Constrained Hitting Set Problem with Intervals

Ankush Acharyya[§⋆], Vahideh Keikha[§⋆], Diptapriyo Majumdar[†], and Supantha Pandit[‡]

[§] The Czech Academy of Sciences, Institute of Computer Science, Czech Republic
[†] Royal Holloway, University of London, Egham, United Kingdom
[‡] Dhirubhai Ambani Institute of Information and Communication Technology, India
acharyya@cs.cas.cz, keikha@cs.cas.cz, diptapriyo.majumdar@rhul.ac.uk, pantha.pandit@gmail.com

Abstract. We study a constrained version of the *Geometric Hitting Set* problem where we are given a set of points, partitioned into disjoint subsets, and a set of intervals. The objective is to hit all the intervals with a minimum number of points such that if we select a point from a subset then we must select all the points from that subset. In general, when the intervals are disjoint, we prove that the problem is in FPT, when parameterized by the size of the solution. We also complement this result by giving a lower bound in the size of the kernel for disjoint intervals, and we also provide a polynomial kernel when the size of all subsets is bounded by a constant.
Next, we consider two special cases of the problem where each subset can have at most 2 and 3 points. If each subset contains at most 2 points and the intervals are disjoint, we show that the problem admits a polynomial-time algorithm. However, when each subset contains at most 3 points and intervals are disjoint, we prove that the problem is NP-Hard and we provide two constant factor approximations for the problem.

1 Introduction

The *Hitting Set* problem is a well-studied problem in theoretical computer science, especially in combinatorics, computational geometry, operation research, complexity theory, etc. In the classical setup of the *Hitting Set* problem, a universe of elements U and a collection $\mathcal{F} \subseteq 2^U$ are given. The goal is to find the smallest subset $S \subseteq U$ that intersects every set in $\mathcal{F}$. The decision version of the *Hitting Set* problem is known to be NP-Complete, whereas the optimization version of the problem is NP-Hard [16]. Significant attention is also given to the geometric version of the *Hitting Set* problem due to its practical importance. In this version, U is considered to be a set of points and $\mathcal{F}$ is a set of geometric objects (such as intervals, disks, boxes, etc.). Due to the underlying geometric structure of these objects, different *Geometric Hitting Set* problems are shown to be polynomial-time solvable, however many problems remain NP-Hard [15,16].

⋆ The author is supported by the Czech Science Foundation, grant number GJ19-06792Y, and by institutional support RVO: 67985807

We study a constrained variation of the *Geometric Hitting Set* problem, the *Constrained Hitting Set* problem with intervals, defined as follows:

Constrained Hitting Set with Intervals (*CHSI*): We are given a set of closed intervals, $\mathcal{I}$ and a set P of n points in $\mathbb{R}$ partitioned into d nonempty subsets $P_1, P_2, \ldots, P_d$, such that $\bigcup_{i=1}^{d} P_i = P$ and $P_i \cap P_j = \emptyset$ for all $i \neq j$, $i, j \in \{1, 2, \ldots, d\}$. The objective is to find a subset $P' \subseteq P$ of minimum number of points such that each interval in $\mathcal{I}$ is hit[a] and for each $p \in P'$, if $p \in P_i$ for some $i \in \{1, \ldots, d\}$ then $P_i \subseteq P'$.

[a] An interval I is said to be *hit* by a point p if and only if $p \in I$.

To be precise, we are interested in the following variations of the *CHSI* problem based on the size (number of points) of the subsets and the underlying structure of the intervals. We define the *CHSI-tD* (resp. *CHSI-tO*) problem as the *CHSI* problem with intervals where each subset P_i is of size at most t and the given intervals are disjoint (resp. overlapping). Note that for $t = 1$, the *CHSI-tO* problem is the standard *Hitting Set* problem with intervals on the real line and can be solved in $O(n \log n)$ time [20]. When the size of the subsets is not bounded by any fixed number, then we call this variant as *CHSI-D* problem. We also consider a variation where we minimize the number of subsets of points, instead of the total number of points. We call such a variation as *Weak Constrained Hitting Set* with intervals (*WCHSI-D* problem).

We denote the decision version of the *CHSI-D* problem as the *DCHSI-D* problem where one additional parameter k is given as part of the input with usual input of the *CHSI-D* problem and the objective is to decide whether there is a set of at most k points that satisfy the constraints and hit all the intervals. The total number of points in the solution is at most k. Similarly, we denote the decision versions of the variations *CHSI-tD*, *CHSI-tO* as *DCHSI-tD*, *DCHSI-tO* problems. Further, we denote the decision version of the *WCHSI-D* problem as the *DWCHSI-D* problem.

