. Also, the author shows that the problem does not admit an FPTAS in arbitrarily high dimensional spaces. In the same paper, some algorithms with quadratic dependencies to n are also supporting the result.

Kazemi et al. presented a PTAS in high dimensional space for the MDCSS problem and proved that assuming the Exponential Time Hypothesis (ETH), there is no $(1 + \epsilon)$ -approximation algorithm with running time $2^{o(\epsilon^{(1-d)/2})} \text{poly}(n)$ to solve the MDCSS problem [17].

Instead of considering a discrete set for the possible locations of a color code, a continuous region of possible locations may determine a color code. Finding a point in each region such that the chosen set admits the smallest diameter is also extensively studied in this model. This formulation is introduced and extensively studied by Löffler and van Kreveld [22] for disks and squares,

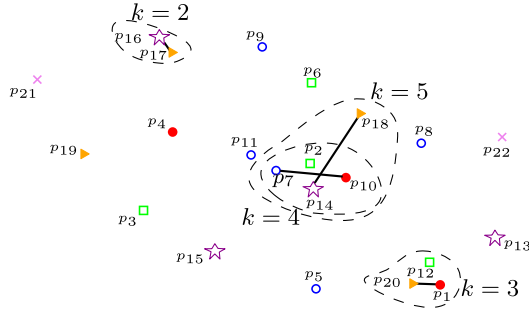


Fig. 2. Problem definition and optimal solutions, with k -rainbow sets for $k = 2, 3, 4, 5$.

and several improvements have been made recently to the complexity of their algorithms by Keikha et al. [19].

Regardless of whether the associated set of each color code is a continuous or a discrete set, the maximum diameter color spanning set problem is to locate a set of points, where the diameter has the largest possible size. This problem usually takes polynomial time as it is involved with the points in convex position. We refer the interested reader to [2, 7, 15, 29] for other related studies on MDCSS.

Our problem is closely related to outlier detection problems, except that their input is a set of monochromatic points: for a given $k < n$, exclude $n - k$ points (referred to as *outliers*) from P , such that the remaining points have the smallest possible diameter. In \mathbb{R}^2 , the best-known algorithm for this problem, developed by Eppstein et al., runs in $O(n \log n + k^2 n \log^2 k)$ time [9]. There also exists a lower bound $\Omega(n \log n)$ for this problem even for one outlier since the diameter picks the outlier as a vertex [3]. This implies that any fixed-parameter algorithm for computing a k -rainbow set is no better than $\Omega(n \log n)$ in \mathbb{R}^2 .

We finally note that to the best of our knowledge, no study has been conducted on our problem or the weighted version of the MDCSS problem.

Contribution. In the following, we formally define our problems: Let $P = \{p_1, \dots, p_n\}$ be a set of n points of m colors in \mathbb{R}^d , let t be the maximum frequency of any color in P , and let $1 < k < m$ be a positive integer.

Definition 1. Minimum Diameter k -Colored Spanning Set (MDkCSS). *The objective of the MDkCSS problem is to find a subset $P^* \subseteq P$ of size k of distinct colors, such that P^* has the smallest possible diameter among all possible choices. Formally $\text{diameter}(P^*) = \min_{\mathcal{P} \in \mathcal{D}(P)} \text{diameter}(\mathcal{P})$, where $\mathcal{D}(P)$ denote the collection of all k -subsets¹ of P of distinct colors.*

We call P^* a k -rainbow set of P ; see Fig. 2 for an illustration. The main application of this problem is in the case where the points have a predefined weight assigned, and the optimal k -rainbow set has the maximum total weight.

¹ i.e., subsets of size k .

Definition 2. Maximum Weight Minimum Diameter k -Colored Spanning Set (MWMDkCSS). We define a maximum weight k -rainbow set P^* as a k -subset of distinct colors that minimizes $\frac{\text{diameter}(P^*)}{\text{weight}(P^*)}$, where $\text{weight}(P^*)$ is the total sum of the weights of the points in P^* .

Results. In this paper, we first focus on the case where all the points have the same weight and then we discuss to what extent our results can be generalized to the weighted version under some restrictions. In particular:

- For the first time, we introduce a relation between the MDCSS problem and higher-order Voronoi diagrams. We first provide a fixed-parameter tractable (FPT) algorithm that has near-linear dependency on n in \mathbb{R}^2 (Theorem 1), which is helpful to improve the existing quadratic FPT algorithm (for small k and t) for the MDCSS problem [27].
- We show that MDkCSS is fixed-parameter tractable in \mathbb{R}^d for any fixed d (Sect. 3.1). We then show our FPT algorithm gives an approximation for the MWMDkCSS problem (Sect. 3.2).
- We have implemented our exact algorithm on a real data-set to consider the efficiency of our technique in practice, and we give several analyses on the studied data-set (Sect. 4).
- We then discuss the decision and the enumeration version of the MDkCSS problem for a given value q , and introduce an $O(n(tk)^2((tk)^{2.5} + \alpha))$ time algorithm, where α is the maximum number of the k -rainbow sets of size at most q . We hope these problems are of independent interest in data mining and database inquiries. To solve these problems, we introduce a reduction to all maximal independent sets of a bipartite graph (Sect. 5).
- We introduce a 2.236-approximation with running time $O(mn \log mn)$ for the enumeration version of MDkCSS, and a 1.154-approximation for the MDkCSS problem in \mathbb{R}^2 with running time $O(m^3n)$ (Sect. 6).

