



Happy Set Problem on Subclasses of Co-comparability Graphs

Hiroshi Eto¹ · Takehiro Ito¹ · Eiji Miyano² · Akira Suzuki¹ · Yuma Tamura¹

Received: 9 May 2022 / Accepted: 28 November 2022 / Published online: 15 December 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

In this paper, we investigate the complexity of the MAXIMUM HAPPY SET problem on subclasses of co-comparability graphs. For a graph G and its vertex subset S , a vertex $v \in S$ is happy if all v 's neighbors in G are contained in S . Given a graph G and a non-negative integer k , MAXIMUM HAPPY SET is the problem of finding a vertex subset S of G such that $|S| = k$ and the number of happy vertices in S is maximized. In this paper, we first show that MAXIMUM HAPPY SET is NP-hard even for co-bipartite graphs. We then give an algorithm for n -vertex interval graphs whose running time is $O(n^2 + k^3n)$; this improves the best known running time $O(kn^8)$ for interval graphs. We also design algorithms for n -vertex permutation graphs and d -trapezoid graphs which run in $O(n^2 + k^3n)$ and $O(n^2 + d^2(k+1)^{3d}n)$ time, respectively. These algorithmic results provide a nice contrast to the fact that MAXIMUM HAPPY SET remains NP-hard for chordal graphs, comparability graphs, and co-comparability graphs.

Keywords Graph algorithm · Co-comparability graphs · Interval graphs · Permutation graphs · d -trapezoid graphs · Happy set problem

A preliminary version has appeared in proceedings of the 16th International Conference and Workshop on Algorithms and Computation (WALCOM 2022) [1].

✉ Yuma Tamura
tamura@tohoku.ac.jp

Hiroshi Eto
eto@ai.kyutech.ac.jp

Takehiro Ito
takehiro@tohoku.ac.jp

Eiji Miyano
miyano@ai.kyutech.ac.jp

Akira Suzuki
akira@tohoku.ac.jp

- ¹ Graduate School of Information Sciences, Tohoku University, Sendai, Miyagi 980-8579, Japan
- ² School of Computer Science and Systems Engineering, Kyushu Institute of Technology, Iizuka, Fukuoka 820-8502, Japan

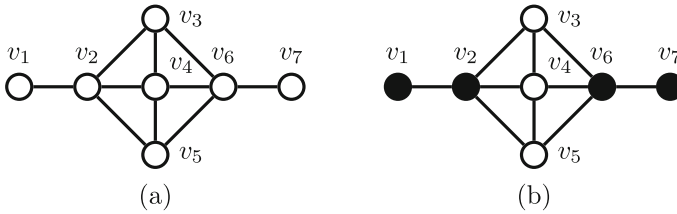


Fig. 1 **a** A graph G , and **b** an optimal solution $S = \{v_1, v_2, v_6, v_7\}$ for G and $k = 4$, where only v_1 and v_7 are happy vertices

1 Introduction

Easley and Kleinberg [2] said that *homophily* is one of the most basic notions governing the structure of social networks. Homophily is the principle that we are likely to associate with people who are similar in characteristics, such as their ages, their occupations, and their interests. Motivated from homophily of social networks, Zhang and Li [3] formulated two graph coloring problems, and recently Asahiro et al. [4] introduced another formulation on graphs. In this paper, we study the latter formulation, defined as follows.

For a graph $G = (V, E)$ and a subset $S \subseteq V$, a vertex $v \in S$ is *happy* if all its neighbors in G are contained in S . Given an undirected graph $G = (V, E)$ and a non-negative integer k , MAXIMUM HAPPY SET is the problem of finding a subset $S \subseteq V$ such that $|S| = k$ and the number of happy vertices in S is maximized. For example, the set $S = \{v_1, v_2, v_6, v_7\}$ in Fig. 1b is an optimal solution to MAXIMUM HAPPY SET for the graph G in Fig. 1a and $k = 4$, where only two vertices v_1 and v_7 are happy.