A possible application of the *CHSI* problem is to provide efficient project management system. To satisfy the requirement of a project with a set of skills like developing, programming, visualizing, etc., the workload needs to be divided among the employees with proficiency in programming, data analysis, design, etc. The requirements of the project can be modeled as intervals and the expertise of employees as the set of points. To manage all the requirements of the project, we need all the employees to have the required expertise. Then the objective is to assign each of the projects to a set of employees satisfying the project requirements, and identify a number of smallest possible resources to complete it. Another possible application of a special case of the *CHSI* problem where the intervals are disjoint is as follows. Suppose that there is a number of working sites where a number of workers work. These sites need to be supervised by a collection of supervisors during the working hours of a day. The total working hour is divided into small chunks of time windows. A supervisor needs to visit many sites as assigned to him/her during the start of a day. Now for a particular site a number of supervisors visit in different time windows. During each time window a supervisor needs to be present. This problem can be modelled as the *CHSI-D* problem, where time windows are represented as intervals, visiting a particular site in a time window by a supervisor represents a point in that time window, and visiting the site by a supervisor in different time windows represents a subset of points hitting a collection of intervals (corresponds to the time windows). Now minimizing the number of supervisors visiting a particular site is same as minimizing the number of subsets that hit all the disjoint intervals.

1.1 Related work

A rich body of work has been done for the classical version of the *Hitting Set* problem that is equivalent to the classical *Set Cover* problem [5]. There is a well-known greedy algorithm for the *Hitting Set* problem that gives an $O(\log n)$-factor approximation [16,17] and we can not get an $o(\log n)$-factor approximation unless P=NP [13]. However, exploiting the underlying geometry, the*Hitting Set* problem on some geometric objects can be solved in polynomial-time or some NP-Hard problems have better approximation factors [4,6,7,22]. More specifically, both *Set Cover* and *Hitting Set* problems with intervals on the real line can be solved in polynomial time using greedy algorithms [20]. In one dimension, the *Geometric Hitting Set* (also *Set Cover*) problem on different objects remains NP-Complete [3], however, for a restricted class of objects they can be solved efficiently [16]. The *Constrained Hitting Set* problem was introduced by Cornet and Lafornet [9] on general graphs. They provided various computational hardness status and approximation algorithms for different problems, such as *Vertex Cover*, *Connected Vertex Cover*, *Dominating Set*, *Total Dominating Set*, *Independent Dominating Set*, *Spanning Tree*, *Connected Minimum Weighted Spanning Graph*, *Matching*, and *Hamiltonian Path* problems. These vertex deletion problems on graphs with obligation can be interpreted as variants of the *Constrained Implicit Hitting Set* problems on graphs. On the contrary, the *conflict-free* versions of *Implicit Hitting Set* problems on graphs have also been studied [2,8,28,19]. In the conflict-free version, a different conflict graph with the same input vertex set is provided as part of the input. The goal is to find a set of size at most k that forms a corresponding implicit hitting set in the original input graph, but an independent set in the conflict graph.

From the perspective of parameterized complexity, *Hitting Set* and the *Set Cover* problems are W[2]-hard [11] parameterized by solution size. However, when all sets in the input have at most d (for some constant d), elements, the d-*Hitting Set* problem admits a polynomial kernel [1] parameterized by solution size. Computing a kernel of smaller size is also studied in [24]. Also, polynomial-sized kernels for hitting set for a fixed d have already been presented in [14]. On the other hand, if the *Set Cover* problem is parameterized by the number of elements in the universe, then it is FPT and does not admit a polynomial kernel unless $\mathsf{NP} \subseteq \mathsf{coNP/poly}$ [10]. Jacob et al. [18] studied the conflict-free version of the Set Cover problem with parameterized complexity and kernelization perspective. Related problems of [18] are also studied in [26]. See also [25] and the references therein.

1.2 Our contribution

➢ We show that the *DCHSI-D* problem admits an algorithm taking $O^*(2^k)$-time, where k is the total number of points in the solution. We also prove that the *DCHSI-tD* problem admits a kernel with k intervals and $O(t^2k^t)$ points.
➢ We prove that the *DWCHSI-D* problem parameterized by the number of intervals does not admit a polynomial kernel unless $\mathsf{NP} \subseteq \mathsf{coNP/poly}$. We also give an algorithmic lower bound of the *DWCHSI-D* problem based on the Set Cover Conjecture.
➢ The *CHSI-2D* problem admits a polynomial-time algorithm.
➢ The *CHSI-3D* problem is NP-Hard. We present two constant factor approximations for this problem.

Due to lack of space, some proofs are omitted; they will be provided in the full version of the paper.

2 Parameterized complexity for disjoint intervals

2.1 Preliminaries

A parameterized problem is $\Pi \subseteq \Sigma^* \times \mathbb{N}$ for some finite alphabet Σ. An instance of a parameterized problem is $(x, k) \in \Sigma^* \times \mathbb{N}$ where k is called the parameter and x is the input. We assume that k is given in unary and without loss of generality $k \leq |x|$, and $|x|$ is the input length. A parameterized problem $\Pi \subseteq \Sigma \times \mathbb{N}$ is said to be *fixed-parameter tractable* (or *FPT*) if there exists an algorithm that runs in $f(k)|x|^c$ time where $f : \mathbb{N} \to \mathbb{N}$ is a computable function and c is a constant.