Our FPT algorithm is efficient when the parameters t and k are small, which is the common assumption of any FPT algorithm. Note that parametrizing a problem by the number of colors is common in computational geometry. We also remark that in the MDCSS problem if the number of the existing colors in P is a small k (possibly constant), we still do not have any exact algorithm with a running time better than $\binom{n}{k}$. In \mathbb{R}^2 , our FPT algorithm is near-linear to n .

2 Preliminaries

Maximum Independent Set (MIS). A maximum independent set of a graph $G = (V, E)$ is a subset $X \subseteq V$ with maximum size, in which there is no edge $e \in E$ between any $a, b \in X$. This problem is NP-hard, fixed-parameter intractable, and also hard to approximate [8]. The best algorithm for computing all maximum independent sets of a bipartite graph takes $O(s^{2.5} + \alpha)$ time [16], where s and α are the number of vertices and the total size of the output, respectively.

k -Order Voronoi Diagram. The *Voronoi diagram* of order k of P is the partitioning of the plane into a set of Voronoi cells, such that each Voronoi cell c is associated with a set $X \subseteq P$ of k points, and for each point p in the cell c , the k nearest neighbors of p are exactly the elements of X . We denote this diagram by V_k . Such diagrams can be computed in $O(k^2n + n \log n)$ time and have at most $O(nk)$ cells [20].

Fixed-Parameter Tractable (FPT). In fixed-parameter tractability, we provide some algorithms which no longer are exponential on the input size but on some other parameter related to the problem. These parameters are called the fixed parameter of the problem. Formally, for a given problem \mathcal{Y} , we characterize the input size, n , and some parameter k , and say \mathcal{Y} is fixed-parameter tractable if \mathcal{Y} can be solved by an algorithm that runs in $O(F(k) \cdot n^c)$ time, where F is a computable function depending on k , and c is any constant independent of k . Also, it is already known that parameterized complexity can be extended to achieve approximation algorithms for hard problems [24]. We use the same idea to achieve an FPT approximation algorithm for MWMDkCSS in Sect. 3.2.

Minimum Color Spanning Circle. For a set of n colored points of m colors, the smallest color spanning circle is a circle of the smallest radius that is covering m distinct colors [1]. In \mathbb{R}^2 , the smallest color spanning circle of m colors can be computed in $O(nm \log n)$ time by computing the upper envelope of some Voronoi surfaces [1, 14]. This problem becomes NP-hard in \mathbb{R}^d , where d is in the input, but admits a $(1 + \epsilon)$ -approximation in $O(dn^{\lceil 1/\epsilon \rceil + 1})$ time [18].

We first note that MDCSS problem is para-NP-hard² for the parameter t since the proof in [12] shows NP-hardness for t bounded by three. It can easily be extended to also show NP-hardness if at most 5 colored points are co-located (if we do a reduction by MAX-E3SAT(5) [10]). Hence, the problem may get easier if the number of colors is large, i.e., more than $\frac{n}{3}$.

3 MDkCSS is in FPT in Any Fixed Dimension

In the following, we assume that the points of P are in general position, that means no four points are co-circular. Recall that a k -rainbow set P^* is a set of points of k distinct colors, where P^* has the smallest possible diameter among all choices. In [12] it is posed as an open question which value of k is the threshold between easy and hard. We partially answer that question, as we do not need to cover all, but only k colors that their instances realize the smallest possible diameter. Our algorithm has a near-linear dependency on the number of points, where its hardness depends on k (and t , but we discussed above that t is not a parameter to determine the hardness). Consequently, we answer the posed

² A problem is para-NP-hard if it is NP-hard already for a constant value of the parameter.

