

Partial interval multicover: Approximation and complexity

Peng Li^{a,b}, Xiangzhi Tu^b, Zhao Zhang^{c,*}, Yixin Cao^d

^a School of Mathematics and Computer Science, Hanjiang Normal University, Shiyao, China

^b School of Mathematical Sciences, Chongqing University of Technology, Chongqing, China

^c School of Mathematical Sciences, Zhejiang Normal University, Jinhua, China

^d Department of Computing, Hong Kong Polytechnic University, Hong Kong, China

ARTICLE INFO

Section Editor: Dr. Vladimiro Sassone

Handling Editor: Dr. Katie Harris

Keywords:

Partial cover

Multicover

Polynomial-time approximation scheme

Dynamic programming

NP-completeness

ABSTRACT

We study a variant of set cover on the real line, where elements are points, sets are intervals, and each point has an integer demand; a point is fully covered when it is contained in at least its demand many chosen intervals. The objective is to select the fewest intervals that fully cover at least a specified number of points. We present the first polynomial-time approximation scheme (PTAS) for the unweighted version of this problem and show that a natural weighted generalization is NP-complete.

1. Introduction

In the *set cover* problem, we are given a universe U and a collection S of subsets of U , and asked to find a smallest sub-collection of S whose union is U [1]. This classic problem is central to combinatorial optimization and has many well-studied variants. In *partial set cover* [2], the goal is to minimize the cost of covering at least a prescribed number of elements. In *set multicover* [3], each element x is additionally associated with a nonnegative demand $d(x)$ and must be covered at least $d(x)$ times; that is, at least $d(x)$ sets in the chosen sub-collection contain x . Combining these yields the *partial set multicover* problem, which is significantly harder than set cover [4].

Set cover becomes easier under structural restrictions on the family of subsets. For example, it reduces to vertex cover when each element appears in exactly two subsets, and it admits specialized algorithms in geometric settings where sets are induced by shapes such as disks, rectangles, or line segments. A natural question is the complexity of partial set multicover under such restrictions. Notably, the *densest k -subgraph* problem, which asks for a k -vertex subgraph with the maximum number of edges, admits the following formulation: let the elements be the edges, each with demand two, and create one subset per vertex containing all incident edges. In this formulation, the requirement that each edge be covered at least twice corresponds to the requirement that both of its endpoints be selected. Selecting k subsets then fully covers exactly the edges induced by those k vertices.

We consider a geometric setting that is arguably the simplest of its kind: the elements are points on the real line and the sets are intervals. Specifically, we study the *partial interval multicover* problem. Given a set of n intervals, a set of points on the real line with associated coverage demands, and an integer p , select a minimum number of intervals that fully cover at least p points (a point is fully covered when it is contained in at least its demand many chosen intervals). See Fig. 1 for an example. This problem was formulated

* Corresponding author.

E-mail addresses: lipengsjtu@163.com (P. Li), tuxiangzhi0928@163.com (X. Tu), zhaozhang@zjnu.cn (Z. Zhang), yixin.cao@polyu.edu.hk (Y. Cao).

<https://doi.org/10.1016/j.tcs.2026.115958>

Received 3 November 2025; Received in revised form 21 February 2026; Accepted 3 April 2026

Available online 10 April 2026

0304-3975/© 2026 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

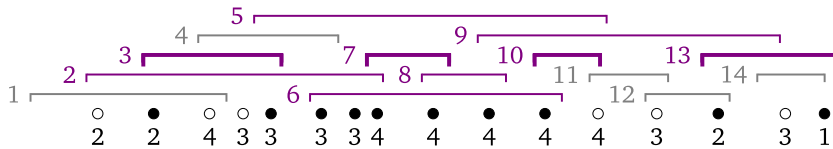


Fig. 1. An instance with 14 intervals and 16 points, whose demands are specified below, and $p = 10$. The nine violet intervals fully cover the ten solid points.

by Ran et al. [5], who presented a 2-approximation. The problem is tractable when only one of the two requirements is imposed, i.e., either partial coverage with unit demands or full multiple coverage of all points. Golab et al. [6] studied the partial interval (single) cover problem (with unit demands) in the context of sequential dependencies in databases and provided an $O(n^2)$ -time algorithm. Related polynomial-time solvability results are known when requiring full multiple coverage of all points, and the techniques and insights of [7] are closely connected to the complexity of our problem.

Our contributions and techniques. Because intervals admit a natural linear order, dynamic programming is a natural approach. However, a naïve decomposition (e.g., by the i leftmost intervals for $i = 1, 2, \dots, n$) does not work. The first challenge is to identify subproblems anchored at a suitably chosen point so that everything to the right of that point can be ignored when deciding coverage to its left.

Since both the interval and point sets are finite, we may assume all interval endpoints are distinct and that intervals are indexed by increasing left endpoint; see Fig. 1. Fix an optimal solution J . Without loss of generality, we can assume J is upward-closed (formalized in Proposition 1): if $I \in J$ and $I' \supseteq I$ is an input interval, then replacing I by I' does not worsen the solution, so we may take I' instead. For $i = 1, 2, \dots, n$, let z_i be the right endpoint of I_i , and let K_i be the set of intervals that contain z_i .

We say that an interval is *minimal* if it contains no other interval from J . We then greedily select a maximal set of pairwise disjoint minimal intervals from J by scanning the intervals of J from left to right (increasing left endpoints) and adding each minimal interval that is disjoint from all previously chosen ones. Let j_1, j_2, \dots, j_q be the indices of the selected intervals (e.g., 3, 7, 10, 13 in Fig. 1). By construction, $J \subseteq K_{j_1} \cup K_{j_2} \cup \dots \cup K_{j_q}$; for instance, in Fig. 1, $K_{j_1} = K_3 = \{2, 3, 4, 5\}$, $K_{j_2} = K_7 = \{5, 6, 7, 8\}$, $K_{j_3} = K_{10} = \{5, 9, 10, 11\}$, and $K_{j_4} = K_{13} = \{13\}$. All points up to z_{j_i} (inclusive) must be covered by intervals in $J \cap (K_{j_1} \cup K_{j_2} \cup \dots \cup K_{j_i})$, because any interval covering such a point must contain at least one of the anchor points z_{j_1}, \dots, z_{j_i} . Crucially, intervals in K_{j_i} may also contribute to covering points beyond z_{j_i} (since $J \cap K_{j_i} \cap K_{j_{i+1}}$ may not be empty). This motivates defining dynamic-programming subproblems indexed by tuples (p', i, K) , where p' is the number of points fully covered up to z_{j_i} (inclusive), $i \in \{1, 2, \dots, n\}$, and $K \subseteq K_i$ encodes which intervals covering z_{j_i} are selected and may carry over to influence coverage to the right.