1.1 Known Results

Although MAXIMUM HAPPY SET was proposed recently,¹ it has been already studied from various viewpoints such as polynomial-time solvability, approximability, and fixed-parameter tractability.

Polynomial-Time Solvability: MAXIMUM HAPPY SET is NP-hard even for bipartite graphs [5], cubic graphs [5], and split graphs [4]. On the other hand, the problem is solvable in $O(k^2n)$ time for block graphs [5], and solvable in $O(kn^8)$ time for interval graphs [5], where n is the number of vertices in a graph.

Approximability: MAXIMUM HAPPY SET admits a polynomial-time approximation algorithm whose approximation ratio depends on the maximum degree of a graph [5].

Fixed-Parameter Tractability: MAXIMUM HAPPY SET is W[1]-hard when parameterized by k even on split graphs [4], and hence it is very unlikely that the problem admits a fixed-parameter algorithm even when restricted to split graphs and parameterized by

¹ We note that the graph coloring problem introduced by Zhang and Li [3] is called a similar name, MAXIMUM HAPPY VERTICES, but it is a different problem from ours.

k . On the other hand, the problem admits fixed-parameter algorithms when parameterized by graph structural parameters such as tree-width, clique-width, neighborhood diversity, and twin-cover number of a graph [4].

1.2 Our Contributions

In this paper, we further investigate the polynomial-time solvability of MAXIMUM HAPPY SET, by focusing on subclasses of co-comparability graphs. In particular, we consider co-bipartite graphs, interval graphs, permutation graphs and d -trapezoid graphs. See the relationship of graph classes illustrated in Fig. 2.

We first show that MAXIMUM HAPPY SET is NP-hard even for co-bipartite graphs. As far as we know, this is the first intractability result of MAXIMUM HAPPY SET on subclasses of co-comparability graphs. We thus need to focus on other subclasses of co-comparability graphs, in order to seek polynomial-time solvable cases, as below.

We then give a polynomial-time algorithm for interval graphs. Recall that the polynomial-time solvability for interval graphs is already known [5]. However, our algorithm runs in $O(n^2 + k^3n)$ time for n -vertex interval graphs, which improves the best known running time $O(kn^8)$ [5].

We finally extend our algorithmic techniques for interval graphs to permutation graphs and d -trapezoid graphs, and give algorithms for n -vertex permutation graphs and d -trapezoid graphs which run in $O(n^2 + k^3n)$ and $O(n^2 + d^2(k + 1)^{3d}n)$ time, respectively. These are new polynomial-time solvable cases, and give a nice contrast to the known fact that MAXIMUM HAPPY SET is NP-hard for comparability graphs and co-comparability graphs (see also Fig. 2). We note that if k and d are fixed constants, then all of our algorithms in this paper run in $O(n^2)$ time.

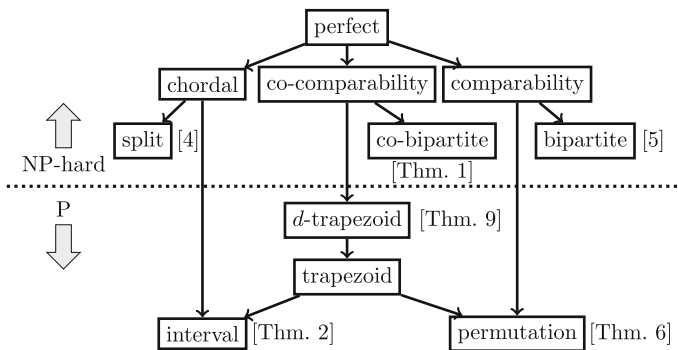


Fig. 2 Our results together with known ones for MAXIMUM HAPPY SET on subclasses of perfect graphs. Each arrow represents the inclusion relationship between graph classes: $A \rightarrow B$ means that the graph class B is a subclass of the graph class A