Kernelization in parameterized complexity is a polynomial-time preprocessing algorithm. Formally, given an instance (x, k) of a parameterized problem $\Pi \subseteq \Sigma^* \times \mathbb{N}$, *kernelization* is a polynomial-time algorithm that transforms the input instance (x, k) to (x', k') such that **(i)** $(x, k) \in \Pi$ if and only if $(x', k') \in \Pi$, and **(ii)** $|x'| + k' \leq f(k)$ for some function $f : \mathbb{N} \to \mathbb{N}$ depending only on k. If $f(k)$ is $k^{O(1)}$, then we say that Π has a *polynomial kernel*. Informally speaking, kernelization is a collection of *reduction rules* that have to be applied in sequence to reduce the original instance into an equivalent instance. A reduction rule that replaces input instance (x, k) by (x', k') is *safe* if (x, k) is a yes-instance if and only if (x', k') is a yes-instance. It is well-known that, a parameterized problem is FPT if and only if it admits a kernelization [11]. Another type of polynomial-time preprocessing in parameterized complexity is called a "compression". Formally, given an instance (x, k) of parameterized problem $\Pi \subseteq \Sigma^* \times \mathbb{N}$, *compression* transforms (x, k) to an equivalent instance $x' \in \Sigma^*$ of an unparameterized problem $L \subseteq \Sigma^*$ in polynomial-time such that x' can be represented by $f(k)$ bits. If $f(k)$ is in $k^{O(1)}$, then we say that Π admits a *polynomial compression*. Informally speaking, polynomial compression is a polynomial-time preprocessing algorithm that transforms the input instance of a parameterized problem to an input instance of a possibly different unparameterized problem with a polynomial number of bits.

Let Π_1 and Π_2 be two parameterized problems. If there exists a polynomial-time reduction that given an instance (x, k) of Π_1, constructs an instance (x', k') such that $k' = O(k^{O(1)})$, then we say that there exists a *polynomial parameter transformation (PPT)* from Π_1 to Π_2.

2.2 Fixed-parameter tractability for disjoint intervals

We show that the *DCHSI-D* problem with disjoint intervals is fixed parameter tractable parameterized by the size of the solution. *DCHSI-D* is NP-Hard when there are subsets of points that are of size at least 3 (see Section 4). Our kernel lower bound results also prove that the *DWCHSI-D* problem is NP-Hard.

We apply the following reduction rules in sequence.

Reduction Rule 1 If the number of intervals is more than k, then the given instance is a "NO" instance.

Reduction Rule 2 If there are two subsets P_i, P_j in P such that $|P_i| = |P_j|$ and both of them hit the same set of intervals, then we remove P_i from the input.

Reduction Rule 3 If there exists a subset P_i that does not hit any interval, i.e. $P_i \cap \mathcal{I} = \emptyset$, then we simply remove that subset P_i from the input.

Reduction Rule 4 If any subset P_i contains more than k points, we can remove P_i from the input. Such a subset only makes the size of the solution more than k. Thus we have the following lemma.

Lemma 1. *Reduction Rules 1, 2, 3, and 4 are safe, and can be implemented in polynomial-time. Thus the DCHSI-D problem admits a kernel of size* $O(2^k k)$ *and an FPT algorithm with* $O^*(2^{k^2})$ *running time.*

Dynamic Programming: Now, we describe an improved $O^*(2^k)$ time algorithm by using *dynamic programming* over subsets of intervals ($\mathcal{I}$) where the set of points are $P = P_1 \cup P_2 \cup \ldots \cup P_d$, and $P_i \cap P_j = \emptyset$ for all $i \neq j$. For every P_i, we denote $w(P_i) = |P_i|$ (number of points in P_i). Since Reduction Rule 1 is not applicable, $|\mathcal{I}| \leq k$. We fix an arbitrary ordering $P_1, P_2, \ldots, P_d$. For every $i \in [d]$, we use $c(P_i)$ to denote the set of intervals hit by the points in P_i.

For every subsets of intervals $\mathcal{X} \subseteq \mathcal{I}$, for every $i \in \{1, \ldots, d\}$, we define $\mathsf{B}[\mathcal{X}, i]$, the weight of a smallest subset $\mathcal{P} \subseteq \{P_1, \ldots, P_i\}$ such that $\mathcal{X} \subseteq \bigcup_{P \in \mathcal{P}} c(P)$. Informally speaking, in the table entry $\mathsf{B}[\mathcal{X}, i]$, we store the weight of a smallest subset $\mathcal{P} \subseteq \{P_1, \ldots, P_i\}$ that hits all intervals in $\mathcal{X}$. Since no point is required to hit $\emptyset \subseteq \mathcal{I}$, for all $i \in \{1, 2, \ldots, d\}$, we initialize $\mathsf{B}[\emptyset, i] = 0$. For $\mathcal{X} \neq \emptyset$, if $\mathcal{X} \subseteq c(P_1)$, then $\mathsf{B}[\mathcal{X}, 1] = |P_1| = w(P_1)$. Otherwise, when $\mathcal{X} \not\subseteq c(P_1)$, we denote $\mathsf{B}[\mathcal{X}, 1] = \infty$. For all other $\mathcal{X} \subseteq \mathcal{I}$, and for all $i \in [d]$, we initialize $\mathsf{B}[\mathcal{X}, i] = \infty$. We use the following recurrence relation. For every $\mathcal{X} \subseteq \mathcal{I}$ such that $\mathcal{X} \neq \emptyset$, and for every $i \geq 2$, we denote $\mathsf{B}[\mathcal{X}, i] = \min\{|P_i| + \mathsf{B}[\mathcal{X} \setminus c(P_i), i-1], \mathsf{B}[\mathcal{X}, i-1]]\}$. Thus we have the following.