Enumerating all subsets of K_i would yield a dynamic program, but it is infeasible because $|K_i|$ can be $\Omega(n)$. For any fixed $\epsilon \in (0, 1)$, we instead construct a polynomial-size family of representative subsets of K_i with the property that, for every feasible choice $J \cap K_i$, there exists a representative that is a $(1 + \epsilon)$ -approximation to $J \cap K_i$ with respect to its contribution to covering points up to and including z_i .

A technical issue is overlap across anchors: for $1 \leq j < i \leq n$, the intersection $J \cap K_j \cap K_i$ may be nonempty, and naïvely accounting for both K_j and K_i would double count those intervals. Observe that all intervals in $K_j \cap K_i$ strictly contain I_j ; we therefore exclude such intervals from the count when processing K_i , charging them only once at their earliest relevant anchor. With these representative families and the de-duplication rule, the dynamic program runs in polynomial time and achieves an approximation ratio of $1 + \epsilon$.

Theorem 1. *There is a polynomial-time approximation scheme for partial interval multicover.*

The complexity of partial interval multicover has remained open. In lieu of resolving it, we study a weighted generalization. Each interval has a nonnegative weight and each point has a nonnegative value (while retaining its coverage demand). The goal is to select intervals of minimum total weight that fully cover a set of points whose total value is at least p . A point is fully covered if the sum of the weights of the chosen intervals containing it meets or exceeds its demand. We show that this weighted version is NP-complete.

Theorem 2. *The weighted partial interval multicover problem is NP-complete.*

2. A polynomial-time approximation scheme

As we will use power sets of solutions, it would be more convenient for us to denote the solution as a set of interval indices instead of the intervals themselves. We formally define the *partial interval multicover* problem as follows.

Input: A collection \mathcal{I} of n intervals $\{I_1, I_2, \dots, I_n\}$, a set X of points on the real line, a demand function $t : X \rightarrow \mathbb{N}$, and a nonnegative integer p .

Output: A minimum subset $J \subseteq \{1, 2, \dots, n\}$ such that $|\{i \in J \mid x \in I_i\}| \geq t(x)$ for at least p points $x \in X$.

For a subset $J \subseteq \{1, 2, \dots, n\}$ and a point $x \in X$, we define

$$c(J, x) = \left| \{j \in J \mid x \in I_j\} \right|.$$

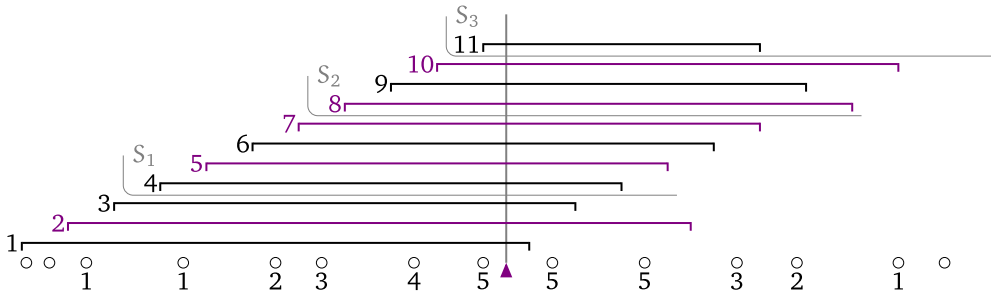


Fig. 2. A sequence of intervals sharing a common point. The number below a point x is $c(K^*, x)$, where K^* comprises the purple intervals. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Hence, a point is fully covered by intervals in J if $c(J, x) \geq t(x)$. For a subset $Y \subseteq X$, we define

$$c(J, Y) = |\{x \in Y \mid c(J, x) \geq t(x)\}|.$$

Thus, the problem is asking for a minimum subset $J \subseteq \{1, 2, \dots, n\}$ such that $c(J, X) \geq p$.

Proposition 1. For any instance of partial interval multicover, there is an optimal solution J satisfying

$$I_j \subset I_i \wedge j \in J \implies i \in J. \tag{1}$$

Proof. Let J^* be an optimal solution of the instance. Suppose that there exist $j \in J^*$ and $i \notin J^*$ such that $I_j \subset I_i$. For all points $x \in X$,

$$c(J^* \cup \{i\} \setminus \{j\}, x) = c(J^* \setminus \{j\}, x) + c(\{i\}, x) \geq c(J^* \setminus \{j\}, x) + c(\{j\}, x) = c(J^*, x),$$

where the inequality holds because $I_j \subset I_i$. Thus, the set $J^* \cup \{i\} \setminus \{j\}$ is also an optimal solution of the instance. We may repeat this process until no such pairs exist: it must terminate because every switch increases the total length of the chosen intervals. The resulting solution satisfies (1). \square

We may assume without loss of generality that all the endpoints of the intervals are distinct: we can always tweak the instance to make it happen because both I and X are finite. As a result, when an interval contains another, the containment must be proper. For notational convenience, we further shift and scale the intervals and points such that for all $i = 1, 2, \dots, n$, the left endpoint of I_i is i . Let z_i denote the right endpoint of I_i , hence $I_i = [i, z_i]$.

2.1. Representatives for intervals sharing a point

The proof of Proposition 1 relies on a simple exchange principle: it is safe to replace a subset J' of a solution by another subset of the same cardinality that yields no worse coverage for every point.¹ This extends naturally to approximation. Fix $\epsilon \in (0, 1)$. We say that a set R is an ϵ -representative for set J' if $|R| \leq (1 + \epsilon)|J'|$ and R provides at least as much coverage as J' for every point.

We will construct ϵ -representatives only for sets of intervals that share a common point. Before giving the general procedure, we illustrate the idea using Fig. 2, where the common point is marked by a triangle, and $K = \{1, 2, \dots, 11\}$ comprise all intervals containing the common point. Set $\epsilon = 0.45$. We look for an ϵ -representative for $K^* = \{2, 5, 7, 8, 10\}$ as follows, using a parameter d that will later be chosen as a fraction of $|K^*|$.