1.2.1 Technical highlight

Our polynomial-time algorithms for interval graphs, permutation graphs and d -trapezoid graphs employ basically the same technique, that is, a dynamic programming approach based on graph representation models. Details and formal definitions will be given later, but we here explain the key point. Given an n -vertex graph $G = (V, E)$, we define a subgraph $G_i = (V_i, E_i)$ for each integer $i = 1, 2, \dots, n$, depending on a representation model for G . Then, we wish to compute a partial solution $S_i = S^* \cap V_i$ for each G_i , where S^* is an optimal solution of G . Note that S_i is not always optimal for G_i , and hence it is not enough to compute an optimal solution of G_i . The key of our algorithms is that partial solutions S_i of G_i can be characterized by only two vertices that are *not* contained in S^* if G is an interval graph or a permutation graph, and characterized by $2d$ vertices that are *not* contained in S^* if G is a d -trapezoid graph. This efficient characterization of partial solutions leads to improving the running time for interval graphs.

1.3 Contrasts to Related Results

Our initial motivation was to develop a polynomial-time algorithm for MAXIMUM HAPPY SET on co-comparability graphs, because it is known that several classical problems are tractable for co-comparability graphs even if they are NP-hard on perfect graphs (see again Fig. 2). Such examples include MINIMUM DOMINATING SET [6], HAMILTONIAN CYCLE [7], and MINIMUM FEEDBACK VERTEX SET [8]. Our result of NP-hardness for co-comparability graphs gives an interesting contrast to these complexity examples.

The DENSEST k -SUBGRAPH problem [9], which has been studied for more than two decades in the field of graph theory, can be seen as an edge variant of MAXIMUM HAPPY SET: given an undirected graph $G = (V, E)$ and a non-negative integer k , the task of the problem is to find a vertex subset $S \subseteq V$ of size exactly k such that the number of edges whose both endpoints are contained in S is maximized. Interestingly, the complexity of DENSEST k -SUBGRAPH remains open for interval graphs, permutation graphs, and planar graphs. Although results on MAXIMUM HAPPY SET cannot be converted directly to DENSEST k -SUBGRAPH, our complexity results in this paper may give new insights to DENSEST k -SUBGRAPH.

2 Preliminaries

Let $G = (V, E)$ be a graph; we denote by $V(G)$ and $E(G)$ the vertex set and the edge set of G , respectively. We assume that all graphs in this paper are simple, undirected, and unweighted. For a vertex v of G , we denote by $N_G(v)$ and $N_G[v]$ the open and closed neighborhood of v in G , respectively, that is, $N_G(v) = \{w \in V(G) : vw \in E(G)\}$ and $N_G[v] = N_G(v) \cup \{v\}$. For a vertex subset $V' \subseteq V$, we denote by $G - V'$ the subgraph of G obtained by deleting all the vertices in V' and their incident edges. We shall often write $G - v$ instead of $G - \{v\}$ for a vertex $v \in V$.

For a graph $G = (V, E)$ and its vertex subset $S \subseteq V$, we say that a vertex $v \in V$ is *happy with respect to S* if $N_G[v] \subseteq S$; otherwise v is *unhappy with respect to S*. We denote by $H(G; S)$ the set of happy vertices with respect to S . We note that $H(G; \emptyset) = \emptyset$. Given a graph $G = (V, E)$ and a non-negative integer k , MAXIMUM HAPPY SET is the problem of finding a vertex subset $S \subseteq V$ such that $|S| = k$ and the size of $H(G; S)$ is maximized. For simplicity, our algorithms in this paper only compute the maximum value of $|H(G; S)|$. However, one can easily modify the algorithms so that they find an actual subset S in the same time complexity.

3 NP-Hardness for Co-bipartite Graphs

A graph is *co-bipartite* if it is the complement of a bipartite graph. In other words, a co-bipartite graph is a graph whose vertex set can be partitioned into two cliques. In this section, we give the following hardness result.

Theorem 1 MAXIMUM HAPPY SET is NP-hard for co-bipartite graphs.