Theorem 1. *The DCHSI-D problem can be solved in* $O^*(2^k)$ *time.*

2.3 Kernelization and FPT lower bound for disjoint intervals:

We prove that the *DWCHSI-D* problem admits no polynomial compression unless $\mathsf{NP} \subseteq \mathsf{coNP/poly}$. In this variant, we aim to minimize the number of distinct subsets of points rather than the total number of points in the solution. We also prove a lower bound based on *Set Cover Conjecture* for the same problem.

We give a reduction from the *Set Cover* as follows. The input to a *Set Cover* is a universe $U = \{1, 2, \ldots, n\} = [n]$, and a family $\mathcal{F} \subseteq 2^U$, and an integer k. The objective is to find a subfamily $\mathcal{F}' \subseteq \mathcal{F}$ such that $|\mathcal{F}'| \leq k$ and $U = \bigcup_{A \in \mathcal{F}'} A$.

Lemma 2 ([12]). *The Set Cover problem parameterized by $|U|$ admits no polynomial compression unless $\mathsf{NP} \subseteq \mathsf{coNP/poly}$.*

Conjecture 1 (Set Cover Conjecture [10]) *The Set Cover problem cannot be solved in $O^*((2-\epsilon)^{|U|})$ time.*

Lemma 3 ([11]). *Let Π_1, Π_2 be two parameterized problems and suppose that there exists a polynomial parameter transformation from Π_1 to Π_2. Then, if Π_1 does not admits a polynomial compression, neither does Π_2.*

Reduction: Let $(U, \mathcal{F}, k)$ be an instance of the *Set Cover* problem such that $U = \{x_1, x_2, \ldots, x_n\}$ and $\mathcal{F} = \{S_1, \ldots, S_m\}$. For every $i \in [n]$, let us denote $\mathsf{oc}(x_i) = \{j \in [m] | x_i \in S_j\}$ and let $\delta = \max\{|\mathsf{oc}(x_i)| : i \in [n]\}$. Informally speaking, δ is the maximum number of sets at which an element of the universe can occur. For every $x_i \in U$, we arrange the indices of $\mathsf{oc}(x_i)$ in increasing order. **(i)** We construct the set of intervals $\mathcal{I} = \{[(i-1)\delta + 1, i\delta] : i \in \{1, 2, \ldots, n\}\}$, **(ii)** we construct the point set P and its partition into m nonempty sets $P_1, \ldots, P_m$ as follows, Observe that for every $x_i \in U$, the set $\mathsf{oc}(x_i)$ denotes the increasing order at which x_i occurs across several sets in $\mathcal{F}$. For every $j \in [m]$, we create P_j as follows. Consider an arbitrary $x_i \in S_j$. If j is the r'th occurrence of x_i in $\mathsf{oc}(x_i)$, then we add the point $(i-1)\delta + r$ into P_j. In other words, every occurrence of an element is represented by a unique point in a specific subset of points, **(iii)** finally, we denote $P = P_1 \cup \ldots \cup P_m$.

Observe that by construction, every element in U has its corresponding interval in $\mathcal{I}$. Also observe that the point $(i-1)\delta + r$ (corresponding to $x_i \in U$) hits the interval $[(i-1)\delta, i\delta]$ since $r \leq \delta$. Hence, the sets P_j's are pairwise disjoint. Because the occurrence of an element x_i across distinct sets in the family is represented by various points in the same interval $[(i-1)\delta, i\delta]$. Also observe that for all $x_i \in U$, $\mathsf{oc}(x_i) \neq \emptyset$.

Next consider $\mathcal{F}' = \{S_{j_1}, \ldots, S_{j_k}\}$ be a subfamily of size at most k that covers U. Then, $P' = P_{j_1} \cup \ldots \cup P_{j_k}$ is the solution to *DWCHSI-D* instance. The idea is that the corresponding interval $[(i-1)\delta + 1, i\delta]$ will be hit by r'th $(r \leq \delta)$ occurrence of x_i in P_{j_r}. Therefore, there are k subsets points $P_{j_1} \cup \ldots \cup P_{j_k}$ that hit all intervals and satisfy the constraints. On the other hand, let $P' = P_{j_1} \cup \ldots \cup P_{j_k}$ be k subsets of points that hit all the intervals. If interval $[(i-1)\delta + 1, i\delta]$ is hit by a point in P_{j_r}, then, the element x_i has t'th occurrence $(t \leq \delta)$ in S_{j_r}. Hence, $\mathcal{F}' = \{S_{j_1}, \ldots, S_{j_k}\}$ covers U. This leads to the following lemma.