1. Insurance prefix. Take the d th interval of $K \setminus K^*$ (inclusive) and all intervals in K that precede it. In the figure, the first two intervals of $K \setminus K^*$ are $\{1, 3\}$, so we take $\{1, 2, 3\}$.
2. Partition. On the remaining suffix, partition into contiguous parts so that, except possibly the last part, each part contains exactly d intervals from K^* . In the figure, the parts are $\{4, 5, 6, 7\}, \{8, 9, 10\}, \{11\}$.
3. Right-shift within parts. In each part, replace the intervals from K^* by the same number of intervals whose right endpoints are as large as possible within that part.

For the example (with $d = 2$), the resulting representative set is

$$R = \{1, 2, 3, 6, 7, 8, 10\},$$

with $|R| = |K^*| + d \leq (1 + \epsilon)|K^*|$.

Right-shifting intervals within a part may reduce coverage only to the left, never to the right. Within any single part, at most d intervals are shifted, so any given point can lose coverage from at most d intervals in that part. Crucially, each point can be affected by at most one part (because the parts are disjoint contiguous blocks along the line, and all intervals contain the same anchor point).

¹ It suffices to consider the points in X , but in this subsection, the arguments apply to all the points on the real line. Hence, we do not specify “in X .”

1. $\mathcal{R}_k^\epsilon \leftarrow \emptyset$;
2. **for each** subset $X \subseteq K_k^>$ with $|X| \leq \frac{1}{\epsilon}$ **do**
3. add $X \cup K_k^{\leq}$ to \mathcal{R}_k^ϵ ;
4. **if** $|K_k^>| \leq \frac{1}{\epsilon}$ **then return** \mathcal{R}_k^ϵ ;
5. **for** $d = 1, 2, \dots, \lfloor \epsilon |K_k| \rfloor$ **do**
6. **for each** $(d, \frac{1}{\epsilon})$ partition $\langle S_0, S_1, \dots, S_{s+1} \rangle$ of $K_k^>$ **do**
7. $X \leftarrow S_0$;
8. **for** $i = 1, 2, \dots, s$ **do**
9. add the d rightmost intervals in S_i to X ;
10. **for** $d' = 1, 2, \dots, \min\{d, |S_{s+1}|\}$ **do**
11. add the d' rightmost intervals in S_{s+1} to X ;
12. add $X \cup K_k^{\leq}$ to \mathcal{R}_k^ϵ ;
13. **return** \mathcal{R}_k^ϵ .

Fig. 3. Calculating representative subsets for K_k .

The d leftmost intervals taken from $K \setminus K^*$ serve as insurance to compensate for any such loss. Hence R provides coverage no weaker than K^* while using at most $(1 + \epsilon)|K^*|$ intervals. This “insurance plus right-shift” idea underlies the general construction in Lemma 1: for any large enough K^* , we identify a suitable insurance prefix S_0 and partition the suffix into parts S_1, \dots, S_{s+1} .

We now formalize the construction to obtain a family of polynomial size (in $|K|$ and $\frac{1}{\epsilon}$) that provides ϵ -representatives for an exponential number of subsets. For our purposes, it suffices to handle the following case: the candidate set K consists of all intervals containing z_i for some $i \in \{1, 2, \dots, n\}$, and K^* contains I_i together with all intervals that strictly contain I_i . For $i = 1, 2, \dots, n$, define

$$\begin{aligned}
 K_i &= \{j \mid z_i \in I_j\}, \\
 K_i^{\leq} &= \{j \in K_i \mid j \leq i\}, \\
 K_i^> &= \{j \in K_i \mid j > i\}.
 \end{aligned}$$

Note that K_i^{\leq} and $K_i^>$ form a partition of K_i , and $j \in K_i^{\leq}$ if and only if $I_i \subseteq I_j$. Also note that the indices in K_i do not need to be contiguous. Within $K_i^>$ we will use the natural order by increasing right endpoint; “rightmost” refers to this order.

We are now ready to present our procedure. Let $k \in \{1, 2, \dots, n\}$. For a positive integer d , an ordered partition $\langle S_0, S_1, \dots, S_{s+1} \rangle$ of $K_k^>$ is called a $(d, \frac{1}{\epsilon})$ partition of $K_k^>$ if $s \leq \frac{1}{\epsilon}$ and for all $i = 0, 1, \dots, s$,

- $|S_i| \geq d$, and
- $j_1 < j_2$ for every of $j_1 \in S_i$ and $j_2 \in S_{i+1}$ (i.e., parts are consecutive blocks in the chosen order).

We remark that the size of the last set S_{s+1} is unspecified. We then generate candidate representatives as in Fig. 3.

Lemma 1 (Approximation Guarantee of \mathcal{R}_k^ϵ). *For any $k \in \{1, 2, \dots, n\}$ and any $\epsilon \in (0, 1)$, we can compute, in $O(\epsilon n^{1/\epsilon+3})$ time, a collection \mathcal{R}_k^ϵ of $O(\epsilon n^{1/\epsilon+2})$ subsets of K_k such that, for any K^* with $K_k^{\leq} \subseteq K^* \subseteq K_k$, there exists $R \in \mathcal{R}_k^\epsilon$ satisfying:*

- $K_k^{\leq} \subseteq R$,
- $|R \setminus K_k^{\leq}| \leq (1 + \epsilon)|K^* \setminus K_k^{\leq}|$, and
- $c(R, x) \geq c(K^*, x)$ for all points x .

Proof. If $|K^* \cap K_k^>| = |K^* \setminus K_k^{\leq}| \leq \frac{1}{\epsilon}$, then steps 2–3 add K^* itself, and the claim holds. Otherwise, let $d = \lfloor \epsilon |K^* \cap K_k^>| \rfloor \geq 1$. If $|K_k^>| \leq \frac{1}{\epsilon}$, we already returned all small subsets (step 4). So assume $|K_k^>| \geq \frac{1}{\epsilon}$, and we show there exists a $(d, \frac{1}{\epsilon})$ -partition for which one of the sets produced in step 12 is an ϵ -representative for K^* . If $|K^*| \geq |K_k| - d$, then K_k (which is generated with the trivial partition $\langle K_k^> \rangle$) is already an ϵ -representative; otherwise, $d < |K_k \setminus K^*| = |K_k^> \setminus K^*|$.