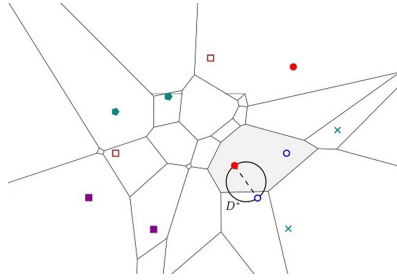


Fig. 3. Illustration of Lemma 1, the case where D^* has two points on its boundary. On a set P of colored points with $t = 2$, the optimal solution with $k = 2$ is associated with a cell c (shown in gray) of $V_{t(k-1)+1}$, and uses a pair of red and blue points (connected by a dashed line segment). Observe that D^* cannot contain more than 3 points of P , otherwise, there must be two points of different colors strictly within D^* , such that they realize a smaller diameter than the diameter of D^* . (Color figure online)

question in [12] partially as follows: for any constant number of colors which we need to span, the MDCSS problem in \mathbb{R}^2 can be answered in near-linear time.

In the following, we will show that any set of k colored points of smallest diameter is a subset of the points which are associated to a Voronoi cell of a Voronoi diagram (of P) of order $t(k - 1) + 1$, or $3t(k - 1) + 1$.

Lemma 1. *Let P be a set of n colored points, and let P^* be a k -rainbow set of P . Then P^* is a subset of the points which are corresponding to a Voronoi cell of a Voronoi diagram either of order $t(k - 1) + 1$, or $3t(k - 1) + 1$.*

Proof. Let $c(P)$ denote a subset of points of P that are associated with only one cell c of a Voronoi diagram $V_{t(k-1)+1}$, or $V_{3t(k-1)+1}$. Recall that for each Voronoi cell c , there exists a disk D having its center within c , where D contains no other point of $P - c(P)$. The set P^* also realizes a disk D^* such that either two or three points of P^* are located on its boundary.

Suppose by contradiction that the lemma is false, and P^* of k colors is not associated with one cell of a Voronoi diagram of order $t(k - 1) + 1$, or $3t(k - 1) + 1$.

By definition, in $V_{t(k-1)+1}$, the points of each cell of the diagram have the same $t(k - 1) + 1$ nearest neighbors. Observe that in the case where there are two points on the boundary of D^* , D^* cannot contain more than $t(k - 1) + 1$ points. If not, there always exist at least another set P' of k points from k distinct colors, which they all are entirely located within D^* , and the diameter of P' is strictly smaller than the diameter of D^* (i.e., P^*). This gives a contradiction. It follows that D^* cannot contain more than $t(k - 1) + 1$ points and P^* is contained in some Voronoi cell of a Voronoi diagram of order $t(k - 1) + 1$. See Fig. 3 for an illustration.

In the case where D^* has three points on its boundary, we partition D^* into three sectors by connecting the center of D^* to the points of P^* on its boundary.

Algorithm 1. Exact Algorithm

Input: $P = \{P_1, \dots, P_n\}$, and $k > 0$

Output: k -rainbow set of P

- 1: $d^* = \infty$
- 2: Compute $V_{3t(k-1)+1}$ (or $V_{t(k-1)+1}$ if $V_{3t(k-1)+1}$ does not exist)
- 3: **for** each cell c in $V_{3t(k-1)+1}$ **do**
- 4: $P_c =$ the associated points of P to c
- 5: **for** any k -subset P_c of c **do**
- 6: **if** there are k distinct colors in P_c **then**
- 7: $d_c =$ the diameter of P_c
- 8: **end if**
- 9: **if** $d^* > d_c$ **then**
- 10: $d^* = d_c, P_c^* = P_c$
- 11: **end if**
- 12: **end for**
- 13: **end for**
- 14: **return** d^* and P_c^*

Then each sector cannot contain more than $t(k-1)$ points since otherwise there would be k points from distinct colors in that sector so that the determined diameter by those points is strictly smaller than the diameter of P^* . Hence, D^* is contained in the associated points of a Voronoi cell of $V_{3t(k-1)+1}$. □

3.1 Algorithm

Observe that we only need to consider $V_{3t(k-1)+1}$ as the associated points in its cells strictly cover all possibilities in $V_{t(k-1)+1}$. From Lemma 1, the smallest diameter among each subset of k points of distinct colors that is associated to a Voronoi cell of $V_{3t(k-1)+1}$ determines an optimal solution. We design our algorithm based on this fact.

In the algorithm, we first make the Voronoi diagram of order $3t(k-1)+1$, of all the n points of P , without considering their colors in the construction. In each step of the algorithm, we consider the associated points of each cell of $V_{3t(k-1)+1}$ independently. Let d^* denote the diameter of the k -rainbow set P^* , and let P_c denote the associated points of a cell c . We use a brute force idea on P_c to find a subset $P_c^* \subseteq P_c$ of k distinct colors with the smallest possible diameter, and remember the P_c^* with the smallest d_c^* among all the cells of $V_{3t(k-1)+1}$.

In Lemma 1 we observed that each set P_c has a reasonable size with only a linear dependency to k and t , which means our algorithm has exponential dependence only in k and t . Since the complexity of the number of the cells of a Voronoi diagram of order tk is $O(ntk)$, our method gives an FPT algorithm with k and t as the parameters; see Algorithm 1.