Lemma 4. *For a Set Cover instance $(U, \mathcal{F}, k)$, the instance $(U, \mathcal{F}, k)$ has a feasible solution of size at most k if and only if there are k subsets of points that hit all intervals in DWCHSI-D instance we created by the reduction.*

Lemmas 2, 4, and Conjecture 1 lead the following theorem.

Theorem 2. *The DWCHSI-D problem parameterized by $|\mathcal{I}|$ admits no polynomial compression unless* $\mathsf{NP} \subseteq \mathsf{coNP/poly}$. *Moreover, unless Conjecture 1 fails, the DWCHSI-D problem cannot be solved in $O^*((2-\epsilon)^{|\mathcal{I}|})$ time.*

2.4 Polynomial kernel for subsets of size at most t for fixed t

We provide a polynomial kernel for the *DCHSI-tD* problem parameterized by solution size (k). Recall that all subsets of points in the input instance has size at most t. Thus we have the following:

Theorem 3. *When Reduction Rules 1, 2, 3, and 4 are not applicable, then an instance of the DCHSI-tD problem has k intervals and at most $O(t^2k^t)$ points. Hence, the DCHSI-tD problem parameterized by solution size (k) admits a kernel with k intervals and $O(t^2k^t)$ points.*

3 Subset size at most 2, disjoint intervals

In this section, we show that the *CHSI-2D* problem can be solved in polynomial-time. We first convert the *CHSI-2D* problem to an equivalent problem, where the size of each subset is exactly 2. We call it as *CHSI-2D-exact* problem. Next, we reduce the *CHSI-2D-exact* problem to the edge cover problem[1] in a graph.

***CHSI-2D-exact* problem instance construction:**

Let $\mathcal{I} = \{i_1, i_2, \ldots, i_\gamma\}$ be a set of pairwise disjoint intervals and P be a set of points partitioned into subsets $\{P_1, P_2, \ldots, P_d\}$, where $2d \geq \gamma$. Note that, in the given instance each P_i, $1 \leq i \leq d$, contains at most 2 points. Without loss of generality, we assume that each point hits at least one interval in the set $\mathcal{I}$, otherwise, we can do the following. If any set (having one or two points) does not hit any interval then we delete that set from P. If only point α of a set hits an interval $\tilde{i}$ (the other point does not) then the following cases can happen:
(i) $\tilde{i}$ is only hit by α, then we include the set containing α into our solution and delete the interval from set $\mathcal{I}$,
(ii) $\tilde{i}$ is hit by other points also apart from α, then remove the set containing α from our consideration. Next, for each subset of $P_\ell \in P$ that contains exactly one point, say p_ℓ^1, we do the following, as illustrated in Figure 1(a) :

Take one extra (dummy) point $p_\ell^2 \in P_\ell$, take two extra (dummy) points $\tilde{p}_\ell^1$ and $\tilde{p}_\ell^2$ that belongs to a single new set, say $\tilde{P}_\ell$, and take two additional (dummy) disjoint intervals i'_ℓ and i''_ℓ. We place the intervals i'_ℓ and i''_ℓ to the extreme right of the current configuration such that these two intervals do not overlap with the existing configuration. The points p_ℓ^2 and $\tilde{p}_\ell^1$ hit the interval i'_ℓ and the point $\tilde{p}_\ell^2$ hit the interval i''_ℓ.

[1] The edge cover problem defined on a graph finds the set of edges of a minimum size such that every vertex of the graph is incident to at least one edge of the set.

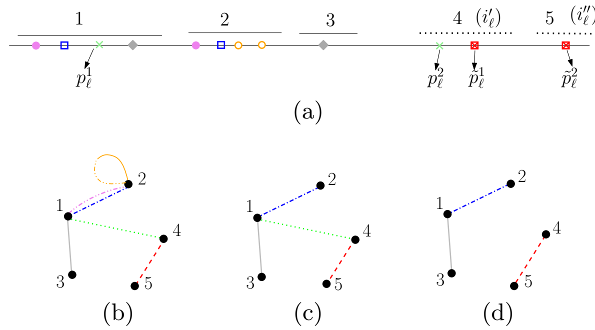

Fig. 1: (a) From the *CHSI*-2*D* problem to the *CHSI*-2*D*-*exact* problem: p_ℓ^2 is a dummy point for the single point p_ℓ^1, $\tilde{p}_\ell{}^1$ and $\tilde{p}_\ell{}^2$ are the dummy points for the dummy intervals i'_ℓ and i''_ℓ. (b) converting into an equivalent graph G_τ (c) removing self-loops and parallel edges (d) corresponding edge-cover of G_τ (for interpretation of the references to color in the figure legends, the reader is referred to the web version).

Let τ be an original instance of the *CHSI*-2*D* problem with exactly one set P_ℓ that contains exactly one point and τ^* be the instance constructed above. We have the following lemma.