We consider the following partition of $K_k^>$. Let j_0 be the index of the d th interval in $K_k^> \setminus K^*$, and set

$$S_0 = \{i \in K_k \mid k < i \leq j_0\}.$$

For $i = 1, 2, \dots$, as long as $|\{j \in K^* \mid j > j_{i-1}\}| > d$, let j_i be the index of the d th such vertex and set

$$S_i = \{j \in K_k \mid j_{i-1} < j \leq j_i\}.$$

Let s be the last such i , set $S_{s+1} = \{j \in K_k \mid j > j_s\}$, and set $d' = |K^* \cap S_{s+1}|$; by construction, $d' \leq d$.

Consider the set R constructed from this partition and d' by steps 7–12. Because $S_0 \subseteq R$,

$$\begin{aligned}
 |R \cap K_k^>| &= \left| \bigcup_{i=0}^{s+1} (R \cap S_i) \right| \\
 &= \sum_{i=0}^{s+1} |R \cap S_i| \\
 &= |S_0| + \sum_{i=1}^{s+1} |R \cap S_i| \\
 &= |S_0 \setminus K^*| + |S_0 \cap K^*| + \sum_{i=1}^{s+1} |K^* \cap S_i| \\
 &= d + \sum_{i=0}^{s+1} |K^* \cap S_i| \\
 &= d + \left| \bigcup_{i=0}^{s+1} (K^* \cap S_i) \right| \\
 &= d + |K^* \cap K_k^>| \\
 &\leq (1 + \epsilon) |K^* \cap K_k^>|.
 \end{aligned}$$

It remains to show that $c(R, x) \geq c(K^*, x)$ for all points x . It follows from $|R| > |K^*|$ when $x \in \bigcap_{i \in K_k} I_i$; it follows from $S_0 \subseteq R$ when $x \leq j_0$, and hence we assume otherwise.

- Case 1, $x > z_k$. Within each part S_i , the constructed set R uses at least as many intervals as K^* and, by construction, chooses the rightmost ones, so $c(R \cap S_i, x) \geq c(K^* \cap S_i, x)$ for all i , hence $c(R, x) \geq c(K^*, x)$.
- Case 2, $j_0 < x < z_k$. Let i be such that $j_i \leq x < j_{i+1}$. Any interval from $K^* \setminus R$ covering x lies in a single part, and there are at most d of them. The d intervals in $S_0 \setminus K^*$ (included in R) all cover x , compensating any loss. Thus, $c(R, x) \geq c(K^*, x)$.

We now analyze the running time and the size of \mathcal{R}_k^ϵ . Steps 2 and 3 enumerate $O(n^{1/\epsilon})$ subsets and take $O(n^{1/\epsilon+1})$ time. The number of (ordered, contiguous) partitions of $K_k^>$ into at most $\frac{1}{\epsilon}$ parts is $\binom{n+1/\epsilon}{1/\epsilon} = O(n^{1/\epsilon})$. For each d (there are $O(\epsilon n)$ choices), we iterate over all such partitions; each iteration costs $O(n^2)$ time to select rightmost elements and build X , yielding total time $O(\epsilon n^{1/\epsilon+3})$. The number of representatives added is $O(n^{1/\epsilon})$ from step 3 and $O(\epsilon n^{1/\epsilon+2})$ from step 12, so $|\mathcal{R}_k^\epsilon| = O(\epsilon n^{1/\epsilon+2})$. \square

2.2. The algorithm

We use dynamic programming to compute an approximate solution. The dynamic programming table has an entry for each tuple (p', i, R) , where $p' \in \{0, 1, \dots, p\}$, $i \in \{1, 2, \dots, n\}$, and R ranges over \mathcal{R}_i^ϵ . Let $X \cap (x_1, x_2] = \{x \in X \mid x_1 < x \leq x_2\}$. We set the initial values as

$$B(p', i, R) = \begin{cases} R & \text{if } c(R, X \cap (0, z_i]) \geq p', \\ \{1, 2, \dots, n\} & \text{otherwise.} \end{cases}$$

Intuitively, for each triple (p', i, R) , the entry $B(p', i, R)$ stores the best (i.e., smallest) set of intervals that we can obtain by using anchors up to z_i (i.e., only intervals whose index is at most i), subject to the following two conditions:

- all intervals in R (a representative subset of K_i) are included, and
- at least p' points in $X \cap (0, z_i]$ are fully covered.

If these conditions cannot be satisfied (i.e., no such set of intervals exists), we set $B(p', i, R) = \{1, 2, \dots, n\}$ as a dummy value indicating an infeasible state.

We then update $B(*, i, *)$ in the order of $i = 1, 2, \dots, n$. Intuitively, we try to extend a partial solution that ends at some earlier anchor z_j (represented by (p', j, R')) by adding R and accounting for the new points fully covered between z_j and z_i . If $B(p', i, R) = \{1, 2, \dots, n\}$, replace it with a minimum set from

$$\left\{ R \cup B(p' - c(R \cup R', X \cap (z_j, z_i]), j, R') \mid z_j < i, R' \in \mathcal{R}_j^\epsilon, p' \geq c(R \cup R', X \cap (z_j, z_i]) \right\}.$$

Note that we only apply transitions to entries that are still in the dummy state, thereby keeping $B(p', i, R)$ equal to the smallest set found so far for this tuple. After all updates, return the smallest set in

$$\left\{ B(p - c(R, X \cap (z_i, z_n)), i, R) \mid 1 \leq i \leq n, R \in \mathcal{R}_i^\epsilon, p \geq c(R, X \cap (z_i, z_n]) \right\}.$$

We summarize the algorithm in Fig. 4.

```

0. for each  $i = 1, 2, \dots, n$  do compute  $\mathcal{R}_i^\varepsilon$ ;
1. for each  $i = 1, 2, \dots, n$  do
2.   for each  $p' = 0, 1, \dots, p$  do
3.     for each  $R \in \mathcal{R}_i^\varepsilon$  do
4.        $B(p', i, R) \leftarrow \{1, 2, \dots, n\}$ ;
5.       if  $c(R, X \cap (0, z_i]) \geq p'$  then  $B(p', i, R) \leftarrow R$ ;
6.       else for each  $j$  s.t.  $z_j < i$  do
7.         for each  $R' \in \mathcal{R}_j^\varepsilon$  s.t.  $p' \geq c(R \cup R', X \cap (z_j, z_i])$  do
8.            $Y \leftarrow R \cup B(p' - c(R \cup R', X \cap (z_j, z_i]), j, R')$ ;
9.           if  $|Y| < |B(p', i, R)|$  then  $B(p', i, R) \leftarrow Y$ ;
10.   $X \leftarrow \{1, 2, \dots, n\}$ ;
11. for each  $i = 1, 2, \dots, n$  do
12.   for each  $R \in \mathcal{R}_i^\varepsilon$  s.t.  $p \geq c(R, X \cap (z_i, z_n])$  do
13.      $p' \leftarrow p - c(R, X \cap (z_i, z_n])$ ;
14.     if  $|B(p', i, R)| < |X|$  then  $X \leftarrow B(p', i, R)$ ;
15. return  $X$ .