Running Time Analysis. We will now elaborate on the complexity of the algorithm for a cell of $V_{3t(k-1)+1}$. To analyse the running time of considering

all k -subsets of $3tk - 3t + 1$ points, we use the Stirling's formula: $\binom{3tk-3t+1}{k} = \frac{2^{\log(3tk-3t+1)} - \log k! - \log(3tk-3t-k+1)!}{2^{\log(3tk-3t+1) - \log k! - \log(3tk-3t-k+1)!}}$. Then we have $\log(3tk - 3t + 1)! - \log k! - \log(3tk - 3t - k + 1)! = 3tk \ln(3tk - 3t + 1) - (3tk - 3t + 1) + O(\ln(3tk - 3t + 1)) - k \ln k + k - O(\ln k) - (3tk - 3t + 1) \ln(3tk - 3t - k + 1) + (3tk - 3t - k + 1) - O(\ln(3tk - 3t - k + 1)) \in O(tk)$.

Hence, considering all possible k -rainbow sets of one cell of $V_{3t(k-1)+1}$ takes $O(2^{O(tk)})$ time.

For each cell c of $V_{3t(k-1)+1}$, we can find a solution to the MDkCSS problem by finding a k -rainbow set with the smallest possible diameter among the corresponding points of c in $O(k \log k 2^{O(tk)})$ time, in which, in $O(k)$ time we determine whether the selected set contains k distinct colors, and $O(k \log k)$ time is required to find the diameter of this k -subset.

To generalize our FPT to a higher dimension d , we first need to construct the k -order Voronoi diagram in that dimension. Recall that a k -order Voronoi diagram in \mathbb{R}^d can be constructed in $O(n^{\lceil d/2 \rceil} k^{\lfloor d/2 \rfloor + 1})$ time [6].

Theorem 1. *Let P be a set of n colored points in \mathbb{R}^d . MDkCSS can be solved in $O(n(2^{O(tk)} + \log n) + n^{\lceil d/2 \rceil} k^{\lfloor d/2 \rfloor + 1})$ time.*

Proof. A Voronoi diagram of order $O(tk)$ can be computed in $O((tk)^2 n + n \log n)$ time and has at most $O(ntk)$ cells. A k -order Voronoi diagram in the dimension d can be constructed in $O(n^{\lceil d/2 \rceil} k^{\lfloor d/2 \rfloor + 1})$ time, so by repeating the algorithm of Sect. 3.1 for all the cells of the Voronoi diagram $V_{3t(k-1)+1}$ in \mathbb{R}^d , the algorithm takes $O(n2^{O(tk)} + n \log n)$ time. Hence, the problem can be solved in $O(n(2^{O(tk)} + \log n) + n^{\lceil d/2 \rceil} k^{\lfloor d/2 \rfloor + 1})$ time. □

Corollary 1. *MDkCSS is in FPT in \mathbb{R}^d for any fixed d , with k and t as the parameters.*

3.2 Maximum Weight k -rainbow Set

For any point $p_i \in P$, let w_i denote the weight of p_i . W.l.o.g, we assume $w_i > 0$, $i = 1, \dots, n$. It is easy to observe that the problem at which a k -rainbow set P^* (for general values of k) minimizes $\frac{\text{diameter}(P^*)}{\text{weight}(P^*)}$, where $\text{weight}(P^*) = \sum_{p_i \in P^*} w_i$ is NP-hard with the same reduction in [12] for the MDCSS, by using an extra assumption of assigning the same weight to all the points in P . We discuss that Algorithm 1 is applicable on particular cases of this problem, at which the ratio of the weights of any two points in P is at most ω . This is a reasonable assumption since in any environment, the input data are usually relevant and are not that much different in the sense of measurement precisions. Also, we can measure the ratio of the weights in polynomial time. Then Algorithm 1 gives a ω -approximation for the MWMDkCSS problem, as in the worst case the two points of large weight that are far apart from each other, and have to be in an optimal solution, will not land in the same cell. So we may not consider solutions containing both these points. But the ratio of the weight of a point in the reported and the optimal solution is at most within a factor ω . Assuming all the elements

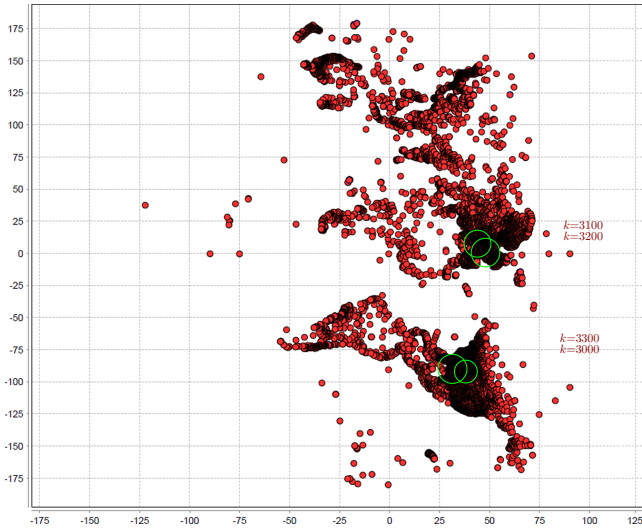


Fig. 4. Illustration of the output of Algorithm 1 on the data-set [5, 21], and the ranges at which the optimal solutions for $k = 3000, 3100, 3200, 3300$ appear.