Proof We give a polynomial-time reduction from MAXIMUM HAPPY SET on general graphs to MAXIMUM HAPPY SET on co-bipartite graphs. Let (G, k) be an instance of MAXIMUM HAPPY SET on general graphs. We construct an instance (G', k') of MAXIMUM HAPPY SET on co-bipartite graphs from (G, k) as follows. Let $n = |V(G)|$ and $V(G) = \{v_1, v_2, \dots, v_n\}$. The graph G' consists of disjoint two cliques whose vertex subsets are $A = \{a_1, a_2, \dots, a_n\}$ and $B = \{b_1, b_2, \dots, b_{n+k+1}\}$. In addition, G' has an edge $a_i b_j$ if and only if $i = j$ or $v_i v_j \in E(G)$ for each $i, j \in \{1, 2, \dots, n\}$. This completes the construction of G' , and we let $k' = n + k$. Clearly, this reduction can be done in polynomial time.

We show that for any integer $\ell \geq 0$, there exists a subset $S \subseteq V(G)$ such that $|S| = k$ and $|H(G; S)| \geq \ell$ if and only if there exists a subset $S' \subseteq V(G')$ such that $|S'| = k'$ and $|H(G'; S')| \geq \ell$.

We first prove the necessity. Assume that there exists a subset $S \subseteq V(G)$ such that $|S| = k$ and $|H(G; S)| \geq \ell$. Let $S' = A \cup \{b_j \mid v_j \in S\}$. Clearly, $|S'| = n + k = k'$. To show that $|H(G'; S')| \geq \ell$, we indicate that for any happy vertex $v_i \in H(G; S)$, the corresponding vertex $a_i \in A$ of G' is also happy with respect to S' . Obviously, $N_{G'}[a_i] \cap A \subseteq S'$. Moreover, since v_i is happy with respect to S on G , any vertex $v_j \in N_G[v_i]$ is in S . By the construction of S' , we have $b_j \in S'$ that is adjacent to a_i on G' . Therefore, $N_{G'}[a_i] \subseteq S'$, that is, $a_i \in H(G'; S')$. This implies that $|H(G'; S')| \geq |H(G; S)| \geq \ell$.

Conversely, we prove the sufficiency. Assume that there exists a subset $S' \subseteq V(G')$ such that $|S'| = k' = n + k$ and $|H(G'; S')| \geq \ell$. Since $|S'| = n + k$ and B forms a clique of $n + k + 1$ vertices, at least one vertex $b_j \in B$ is not in S' and hence all vertices in B are unhappy with respect to S' . This means that any happy vertex with respect to S' is in A . In the remainder of this proof, we assume that $A \subseteq S'$; otherwise $|H(G'; S')| = 0$. In addition, if there exists a vertex $b_j \in S'$ for an integer j with $n + 1 \leq j \leq n + k + 1$, then we remove b_j from S' and arbitrarily add a vertex $b_{j'} \notin S'$ into S' for an integer $j' \in \{1, \dots, n\}$. We denote by S'' a set obtained by repeating this operation until $S'' \cap \{b_{n+1}, \dots, b_{n+k+1}\} = \emptyset$. Since no vertex in A is

adjacent to b_j for an integer $j \in \{n + 1, \dots, n + k + 1\}$ from the construction of G' , $H(G'; S') \subseteq H(G'; S'')$ holds. Let $S = \{v_j \mid j \in \{1, 2, \dots, n\} \text{ and } b_j \in S''\}$. Notice that, since S'' contains all vertices in A , we have $|S| = |S''| - |A| = k$. We now show that for any happy vertex $a_i \in H(G'; S'')$, the corresponding vertex $v_i \in V(G)$ is also happy with respect to S . Since a_i is happy, any vertex $b_j \in N_{G'}[a_i] \cap B$ is in S'' and hence $v_j \in S$. This implies that $N_G[v_i] \subseteq S$, that is, $v_i \in H(G; S)$. Therefore, $|H(G; S)| \geq |H(G'; S'')| \geq |H(G'; S')| \geq \ell$. This completes the proof. \square