```

Fig. 4. The approximation algorithm.

Proposition 2. For every tuple (p', i, R) :

- (i) $R \subseteq B(p', i, R)$;
- (ii) if $c(B(p', i, R), X \cap (0, z_i]) < p'$, then $B(p', i, R) = \{1, 2, \dots, n\}$.
- (iii) If $p' < p$, then $|B(p', i, R)| \leq |B(p' + 1, i, R)|$.

Proof. (i) By construction, whenever $B(p', i, R)$ is set (either directly in steps 4–5, or to $R \cup B(*, *, *)$ in a transition in steps 8–9), R is included. Hence, $R \subseteq B(p', i, R)$.

(ii) We prove by induction on i . The claim is immediate when $B(p', i, R)$ is $\{1, 2, \dots, n\}$ or R . For $i = 1$, one of these two cases holds, so the base case is trivial. Assume $i > 1$ and that the statement holds for all smaller indices. Suppose $B(p', i, R)$ was set via a transition: $B(p', i, R) = R \cup B(p' - c(R \cup R', X \cap (z_j, z_i]), j, R')$, for some $j < i$ and $R' \in \mathcal{R}_j^\varepsilon$. Let $p'' = p' - c(R \cup R', X \cap (z_j, z_i]) \geq 0$ (by the transition's precondition). Since $B(p'', j, R') \neq \{1, 2, \dots, n\}$, the inductive hypothesis implies

$$c(B(p'', j, R'), X \cap (0, z_j]) \geq p''.$$

Therefore,

$$c(B(p', i, R), X \cap (0, z_i]) \geq c(R \cup R', X \cap (z_j, z_i]) + p'' = p'.$$

As a result, if $c(B(p', i, R), X \cap (0, z_i]) < p'$, we cannot be in the transition case; hence $B(p', i, R)$ must be $\{1, 2, \dots, n\}$. In the transition case, the contribution from intervals chosen at z_j already covers p'' points in $(0, z_j]$, and the newly added intervals $R \cup R'$ cover $p' - p''$ points in $(z_j, z_i]$, so in total we reach p' points in $(0, z_i]$.

(iii) The statement is trivial when $B(p' + 1, i, R) = \{1, 2, \dots, n\}$. If $B(p' + 1, i, R) = R$, then $c(R, X \cap (0, z_i]) \geq p' + 1 > p'$, and $B(p', i, R) = R$ as well and the inequality holds. We proceed by induction in i . The base case must be one of the cases discussed above. In the general case, suppose $B(p' + 1, i, R) = R \cup B(p' + 1 - c(R \cup R', X \cap (z_j, z_i]), j, R')$ for some $j < i$ and $R' \in \mathcal{R}_j^\varepsilon$. By the inductive hypothesis applied at index j ,

$$|B(p' - c(R \cup R', X \cap (z_j, z_i]), j, R')| \leq |B(p' + 1 - c(R \cup R', X \cap (z_j, z_i]), j, R')|.$$

By (i), $R \subseteq B(p', i, R) \cap B(p' + 1, i, R)$, and hence

$$|B(p', i, R)| \leq |R \cup B(p' - c(R \cup R', X \cap (z_j, z_i]), j, R')| \leq |B(p' + 1, i, R)|.$$

This concludes the proof. \square

We say that an instance is *valid* if $c(\{1, 2, \dots, n\}, X) \geq p$. It is easy to check whether an instance is valid, and hence we focus on valid ones. By Proposition 2(ii), the algorithm always produces a feasible solution for a valid instance. Also note that Proposition 2(iii) can be stated as: $|B(p', i, R)| \leq |B(p'', i, R)|$ when $p' < p'' \leq p$. We are now ready to summarize the algorithm for the formal statement of Theorem 1.

Theorem 3. For any constant $\epsilon \in (0, 1)$, the algorithm in Fig. 4 runs in $O(\epsilon^2 p |X| n^{2/\epsilon+7})$ time and returns a $(1 + \epsilon)$ -approximation for partial interval multicover.

Proof. Let (I, X, t, p) be a valid instance (i.e., one with $c(\{1, \dots, n\}, X) \geq p$), and let J^* be an optimal solution satisfying (1). Initialize $j_0 = z_0 = 0$ and $J_0^* = \emptyset$. (For clarity, we write J_i^* for the partial solution built from J^* at stage i , to distinguish it from the algorithmic objects B_i .) For $i = 1, 2, \dots$, as long as $J_{i-1}^* \subset J^*$, define j_i to be the minimum index in J^* such that

$$j_i > z_{j_{i-1}} \quad \text{and} \quad \{I \mid I \subset I_{j_i}\} \cap J^* = \emptyset,$$

i.e., I_{j_i} is the first minimal interval in $J^* \setminus J_{i-1}^*$. Then define

$$J_i^* = \{j \in J^* \mid j < z_{j_i}\}.$$

Let q be the smallest index such that $J_q^* = J^*$. Note that $J_i^* \setminus J_{i-1}^* \subseteq K_{j_i}$ by the selection of j_i . Thus,

$$J_i^* = \bigcup_{k=1}^i K_{j_k} \cap J^*.$$

For $i = 1, 2, \dots, q$, let $X_i = \{x \in X \mid x \leq z_{j_i}\}$, let R_i be the representative in $\mathcal{R}_{j_i}^\epsilon$ for $K_{j_i} \cap J^*$ (Lemma 1), and let

$$B_i = B(c(J_i^*, X_i), j_i, R_i). \tag{2}$$

Let J be the algorithm's output; its feasibility is ensured by Proposition 2(ii). Note that

$$\begin{aligned} |B_q| &= \left| B(c(J_q^*, X_q), j_q, R_q) \right| \\ &= \left| B(c(J^*, X) - c(J^*, X \cap (z_{j_q}, z_n]), j_q, R_q) \right| \\ &\geq \left| B(p - c(R_q, X \cap (z_{j_q}, z_n]), j_q, R_q) \right| \\ &\geq |J|, \end{aligned}$$

where the last inequality follows from lines 11–14 of the algorithm. Thus, it suffices to show that

$$|B_q| \leq (1 + \epsilon) |J^*|. \tag{3}$$

We show by induction a stronger statement: for all $i = 1, 2, \dots, q$,

$$\left| B_i \setminus K_{j_i}^{\leq} \right| \leq (1 + \epsilon) \left| J_i^* \setminus K_{j_i}^{\leq} \right|. \tag{4}$$

This implies (3) because $K_{j_i}^{\leq} \subseteq J_i^*$ (note that J^* satisfies (1)) and $K_{j_i}^{\leq} \subseteq R_i \subseteq B_i$ (Lemma 1 and the algorithm).