of P^* lands at different cells and summing up the weights of such points gives the approximation factor at most ω . Note that an arbitrary k -rainbow set has the same approximation ratio only for the total sum of the weights of the points. But such a set cannot guarantee to have the minimum possible diameter among all choices for approximating $\frac{\text{diameter}(P^*)}{\text{weight}(P^*)}$ within a factor ω .

Theorem 2. *If the ratio of the weights of any two points in P is at most ω , Algorithm 1 gives an FPT ω -approximation for MWMDkCSS in \mathbb{R}^d for any fixed d , with k and t as the parameters.*

4 Experimental Studies

We discussed an application for our problem in the Introduction. In this section, we discuss another application along with our experimental tests to evaluate the performance of Algorithm 1 in practice. We do our computational tests on a real data-set in \mathbb{R}^2 .

Our data-set characterizes the locations and times of check-ins of the Brightkite social network, which has a reasonable size and several users with different check-ins each, so that we can model each user as a color code. This network has 58,228 users and 4,491,143 check-ins of these users ranging in the period of April 2008 to October 2010. This data-set is contained in the SNAP network [5, 21].

We assign a color to each user, and of course we need the users with at least one check-in (to denote the frequency at least one for each color). This number equals 51,685 in this data-set. The total number of colors (m) is also

Table 1. Experimental results of Algorithm 1 on the Brightkite data-set [5, 21].

Input size	k	# cells of $V_{t(k-1)+1}$	Alg 1 Time (s)	BF Time (s)
4,491,143	3000	10272	2.3867	0.1191e+2
	3100	13070	2.8967	0.1259e+2
	3200	14221	3.9683	0.1436e+2
	3300	19744	4.1662	0.1696e+2

51,685, and n equals 4,491,143, that is the total number of the check-ins. Each user had at most 325,821 check-ins which means $t = 325,821$. For a given k , our objective is to find k customers whose target check-ins are as close as possible. One may use this information to locate a facility for at least k customers in the neighbourhood of their check-in places. Our experiments have shown that for $k \leq 2876$, the solution to the MDkCSS was zero in this data-set, which means this number of customers have at least one common check-in station. In our experiments, we set $k = 3000, 3100, 3200, 3300$; see Fig. 4.

We have implemented our algorithm in C++ with Visual Studio 2013. The algorithm is performed on a Core (TM) i9CPU and 8 GB RAM computer with Windows 10 operating system. In some of the computations of the Voronoi diagrams, we have used CGAL-5.1. The reported running time in Table 1 is the elapsed time of searching for a solution on $V_{t(k-1)+1}$, since the condition $3t(k - 1) + 1 \leq n$ did not hold, and there is no solution on $V_{3t(k-1)+1}$.

In each test, we have verified the output of our algorithm with the brute force algorithm which is trying all k -subsets, as this problem is not considered so far, and the brute force is the only existing current algorithm. We have reported the running times in Table 1. The last column contains the running time of the brute force algorithm which is comparable to the running time of our algorithm in the previous column.

We observe that our algorithm has a reasonable performance in the reported experimental studies in this paper. In our experiments, computing a Voronoi diagram of a high order was the time-consuming part, and was taking at least 67.46% of the reported elapsed times. Based on this, we conclude the other computations were relatively quick; that is because the dependency of the algorithm to the number of points is near-linear in \mathbb{R}^2 . The results of the implementation are summarized in Table 1.

5 Enumerating All MDkCSS of Diameter at Most q

In this section, we study the following problems: given a set P of colored points and a positive value q , determine whether there is any k -rainbow set in P of diameter at most q , and report all k -rainbow sets of P of diameter at most q .