4 Polynomial-Time Algorithm for Interval Graphs

A graph $G = (V, E)$ with vertices v_1, v_2, \dots, v_n is called an *interval graph* if, for some set $\mathcal{I} = \{I_1, I_2, \dots, I_n\}$ of intervals on the real line, there is a one-to-one correspondence between V and \mathcal{I} such that $v_i v_j \in E$ if and only if I_i intersects I_j for each $i, j \in \{1, 2, \dots, n\}$. Such a set \mathcal{I} of intervals is called an *interval representation* of G . For instance, Fig. 3 shows the interval representation and its corresponding interval graph. In this section, we give a polynomial-time algorithm for MAXIMUM HAPPY SET on interval graphs.

Theorem 2 *Given an n -vertex interval graph G and a non-negative integer k , MAXIMUM HAPPY SET is solvable in $O(n^2 + k^3 n)$ time.*

Before the detailed description of our algorithm, we give the following simple but useful lemma.

Lemma 3 *Let $G = (V, E)$ be a graph, and let V' and S be subsets of V . Then, it holds that $H(G; S) \setminus V' \subseteq H(G - V'; S \setminus V')$. Moreover, if $V' \subseteq S$, then it holds that $H(G; S) \setminus V' = H(G - V'; S \setminus V')$.*

Proof We first prove the former statement. The case of $H(G; S) \setminus V' = \emptyset$ is trivial. Suppose that $H(G; S) \setminus V' \neq \emptyset$ and consider a vertex $v \in H(G; S) \setminus V'$. Since $N_G[v] \subseteq S$ holds from the definition, $N_{G-V'}[v] \subseteq S \setminus V'$. This implies that $v \in H(G - V'; S \setminus V')$, that is, $H(G; S) \setminus V' \subseteq H(G - V'; S \setminus V')$.

To prove the latter statement, we also show that $H(G - V'; S \setminus V') \subseteq H(G; S) \setminus V'$ if $V' \subseteq S$. The case of $H(G - V'; S \setminus V') = \emptyset$ is trivial and hence assume that there is a vertex $v \in H(G - V'; S \setminus V')$. Since $N_{G-V'}[v] \subseteq S \setminus V'$ and $V' \subseteq S$, we have $N_G[v] \subseteq S$, that is, $v \in H(G; S)$. It follows from $v \notin V'$ that $v \in H(G; S) \setminus V'$ and thus $H(G - V'; S \setminus V') \subseteq H(G; S) \setminus V'$. \square

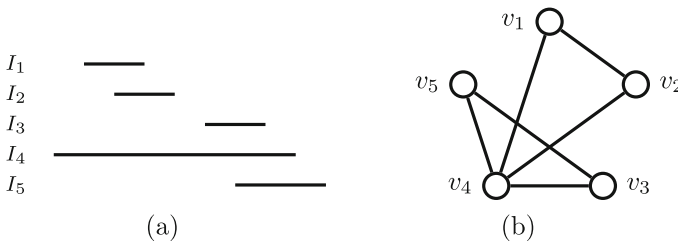
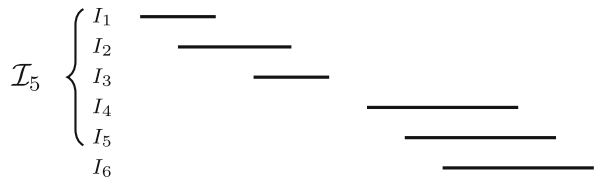


Fig. 3 **a** The interval representation and **b** its corresponding interval graph

Fig. 4 If $k = 2$, the optimal solution of \mathcal{I}_5 is $\{I_4, I_5\}$, while the optimal solution \mathcal{S}^* of \mathcal{I}_6 and the partial solution $\mathcal{S}_5 = \mathcal{S}^* \cap \mathcal{I}_5$ are $\{I_1, I_2\}$ or $\{I_2, I_3\}$



Notice that Lemma 3 is applicable to general graphs as well as interval graphs.