Base case ($i = 1$). By Lemma 1, $c(R_1, X_1) \geq c(K_{j_1} \cap J^*, X_1) = c(J_1^*, X_1)$. Hence, $B_1 = R_1$, and (4) follows from Lemma 1.

Inductive step. Fix $i \in \{1, 2, \dots, q-1\}$. Since $J^* \subseteq \bigcup_{k=1}^q K_{j_k}$, and an interval in J^* can cover points in $(z_{j_i}, z_{j_{i+1}}]$ only if it lies in $K_{j_i} \cup K_{j_{i+1}}$ (since such an interval must contain the anchor point of some minimal interval in J^* that spans this region, and by construction these anchors are exactly z_{j_i} and $z_{j_{i+1}}$), we have

$$\begin{aligned} &c(J_{i+1}^*, X \cap (z_{j_i}, z_{j_{i+1}}]) \\ &= c(J^*, X \cap (z_{j_i}, z_{j_{i+1}}]) \\ &= c(J^* \cap (K_{j_i} \cup K_{j_{i+1}}), X \cap (z_{j_i}, z_{j_{i+1}}]) \\ &= c((J^* \cap K_{j_i}) \cup (J^* \cap K_{j_{i+1}} \setminus K_{j_i}), X \cap (z_{j_i}, z_{j_{i+1}}]) \\ &\leq c(R_i \cup R_{i+1}, X \cap (z_{j_i}, z_{j_{i+1}}]), \end{aligned} \tag{5}$$

where the inequality follows from Lemma 1 and the fact $K_{j_i} \cap K_{j_{i+1}} \subseteq K_{j_{i+1}}^{\leq}$. Hence,

$$\begin{aligned} &|B_{i+1}| \\ &= \left| B(c(J_{i+1}^*, X_{i+1}), j_{i+1}, R_{i+1}) \right| \\ &\leq \left| R_{i+1} \cup B(c(J_{i+1}^*, X_{i+1}) - c(R_i \cup R_{i+1}, X \cap (z_{j_i}, z_{j_{i+1}}]), j_i, R_i) \right| \\ &\leq \left| R_{i+1} \cup B(c(J_{i+1}^*, X_{i+1}) - c(J_{i+1}^*, X \cap (z_{j_i}, z_{j_{i+1}}]), j_i, R_i) \right| \\ &= \left| R_{i+1} \cup B(c(J_{i+1}^*, X_i), j_i, R_i) \right| \\ &= \left| R_{i+1} \cup B(c(J_i^*, X_i), j_i, R_i) \right| \\ &= |R_{i+1} \cup B_i|, \end{aligned}$$

where the first step follows from (2), the second the definition of the dynamic programming table B , the third from Proposition 2(iii) and (5), the fourth the definition of X_i and X_{i+1} , and the fifth holds because no interval in $J_{i+1}^* \setminus J_i^*$ covers any point in X_i .

Since $K_{j_{i+1}}^{\leq} \subseteq R_{i+1} \subseteq B_{i+1}$ (Proposition 2(i)),

$$\begin{aligned} & \left| B_{i+1} \setminus K_{j_{i+1}}^{\leq} \right| \\ & \leq \left| R_{i+1} \cup B_i \setminus K_{j_{i+1}}^{\leq} \right| \\ & \leq \left| (R_{i+1} \setminus K_{j_{i+1}}^{\leq}) \cup (B_i \setminus K_{j_i}^{\leq}) \cup (K_{j_i}^{\leq} \setminus K_{j_{i+1}}^{\leq}) \right| \\ & \leq \left| R_{i+1} \setminus K_{j_{i+1}}^{\leq} \right| + \left| B_i \setminus K_{j_i}^{\leq} \right| + \left| K_{j_i}^{\leq} \setminus K_{j_{i+1}}^{\leq} \right| \\ & \leq (1 + \epsilon) \left| J_{i+1}^* \cap K_{j_{i+1}}^{\geq} \right| + (1 + \epsilon) \left| J_i^* \setminus K_{j_i}^{\leq} \right| + (1 + \epsilon) \left| K_{j_i}^{\leq} \setminus K_{j_{i+1}}^{\leq} \right| \\ & = (1 + \epsilon) \left| J_{i+1}^* \cap K_{j_{i+1}}^{\geq} \right| + (1 + \epsilon) \left| J_i^* \setminus K_{j_{i+1}}^{\leq} \right| \\ & = (1 + \epsilon) \left| J_{i+1}^* \setminus K_{j_{i+1}}^{\leq} \right|. \end{aligned}$$

Finally, $|J| \leq |B_q| = \left| B_q \setminus K_q^{\leq} \right| + \left| K_q^{\leq} \right| \leq (1 + \epsilon) \left| J_q^* \setminus K_q^{\leq} \right| + \left| K_q^{\leq} \right| \leq (1 + \epsilon) \left| J_q^* \right| = (1 + \epsilon) |J^*|$.

By Lemma 1, step 0 can be done in $O(\epsilon n^{1/\epsilon+3})$ time. The number of subproblems is $O(\epsilon p n^{1/\epsilon+3})$, each solvable in $O(\epsilon |X| n^{1/\epsilon+4})$ time (the dominating steps are 6–9), giving total time $O(\epsilon^2 p |X| n^{2/\epsilon+7})$. Steps 10–15 take another $O(\epsilon n^{1/\epsilon+3})$ time. \square

3. Complexity results

We formally define the decision version of *weighted partial interval multicover* as follows.

Input: A collection \mathcal{I} of n intervals $\{I_1, I_2, \dots, I_n\}$, associated with a weight function $w : \mathcal{I} \rightarrow \mathbb{N}$, a set X of points on the real line, associated with a demand function $t : X \rightarrow \mathbb{N}$ and a gain function $g : X \rightarrow \mathbb{N}$, and two nonnegative integers b and p .