Let c be any cell of $V_{3t(k-1)+1}$ that has at most $O(tk)$ points, and let $X \subseteq P$ denote the associated points of P to c . For any pair $p_i, p_j \in X$ of distinct colors, let $z = d(p_i, p_j)$ denote their Euclidean distance. Our first objective is to

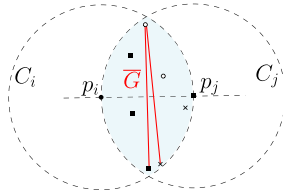


Fig. 5. Illustration of the graph \overline{G} in Lemma 2.

determine whether there is any set of k points of distinct colors in X , where the pairwise distances between the points are at most $z \leq q$. Consider two circles C_i and C_j of radius z , one is centered at p_i and the other at p_j . Let X' denote the set of points in $C_i \cap C_j \cap P$. Construct a graph G on X' by connecting any pair of points with a distance at most z , and let \overline{G} denote the complement of G . Observe that the vertices of X' which are lying on exactly one side of the line through $p_i p_j$ are at a distance less than z . Consequently, in \overline{G} , the connected pair of vertices lie on opposite sides of the line through $p_i p_j$.

Lemma 2. \overline{G} is a bipartite graph.

Proof. The vertices of the graph which lie on only one side of the line through $p_i p_j$ have a smaller distance of z . Consequently, the vertices which are already connected in \overline{G} , have a further distance than z , and lie at different sides of the line through $p_i p_j$; see Fig. 5. Hence, the vertices at each side of the line through $p_i p_j$ in \overline{G} determine a part in a bipartite graph. \square

Observe that the points forming a clique in G are points such that each pair of points has a distance smaller than z , and thus, such points form an independent set in \overline{G} .

Lemma 3. Any k -rainbow set of diameter at most z is a subset of at least one maximum independent set in \overline{G} .

Proof. Suppose, by contradiction, there is a k -rainbow set S of diameter at most z that is not a subset of any maximum independent set in \overline{G} . Every independent set (including the ones in \overline{G}) has this property that there is no edge between any pair of the vertices. If S is not a subset of any of the independent sets (including the maximum ones) in \overline{G} , there must be an edge between at least one pair of vertices in S . But this means the distance between those vertices is strictly larger than z ; contradiction. \square

Hence, maximum independent set enumeration algorithm can be used for our problem but only reports the ones having our cardinality and color constraint. We check whether there is any maximum independent set X^* of size at least k in \overline{G} , where at least k vertices in X^* has distinct colors. To treat this, for each possible maximum independent set we check whether there is any set of k distinct colors among the reported vertices or not. Using the presented algorithm

Algorithm 2. Enumeration Algorithm

Input: $P = \{P_1, \dots, P_n\}$, and $k, q > 0$
Output: all k -rainbow set of P of diameter at most q

- 1: Compute $V_{3t(k-1)+1}$ (or $V_{t(k-1)+1}$ if $V_{3t(k-1)+1}$ does not exist)
- 2: **for** each cell c in $V_{3t(k-1)+1}$ **do**
- 3: X = the associated points of P to c
- 4: **for** each pair $p_i, p_j \in X$ **do**
- 5: **if** they have distinct colors and $z = d(p_i, p_j) \leq q$ **then**
- 6: $G = (X', E) = \emptyset$
- 7: Construct C_i, C_j
- 8: $X' = C_i \cap C_j \cap P$
- 9: **end if**
- 10: **for** any pair $x, y \in X'$ **do**
- 11: **if** $d(x, y) \leq z$ **then**
- 12: add xy to E
- 13: **end if**
- 14: **end for**
- 15: \bar{G} = the complement of G
- 16: report any MIS of \bar{G} which has k distinct colors
- 17: **end for**
- 18: **end for**

in [16] and considering the freedom of p_i and p_j in $O(n tk)$ cells of $V_{3t(k-1)+1}$, the enumeration algorithm takes $O(n tk) \times O(tk)^2 \times O((tk)^{2.5} + \alpha)$ time, where α is the maximum number of the k -subsets of diameter at most q . This procedure is outlined in Algorithm 2. We note that one may use this algorithm combined with a binary search to compute the optimal q among $O(n^2)$ possible candidates. But the asymptotic running time would be worse than what we discussed in Sect. 3.1.

Theorem 3. *The decision or the enumeration version of MDkCSS can be solved in $O(n(tk)^2((tk)^{2.5} + \alpha))$ time, where α is the maximum number of k -subsets of diameter at most q .*

6 Approximation Algorithms

In this section, we discuss several approximation algorithms, mostly by geometric reductions to other problems. We first reduce MDkCSS to a well-known problem in *trajectory analysis*, the *discrete popular places* problem. Given is a set Π of polygonal paths with a total of n vertices, that is modelling a set of moving points (so-called *entity*) belonging to m distinct entities in the plane, an integer threshold $k > 0$ and a real value $r > 0$. A *popular place* is a square of side length r , that is visited by at least k distinct entities [4]. This problem can be solved in $O(mn \log mn)$ time and $O(mn)$ space [4]. In our setting, we assign the points of the same color to a single entity. The path between them is arbitrary. Hence, a popular place with a maximum number of entities gives a $\sqrt{2}$ -approximation for the MDCSS problem. Also, any algorithm for squares assuming a threshold k as

a popular place gives also an approximation for the MDkCSS problem. Reporting all the popular places for rectangles of threshold k takes $O(mn \log mn)$ time and $O(mn)$ space [4]. Reporting all popular places with at least k entities, where the popular places modelled by a rectangle of size 1×2 , reports all k -rainbow sets of diameter at most γq for a given $q > 0$, where $\gamma = \sqrt{5} \approx 2.236$.