Lemma 5. *The instance τ has a solution of size s if and only if either (i) τ^* has a solution of size $s+3$ if P_ℓ is in the solution of τ or (ii) τ^* has a solution of size $s+2$ if P_ℓ is not in the solution of τ.*

We repeat the above procedure for each subset of P one by one that contains exactly 1 point. Therefore, in the final instance, say τ' all the subsets contain exactly 2 points. By repeated application of Lemma 5 we ensure that finding a solution of the *CHSI*-2*D* problem is equivalent to finding a solution to the *CHSI*-2*D*-*exact* problem. Observe that τ' can contain at most 3γ number of intervals and at most $4d$ number of points partitioned into at most $2d$ subsets. The instance τ' can be constructed in linear time with respect to the number of intervals, points, and subsets. Hence, in polynomial-time, we can get a solution of the *CHSI*-2*D* problem from the *CHSI*-2*D*-*exact* problem.

Edge cover instance construction: Let us consider the modified instance τ' of the *CHSI*-2*D*-*exact* problem contains a set $\mathcal{I}' = \{i_1, i_2, \ldots, i_{\gamma'}\}$, $\gamma' \leq 3\gamma$, of pairwise disjoint intervals and a set of points $P = \{P_1, P_2, \ldots, P_{d'}\}$ where each P_i, $1 \leq i \leq d'$, $d' \leq 2d$, contains exactly 2 points. We construct a graph $G_{\tau'} = (V, E)$ as follows:

Construction: For each interval $i_l \in \mathcal{I}'$, take a vertex $v_l \in V$ and for each subset P_j containing points p_j^1 and p_j^2, we take an edge $e_j \in E$. The edge e_j connects the vertices $v_{l'}$ and $v_{l''}$ if and only if the interval $i_{l'}$ corresponding to $v_{l'}$ contains the point p_j^1 and the interval $i_{l''}$ corresponding to $v_{l''}$ contains the point p_j^2. Note that, if a single interval i_l contains both the points p_j^1 and p_j^2 then the edge e_j is a self loop on the vertex v_l. If both intervals $i_{l'}$ and $i_{l''}$ are hit by the two points of the subset $P_{j'}$ as well as by the two points of the subset $P_{j''}$, then there are parallel edges between $v_{l'}$ and $v_{l''}$.

We now process (removing redundant and trivial edges) the graph $G_{\tau'}$ without affecting the size of the solution. If there are parallel edges between two vertices of $G_{\tau'}$, then we keep exactly one edge between them and remove the remaining edges. Note that this modification does not affect the size of the optimal solution, since the subsets corresponding to the parallel edges hit the same set of intervals, and hence only one of them can be selected in the optimal solution. Let the resultant graph be $\tilde{G_\tau}$. Next, we remove all the self-loops from $\tilde{G_\tau}$. Let the resultant graph be $\tilde{G'}$ (Figure 1(c)). Let v be a vertex in $\tilde{G'}$. Here two cases may arise based on the number of the loops and edges incident on v; Case (i) only loops (≥ 1) are incident on v and Case (ii) loops as well as some other edges are incident on v. In Case (i), the interval corresponding to v covers the subsets (both points) corresponding to the loops. We arbitrarily choose one loop and insert the corresponding subset into our solution P' and remove v from the graph $\tilde{G_\tau}$. However, in Case (ii), the interval corresponding to v covers the subsets (both points) corresponding to the loops and at least one subset that has exactly one point hit the interval. In this case, we delete the self-loops incident on v, because choosing an edge as opposed to choosing a self-loop incident on v does not worsen the size of the solution. After processing the parallel edges and self-loops let the resultant graph is $G' = (V', E')$.

Next, we find an edge cover (see Figure 1(d)) in the graph G' using the maximum matching algorithm and then greedily add a minimum number of edges such that all the vertices are covered. We add all the points corresponding to those edges to our solution P'. Given a graph G' of n vertices and m edges then we can find its edge cover in $O(mn)$-time [16,21]. Thus we have the following:

Theorem 4. *The CHSI-2D-exact problem (hence the CHSI-2D problem) can be solved in $O(n \log n + \gamma n)$ time, where γ denotes the number of intervals in $\mathcal{I}$.*

4 Subset size at most 3, disjoint intervals

We now prove that the *CHSI*-3*D* problem is NP-hard. We give a reduction from the *Positive-1-in-3-SAT* problem that is known to be NP-complete [16,23].
***Positive-1-in-3-SAT* [16,23]:** We are given a 3-SAT formula ϕ with m clauses and n variables such that each clause contains exactly three positive literals, the objective is to decide whether there exists an assignment of truth values to the variables of ϕ such that exactly one literal is true in each clause of ϕ.
Reduction: We create an instance I_ϕ of the *DCHSI*-3*D* problem from an instance ϕ of the *Positive-1-in-3-SAT* problem as follows.
Overall Structure: We place the variable and clause gadgets one by one from left to right on a line L (see Figure 2 for a schematic diagram). To the left, place the variable gadgets, and after that place the clause gadgets one after another.