Output: Are there a subset $J \subseteq \{1, 2, \dots, n\}$ and a subset $Y \subseteq X$ such that

$$\begin{aligned} \sum_{i \in J} w(I_i) & \leq b, & \text{(budget)} \\ \sum_{x \in Y} g(x) & \geq p, & \text{(gain)} \\ \sum_{i \in J : x \in I_i} w(I_i) & \geq t(x) \quad \text{for all } x \in Y. & \text{(coverage)} \end{aligned}$$

We reduce from the *paired path decomposition* problem defined as follows [8]; see Fig. 5a for an instance.

Input: A path P on $2k$ of vertices, a partition \mathcal{P} of $V(P)$ into k pairs, and a positive integer c .

Output: Does there exist a subset $S \subseteq V(P)$ that contains exactly one vertex from each pair in the partition such that the subgraph $P[S]$ has at most c components?

Let (P, \mathcal{P}, c) be an instance of paired path decomposition. The reduction constructs an instance $(\mathcal{I}, X, w, t, g, b, p)$ of weighted partial interval multicover as follows; see Fig. 5. Label the vertices of the path P in order as u_1, u_2, \dots, u_{2k} . For each $i \in \{1, \dots, 2k\}$, let $p(i) \in \{1, \dots, k\}$ be the index of the pair in \mathcal{P} that contains u_i . Let $n = 2k$ and define:

- Intervals: $\mathcal{I} = \{I_i = [i, i + 1] \mid 1 \leq i \leq 2k\}$. For each $i \in \{1, \dots, 2k\}$, set $w(I_i) = \phi(p(i))$, where $\phi(a) = 2^{3k} + 2^{k+a}$.
- Points: $X = \{2, 3, \dots, 2k\} \cup \{1.5, 2.5, \dots, 2k + 0.5\}$. For integer points $x \in \{2, 3, \dots, 2k\}$, set $t(x) = 2^{3k+1}$ and $g(x) = 1$. For half-integers $y \in \{i + 0.5 \mid 1 \leq i \leq 2k\}$, set $t(y) = 1$ and $g(y) = 2^{3k} + 2^{k+p(i)}$ (matching the weight of its unique incident interval).

Finally, set

$$\begin{aligned} b & = \sum_{i=1}^k (2^{3k} + 2^{k+i}) = k \cdot 2^{3k} + 2^{2k+1} - 2^{k+1}, \\ p & = b + k - c. \end{aligned}$$

Some remarks on the construction are in order. Each integer point $x \in \{2, 3, \dots, 2k\} \subseteq X$ lies in exactly two consecutive intervals, I_{x-1} and I_x ; each half-integer point $y = i + 0.5$ lies only in I_i . The large threshold $t(x)$ for integer points ensures an integer contributes $g(x) = 1$ to the objective only if both incident intervals are selected; half-integers contribute if their unique interval is selected. Thus, a point is fully covered if and only if all intervals containing it are selected. The term $2^{n+p(i)}$ encodes the pair index; the budget b is calibrated so that selecting exactly one endpoint from each pair (together with its pendant) fits the budget bitwise, while selecting both endpoints from any pair would force us to sacrifice one with a higher weight (the lemma below). The parameter c then controls how many integral points need to be fully covered, via the choice of p .

Lemma 2. *Let A_1 and A_2 be two subsets of $\{2^1, 2^2, \dots, 2^n\}$. If $|A_1| = |A_2|$ and $\sum A_1 = \sum A_2$, then $A_1 = A_2$.*

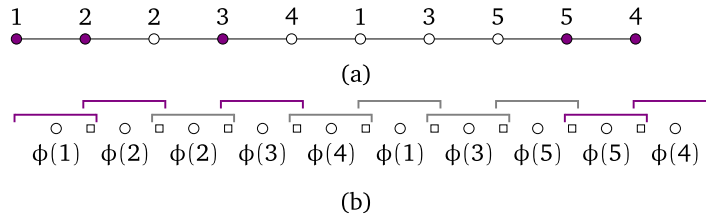


Fig. 5. Illustration for the reduction in [Theorem 2](#). (a) An instance of paired path decomposition with $k = 5$ and $c = 3$, where the numbers are their indices in the partition and the purple points form a solution. (b) The derived instance of the weighted partial interval multicover. All circle points (non-integral) have unit demand and their gains are given below, where $\phi(a) = 2^{3k} + 2^{k+a}$, which is also the weight of the only interval covering it. All square points (integral) have demand 2^{3k+1} and unit gain. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Proof of Theorem 2. The problem is clearly in NP. For NP-hardness, we show that (P, \mathcal{P}, c) is a yes-instance of paired path decomposition if and only if (I, X, w, t, g, b, p) , reduce from paired path decomposition as defined above, is a yes-instance of weighted partial interval multicover.

Necessity. Suppose that S is a solution of (P, \mathcal{P}, c) . Let $J \subseteq \{1, \dots, n\}$ be the index set of vertices in S , so $|J| = k$ and $\{p(j) \mid j \in J\} = \{1, \dots, k\}$. Hence, $\sum_{j \in J} w(I_j) = b$, and the total gain from half-integer points equals b : the gain contributed by covered half-integers matches exactly the total weight of the chosen intervals, since each half-integer is covered by a unique interval whose weight equals its gain. An integer point i gains 1 if and only if both $i - 1$ and i are in J ; i.e., $\{u_{i-1}, u_i\} \subseteq S$. There are at least $k - c$ edges in $P[S]$. Hence, the total gain from integer points equals $k - c$, and (I, X, w, t, g, b, p) is a yes-instance.

Sufficiency. Suppose (J, Y) is a solution of (I, X, w, t, g, b, p) . First, we note that $|J| = k$. If $|J| \geq k + 1$, then $\sum_{i \in J} w(I_i) > 2^{3k}(k + 1) > b$, violating **(budget)**. If $|J| \leq k - 1$, then even counting the gains of all non-integer points that are fully covered and all the $2k - 1$ integer points of unit-gain, we have

$$\sum_{x \in Y} g(x) \leq |J|(2^{3k} + 2^{2k}) + 2k - 1 < k \cdot 2^{3k} < p,$$

violating **(gain)** for sufficiently large k . Hence $|J| = k$, and

$$\sum_{i \in J} w(I_i) = \sum_{i \in J} (2^{3k} + 2^{k+p(i)}) = k \cdot 2^{3k} + 2^{k+1} \left(\sum_{i \in J} 2^{p(i)-1} \right).$$

Next, we show that $\{p(i) \mid i \in J\} = \{1, \dots, k\}$; i.e., J contains exactly one index from each pair in \mathcal{P} . Otherwise, $\sum_{i \in J} 2^{p(i)-1} \neq 0$ by [Lemma 2](#); it cannot be positive because $\sum_{i \in J} w(I_i) \leq b$. Thus, the total gain is at most $(b - 2^{k+1}) + (2k - 1) < p$, a contradiction.