Theorem 4. *For a given $q > 0$, all the k -rainbow sets of diameter of size at most $\sqrt{5}q$ can be listed in $O(mn \log mn)$ time and $O(mn)$ space.*

A 1.154-Approximation for MDkCSS. We discuss another simple efficient approximation algorithm. We start stating our result with the following lemma.

Lemma 4. *For any set X of points, the diameter of X is longer than $\sqrt{3}$ times the radius of the smallest enclosing circle (SEC) of X .*

Proof. Consider the configuration at which three points on the boundary of the SEC form an equilateral triangle, and the side of the triangle determines the diameter. If one translates any pair of these points on the boundary of the SEC, to get closer, the size of the diameter would be increased between at least one pair. The lemma follows. □

Let r_X and d_X denote the radius of the SEC and the diameter of X , respectively. For a set P of points, let X be the set realizing the smallest color spanning circle with k colors, and let P^* denote the set of points realizing the k -rainbow set of smallest diameter. Using the fact that the radius r_X is smaller than the radius of the color spanning circle of P^* , we have $d_{P^*} \leq d_X \leq 2r_{P^*} \leq 2/3\sqrt{3}(\sqrt{3}r_{P^*}) \leq 2/3\sqrt{3}d_{P^*}$. So, the diameter of X approximates the optimal k -rainbow set within a factor $2/3\sqrt{3} \approx 1.154$. An obvious $O(m^3n)$ time algorithm for computing the smallest color spanning circle of at least k colors considers any pair or triple of points of distinct colors that define a circle. We then have the following result.

Theorem 5. *Let P be a set of n colored points of m colors. In \mathbb{R}^2 , a 1.154-approximation for the MDkCSS problem can be computed in $O(m^3n)$ time.*

7 Discussions and Open Problems

In this paper, we introduced an easy proof that MDCSS problem is in FPT in \mathbb{R}^d for any fixed d , and we discussed several new variants of this problem, FPT, exact and approximation algorithms along a practical application.

One open question concerns designing efficient algorithms for the general case of the weighted points, and also for the enumeration problem on particular sets of points, in which the bipartite graph G has a bounded tree-width and admits a polynomial time algorithm for computing all MIS's. The tree-width $O(kt)$ for G is obvious. Another direction is to find the attributes on the point

sets in which the *maximum colorful independent set* (i.e., an independent set of maximum number of colors) on the bipartite graph G admits a polynomial time algorithm. This problem was recently shown to be NP-hard, but admits polynomial time algorithms on trees and cluster graphs [23]. Another open question is the existence of the FPT algorithms for other parameters of a point set in the MDCSS problem, such as assuming a specific extent measure for the points of any color code.

The possibility of improving the running time of our algorithms also remained open. One possible improvement to our results concerns approximating the MDkCSS in fixed dimensions using LP-formulation. According to Theorem 1.2 in [25], computing a circle of smallest radius that intersects n points can be reformulated to satisfy only k of n constraints, in $O(nk^d)$ time, where d equals the geometric dimension of the original problem, and this would be performed by finding the optimal solution to $O(k^d)$ independent LP-type problems. When we are generating an independent LP-type problem from the original problem, we can rewrite the constraint that counts the number of points to count the number of points of distinct colors; let $x_i = 1$ if the color c_i appears in the solution space, and $x_i = 0$, otherwise. Then we need to satisfy the constraint $\sum_{i=1}^m x_i = k$ in any of the independent solution sub-spaces. Thus, we can apply the existing algorithms for computing the smallest color spanning balls in \mathbb{R}^2 [1, 14] and in \mathbb{R}^d [18], that intersect k colors in each of the solution spaces of the independent LP-type problems. This may slightly improve the approximation ratio and the running time we discussed in Theorem 5.

Acknowledgment. V. Keikha was supported by the Czech Science Foundation, grant number GJ19-06792Y, and with institutional support RVO: 67985807.