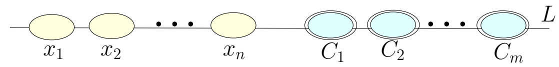

Fig. 2: Overall structure

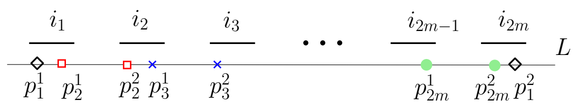

Fig. 3: A variable gadget.

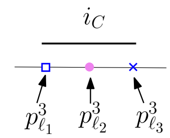

Fig. 4: A clause gadget.

Variable Gadget: For each variable, we take $2m$ subsets of points and each subset P_i contains two points p_i^1 and p_i^2, for $1 \leq i \leq 2m$. The points are ordered left-to-right on a real line L as $p_1^1, p_2^1, p_2^2, \ldots, p_{2m}^1, p_{2m}^2, p_1^2$. We also take $2m$ unit intervals $\{i_1, i_2, \ldots, i_{2m}\}$ such that interval i_1 is hit by points p_1^1 and p_2^1, i_{2m} is hit by points p_{2m}^2 and p_1^2, and for $2 \leq j \leq 2m-1$, interval i_j is hit by points p_j^2 and p_{j+1}^1. See the construction in Figure 3. Observe that there are exactly two optimal solutions that hit the intervals: either $G_1 = \{P_1, P_3, \ldots, P_{2m-1}\}$ or $G_2 = \{P_2, P_4, \ldots, P_{2m}\}$, each solution contains $2m$ points.
Clause Gadget: Let C be a clause that contains the three positive literals x_i, x_j, and x_k. Also let x_i is the l_1-th, x_j is the l_2-th, and x_k is the l_3-th occurrences in the formula ϕ. For C, the gadget consists of three points $p_{l_1}^3$, $p_{l_2}^3$, and $p_{l_3}^3$ and one interval i_C that is hit by these three points. The point $p_{l_1}^3$ is in the subset P_{l_1} of the gadget of x_i. Similarly, the points $p_{l_2}^3$ and $p_{l_3}^3$ is in the subsets P_{l_2} and P_{l_3} of the gadget of x_j and x_k respectively.

This completes the construction that can be done in polynomial-time with respect to the number of variables and clauses in ϕ. We have the following lemma.

Lemma 6. *Exactly one literal is true in every clause of ϕ if and only if the intervals in I_ϕ are hit by $2mn + m$ points.*

Theorem 5. *The DCHSI-3D (hence the CHSI-3D) problem is NP-hard.*

4.1 Approximation algorithms

The *CHSI-kD* problem can be reduced to the standard weighted set cover problem where the size of each set is bounded by k. Thus, we can obtain a H_k factor approximation [27] for the problem. In particular, when $k = 3$ (the *CHSI-3D* problem), we get a $\frac{11}{6}$ approximation.

Lemma 7. *The CHSI-3D problem can be approximated by a $H_3 = \frac{11}{6}$-factor.*

A $\frac{5}{3}$-factor approximation algorithm: We now propose an improved $\frac{5}{3}$ factor approximation algorithm for the *CHSI-3D* problem in Algorithm 1.

For each subset P_i we define its ρ value as $\left[\frac{\text{number of intervals hit by the points in } P_i}{\text{number of points in } P_i}\right]$.

Algorithm 1: $\frac{5}{3}$-factor approximation algorithm

Input: $\mathcal{I}$: set of intervals, $P = \{P_1 \cup \ldots \cup P_d\}$: set of points where $|P_i| \leq 3$, $P_i \cap P_j = \emptyset$.
Output: *A subset P' of P that hits all the intervals of $\mathcal{I}$.*

```
P' ← ∅;
while all the intervals are not "hit" do
    Sort the ρ values of the subsets;
    P' ← P' ∪ set having largest ρ;
    Remove that subset and the intervals those are "hit";
    update the corresponding ρ values of remaining subsets;
return P';
```

Algorithm 1 picks a subset with the highest ρ value in each iteration to P' and also updates the ρ values of the subsets in $P \setminus P'$ after removing the intervals those are hit by P'. As we have disjoint intervals and each subset contains at most 3 points, the possible ρ values are $1, \frac{2}{3}, \frac{1}{2}, \frac{1}{3}$. We select the subsets with respect to the non-decreasing order of their ρ values. It can be justified that Algorithm 1 returns a $\frac{5}{3}$-approximate solution, by ensuring that at each iteration, our algorithm chooses at most $\frac{5}{3}$ points compared to the optimum solution for those set of intervals hit till that step. Thus, we conclude the following theorem.