Therefore, J contains exactly one index from each pair, and the total gain from half-integer points equals b . Let $S = \{u_j \mid j \in J\}$. Since the total gain from integer points is at least $p - b = k - c$, the number of edges in $P[S]$ is at least $k - c$. Thus $P[S]$ has at most c components. Hence (P, \mathcal{P}, c) is a yes-instance.

This completes the reduction and the proof. \square

One may attempt to adapt the reduction to the unweighted case as follows. We replace each non-integral point x in X with $g(x)$ points, all with demand $g(x)$,² and each $i \in I$ with $w(I_i)$ intervals with the same endpoints. It is not difficult to check that of each set of duplicate intervals, either all or none are chosen in a feasible solution. However, the produced instance is no longer polynomial in the size of the original instance. The crucial fact behind the reduction is [Lemma 2](#). The reduction could be rescued if we had a polynomial-bounded set with this property, which is similar to the classic distinct subset sums problem of Erdős [9]. However, this is impossible, since any such set must contain an element of magnitude at least $\Omega(2^n/n^{3/2})$ by the following.

Lemma 3. Let A be a set of n positive integers. If there are no different subsets A_1 and A_2 such that $|A_1| = |A_2|$ and $\sum A_1 = \sum A_2$, then

$$\max A = \Omega\left(\frac{2^n}{n^{3/2}}\right).$$

Proof. Let $N = \max A$ and $k = \lfloor \frac{n}{2} \rfloor$. There are $\binom{n}{k}$ different k -element subsets. Since their sums are distinct and all of them are smaller than kN , we must have $\binom{n}{k} \leq kN$. The statement follows from the Stirling formula. \square

4. Concluding remarks

The complexity of partial interval multicover remains open, and the authors hold no unified conjecture on its resolution. A closely related optimization variant asks to cover as many points as possible with a given budget of intervals; Xu et al. [10] gave a 2-approximation for this problem. It is natural to ask whether our approach can yield an improved approximation for this variant.

² In the weighted case, the demand of a non-integral point is immaterial since it is contained in a unique interval.

Our dynamic program draws inspiration from the polynomial-time approximation scheme of Nonner [7] for *densest k -subgraph* on interval graphs. Note that this is equivalent to selecting k intervals to minimize the number of disjoint pairs. The reduction used in Section 3 can be used to show the NP-hardness of the following weighted version of densest k -subgraph on interval graphs.

Input: A graph G , a weight function $w : V(G) \rightarrow \mathbb{N}$, and two positive integers k and m .

Output: Is there a subset $U \subseteq V(G)$ such that $\sum_{v \in U} w(v) \leq k$ and $\sum_{uv \in E(U)} w(u)w(v) \geq m$.

An intriguing open question is whether there is any approximation-preserving reduction between partial interval multicover and densest k -subgraph on interval graphs. Understanding their relationships may shed light on both problems.

CRedit authorship contribution statement

Peng Li: Writing – original draft, Methodology, Investigation, Formal analysis; **Xiangzhi Tu:** Writing – original draft, Investigation, Conceptualization; **Zhao Zhang:** Writing – review & editing, Supervision, Project administration, Methodology, Investigation, Formal analysis; **Yixin Cao:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization.

Data availability

No data was used for the research described in the article.

Declaration of competing interest

None.

Funding

Z.Z. is supported in part by the [National Natural Science Foundation of China](#) (NSFC) under grant [12571342](#). Y.C. is supported in part by the [National Natural Science Foundation of China](#) (NSFC) under grant [62372394](#). P.L. is supported in part by Chongqing Natural Science Foundation Innovation and Development Joint Fund (Municipal Education Commission) under grant CSTB2022NSCQ-LZX0003.

Acknowledgment

The authors want to thank Yingli Ran for the insightful discussions at an early stage of this project.

References

- [1] V.V. Vazirani, *Approximation Algorithms*, Springer, Corrected Second Printing edition, 2003. <https://doi.org/10.1007/978-3-662-04565-7>
- [2] P. Slavik, Improved performance of the greedy algorithm for partial cover, *Inf. Process. Lett.* 64 (5) (1997) 251–254.
- [3] S. Rajagopalan, V.V. Vazirani, Primal-dual RNC approximation algorithms for set cover and covering integer programs, *SIAM J. Comput.* 28 (2) (1998) 525–540. <https://doi.org/10.1137/S0097539793260763>
- [4] Y. Ran, X. Huang, Z. Zhang, D.-Z. Du, Approximation algorithm for minimum power partial multi-coverage in wireless sensor networks, *J. Global Optim.* 80 (2021) 661–677.
- [5] Y. Ran, J. Jin, Z. Zhang, Approximation algorithm for the minimum interval partial multi-cover problem, *Networks* 85 (3) (2025) 288–293. <https://doi.org/10.1002/net.22263>
- [6] L. Golab, H.J. Karloff, F. Korn, A. Saha, D. Srivastava, Sequential dependencies, *Proc. VLDB Endow.* 2 (1) (2009) 574–585. <https://doi.org/10.14778/1687627.1687693>
- [7] T. Nonner, PTAS for densest k -subgraph in interval graphs, *Algorithmica* 74 (1) (2016) 528–539. <https://doi.org/10.1007/s00453-014-9956-7>
- [8] R. Dondi, G. Fertin, S. Vialette, Complexity issues in vertex-colored graph pattern matching, *J. Discrete Algorithms* 9 (1) (2011) 82–99. <https://doi.org/10.1016/J.JDA.2010.09.002>
- [9] P. Erdős, Problems and results in additive number theory, in: *Colloque sur la Théorie des Nombres, Bruxelles, 1955*, pp. 127–137.
- [10] Y. Xu, Y. Ran, Z. Zhang, A $1/2$ -approximation algorithm for maximum interval multi-cover, *Theor. Comput. Sci.* 1049 (2025) 115369. <https://doi.org/10.1016/j.tcs.2025.115369>