References

1. Abellanas, M., et al.: Smallest color-spanning objects. In: auf der Heide, F.M. (ed.) ESA 2001. LNCS, vol. 2161, pp. 278–289. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44676-1_23
2. Agarwal, P.K., Har-Peled, S., Varadarajan, K.R.: Approximating extent measures of points. *J. ACM* **51**(4), 606–635 (2004)
3. Atanassov, R., et al.: Algorithms for optimal outlier removal. *J. Discrete Algorithms* **7**(2), 239–248 (2009)
4. Benkert, M., Djordjevic, B., Gudmundsson, J., Wollé, T.: Finding popular places. *Int. J. Comput. Geom. Appl.* **20**(01), 19–42 (2010)
5. Cho, E., Myers, S.A., Leskovec, J.: Friendship and mobility: user movement in location-based social networks. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1082–1090 (2011)
6. Clarkson, K.L., Shor, P.W.: Applications of random sampling in computational geometry, II. *Discrete Comput. Geom.* **4**(5), 387–421 (1989). <https://doi.org/10.1007/BF02187740>
7. Consuegra, M.E., Narasimhan, G.: Geometric avatar problems. In: Leibniz International Proceedings in Informatics. LIPIcs, vol. 24. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2013)

8. Cook, S.A.: The complexity of theorem-proving procedures. In: Proceedings of the 3th Annual ACM Symposium on Theory of Computing, pp. 151–158 (1971)
9. Eppstein, D., Erickson, J.: Iterated nearest neighbors and finding minimal polytopes. *Discrete Comput. Geom.* **11**(3), 321–350 (1994). <https://doi.org/10.1007/BF02574012>
10. Feige, U.: A threshold of $\ln n$ for approximating set cover. *J. ACM* **45**(4), 634–652 (1998)
11. Fellows, M.R., Hermelin, D., Rosamond, F.A., Vialette, S.: On the parameterized complexity of multiple-interval graph problems. *Theoret. Comput. Sci.* **410**(1), 53–61 (2009)
12. Fleischer, R., Xu, X.: Computing minimum diameter color-spanning sets is hard. *Inf. Process. Lett.* **111**(21), 1054–1056 (2011)
13. He, Z., Lo, E.: Answering why-not questions on top- k queries. *IEEE Trans. Knowl. Data Eng.* **26**(6), 1300–1315 (2012)
14. Huttenlocher, D.P., Kedem, K., Sharir, M.: The upper envelope of Voronoi surfaces and its applications. *Discrete Comput. Geom.* **9**(3), 267–291 (1993)
15. Ju, W., Fan, C., Luo, J., Zhu, B., Daescu, O.: On some geometric problems of color-spanning sets. *J. Combinat. Optim.* **26**(2), 266–283 (2013)
16. Kashiwabara, T., Masuda, S., Nakajima, K., Fujisawa, T.: Generation of maximum independent sets of a bipartite graph and maximum cliques of a circular-arc graph. *J. Algorithms* **13**(1), 161–174 (1992)
17. Kazemi, M.R., Mohades, A., Khanteimouri, P.: Approximation algorithms for color spanning diameter. *Inf. Process. Lett.* **135**, 53–56 (2018)
18. Kazemi, M.R., Mohades, A., Khanteimouri, P.: On approximability of minimum color-spanning ball in high dimensions. *Discret. Appl. Math.* **279**, 188–191 (2020)
19. Keikha, V., Löffler, M., Mohades, A.: A fully polynomial time approximation scheme for the smallest diameter of imprecise points. *Theoret. Comput. Sci.* **814**, 259–270 (2020)
20. Lee, D.-T.: On k -nearest neighbor Voronoi diagrams in the plane. *IEEE Trans. Comput.* **100**(6), 478–487 (1982)
21. Leskovec, J., Krevl, A.: Snap datasets: Stanford large network dataset collection (2014)
22. Löffler, M., van Kreveld, M.: Largest bounding box, smallest diameter, and related problems on imprecise points. *Comput. Geom.* **43**(4), 419–433 (2010)
23. Manoussakis, Y., Pham, H.P.: Maximum colorful independent sets in vertex-colored graphs. *Electron. Notes Discrete Math.* **68**, 251–256 (2018)
24. Marx, D.: Parameterized complexity and approximation algorithms. *Comput. J.* **51**(1), 60–78 (2008)
25. Matoušek, J.: On geometric optimization with few violated constraints. *Discrete Comput. Geom.* **14**(4), 365–384 (1995). <https://doi.org/10.1007/BF02570713>
26. Preparata, F.P., Shamos, M.I.: Convex hulls: basic algorithms. In: *Computational Geometry. Texts and Monographs in Computer Science*, pp. 95–149. Springer, New York (1985). https://doi.org/10.1007/978-1-4612-1098-6_3
27. Prunte, J.: Minimum diameter color-spanning sets revisited. *Discrete Optim.* **34**, 100550 (2019)
28. Toussaint, G.T.: Solving geometric problems with the rotating calipers. In: *Proceedings of the IEEE MELECON*, vol. 83, p. A10 (1983)
29. Zhang, D., Chee, Y.M., Mondal, A., Tung, A.K., Kitsuregawa, M.: Keyword search in spatial databases: towards searching by document. In: *Proceedings of the 25th IEEE International Conference on Data Engineering*, pp. 688–699 (2009)