Theorem 6. *The CHSI-3D problem can be approximated within a factor of $\frac{5}{3}$ in $O(n \log n)$ time.*

5 Conclusion

We study a constrained version of the *Geometric Hitting Set* problem where the intervals are either disjoint (*CHSI-tD* problem) or overlapping (*CHSI-tO* problem). We show that the *DCHSI-D* problem is in FPT. We also prove that the *CHSI-tD* problem is NP-Hard for $t = 3$ while the *CHSI-tD* problem is polynomial-time solvable for $t = 2$ and gave a $\frac{5}{3}$-factor approximation algorithm for *CHSI-3D*. It would be interesting to investigate whether the approximation can be generalized for any t. The computational complexity of the *CHSI-tO* problem for $t = 2$, the parameterized complexity and approximation algorithm for the *CHSI-tO* problem, for any $t \geq 2$ also remains interesting open questions.

References

1. F. N. Abu-Khzam. A kernelization algorithm for d-Hitting Set. *J. Comput. Syst. Sci.*, 76(7):524–531, 2010.
2. A. Agrawal, P. Jain, L. Kanesh, and S. Saurabh. Parameterized complexity of conflict-free matchings and paths. *Algorithmica*, 82(7):1939–1965, 2020.
3. A. V. Aho, J. E. Hopcroft, and J. D. Ullman. *The Design and Analysis of Computer Algorithms.* Addison-Wesley, 1974.

4. B. Aronov, E. Ezra, and M. Sharir. Small-size ϵ-nets for axis-parallel rectangles and boxes. *SIAM J. Comput.*, 39(7):3248–3282, 2010.
5. G. Ausiello, A. D'Atri, and M. Protasi. Structure preserving reductions among convex optimization problems. *J. Comput. Syst. Sci.*, 21(1):136–153, 1980.
6. H. Brönnimann and M. T. Goodrich. Almost optimal set covers in finite VC-dimension. *Discret. Comput. Geom.*, 14(4):463–479, 1995.
7. K. L. Clarkson and K. R. Varadarajan. Improved approximation algorithms for geometric set cover. *Discret. Comput. Geom.*, 37(1):43–58, 2007.
8. A. Cornet and C. Laforest. Total domination, connected vertex cover and steiner tree with conflicts. *Discret. Math. Theor. Comput. Sci.*, 19(3), 2017.
9. A. Cornet and C. Laforest. Graph problems with obligations. In *COCOA*, pages 183–197, 2018.
10. M. Cygan, H. Dell, D. Lokshtanov, D. Marx, J. Nederlof, Y. Okamoto, R. Paturi, S. Saurabh, and M. Wahlström. On problems as hard as CNF-SAT. *ACM Trans. Algorithms*, 12(3):41:1–41:24, 2016.
11. M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms.* Springer, 2015.
12. M. Dom, D. Lokshtanov, and S. Saurabh. Kernelization lower bounds through colors and ids. *ACM Trans. Algorithms*, 11(2):13:1–13:20, 2014.
13. U. Feige. A threshold of $\ln n$ for approximating set cover. *J. ACM*, 45(4):634–652, 1998.
14. J. Flum and M. Grohe. *Parameterized Complexity Theory.* Springer, 2006.
15. R. J. Fowler, M. Paterson, and S. L. Tanimoto. Optimal packing and covering in the plane are NP-Complete. *Inf. Process. Lett.*, 12(3):133–137, 1981.
16. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.
17. O. Goldschmidt, D. S. Hochbaum, and G. Yu. A modified greedy heuristic for the set covering problem with improved worst case bound. *Inf. Process. Lett.*, 48(6):305–310, 1993.
18. A. Jacob, D. Majumdar, and V. Raman. Parameterized complexity of conflict free set cover. *Theory Comput. Syst.*, 65(3):515–540, 2021.
19. P. Jain, L. Kanesh, and P. Misra. Conflict free version of covering problems on graphs: Classical and parameterized. *Theory Comput. Syst.*, 64(6):1067–1093, 2020.
20. J. M. Kleinberg and É. Tardos. *Algorithm design.* Addison-Wesley, 2006.
21. E. L. Lawler. *Combinatorial optimization: networks and matroids.* Courier Corporation, 2001.
22. N. H. Mustafa, R. Raman, and S. Ray. Quasi-polynomial time approximation scheme for weighted geometric set cover on pseudodisks and halfspaces. *SIAM J. Comput.*, 44(6):1650–1669, 2015.
23. T. J. Schaefer. The complexity of satisfiability problems. In *Proceedings of the 10th Annual ACM Symposium on Theory of Computing*, pages 216–226. ACM, 1978.
24. R. Van Bevern. Towards optimal and expressive kernelization for d-hitting set. *Algorithmica*, 70(1):129–147, 2014.
25. R. van Bevern and P. V. Smirnov. Optimal-size problem kernels for d-hitting set in linear time and space. *Information Processing Letters*, 163:105998, 2020.
26. R. van Bevern, O. Y. Tsidulko, and P. Zschoche. Representative families for matroid intersections, with applications to location, packing, and covering problems. *Discrete Applied Mathematics*, 298:110–128, 2021.
27. V. V. Vazirani. *Approximation algorithms.* Springer, 2001.
28. H. Yinnone. On paths avoiding forbidden pairs of vertices in a graph. *Discret. Appl. Math.*, 74(1):85–92, 1997.