To explain our algorithm, we need several assumptions and notations. Given an interval graph G , an interval representation \mathcal{I} of G can be constructed in linear time [10–12]. For a subset $\mathcal{S} \subseteq \mathcal{I}$, we say that an interval $I \in \mathcal{I}$ is *happy with respect to* \mathcal{S} if $I \cap I' \neq \emptyset$ for every interval $I' \in \mathcal{S}$. It is easy to see that for a subset $\mathcal{S} \subseteq \mathcal{I}$ and its corresponding vertex subset $S \subseteq V(G)$, an interval $I \in \mathcal{I}$ is happy with respect to \mathcal{S} if and only if its corresponding vertex $v \in V(G)$ is happy with respect to S . Therefore, we consider MAXIMUM HAPPY SET on an interval representation \mathcal{I} instead of its original interval graph G . Let $H(\mathcal{I}; \mathcal{S})$ be a set of happy intervals with respect to $\mathcal{S} \subseteq \mathcal{I}$ and our task is to find \mathcal{S} such that $|H(\mathcal{I}; \mathcal{S})|$ is maximized.

We denote by $\text{left}(I_i)$ and $\text{right}(I_i)$ the left endpoint and the right endpoint of an interval $I_i \in \mathcal{I}$, respectively. One can easily transform an interval representation \mathcal{I} of G into another one without changing G so that distinct integers between 1 and $2n$ are assigned to the endpoints $\text{left}(I_i)$ and $\text{right}(I_i)$ of every interval I_i . Moreover, we assume that intervals of \mathcal{I} are sorted in increasing order of the right endpoints, that is, $\text{right}(I_i) < \text{right}(I_j)$ for any integers i, j with $1 \leq i < j \leq n$. We then add dummy intervals I_0 and I_{n+1} with $\text{left}(I_0) = -1, \text{right}(I_0) = 0, \text{left}(I_{n+1}) = 2n + 1,$ and $\text{right}(I_{n+1}) = 2n + 2$ into \mathcal{I} . The addition of I_0 and I_{n+1} is not essential for proving Theorem 2, but this simplifies the description of our algorithm. In the remainder of this section, we assume that \mathcal{I} has I_0 and I_{n+1} . For an integer $i \in \{0, \dots, n + 1\}$, we let $\mathcal{I}_i = \{I_0, I_1, \dots, I_i\}$ and $\overline{\mathcal{I}}_i = \mathcal{I} \setminus \mathcal{I}_i$.

We describe the idea of our algorithm. Let \mathcal{S}^* be a subset of $\mathcal{I} \setminus \{I_0, I_{n+1}\}$ such that $|H(\mathcal{I}; \mathcal{S}^*)|$ is maximized among all subsets of $\mathcal{I} \setminus \{I_0, I_{n+1}\}$ of size k . Since both I_0 and I_{n+1} have no intersection with other intervals of \mathcal{I} , \mathcal{S}^* is also the optimal solution of the original interval representation that has no dummy intervals I_0 and I_{n+1} . In order to find \mathcal{S}^* , we wish to compute a partial solution $\mathcal{S}_i = \mathcal{S}^* \cap \mathcal{I}_i$ for each $i \in \{0, 1, \dots, n + 1\}$ by means of dynamic programming. Since $\mathcal{I} = \mathcal{I}_{n+1}$, we have $\mathcal{S}^* = \mathcal{S}_{n+1}$. Notice that, however, a partial solution \mathcal{S}_i is not always equal to an optimal solution of \mathcal{I}_i (see Fig. 4 as an example). This indicates that it is not enough to find only an optimal solution of \mathcal{I}_i as a partial solution \mathcal{S}_i . To correctly compute \mathcal{S}_i for each $i \in \{0, \dots, n + 1\}$, we guess integers $r, \ell,$ and k' that satisfy the following three conditions for \mathcal{S}^* :

- the interval I_r has the largest right endpoint among all intervals in $\mathcal{I}_i \setminus \mathcal{S}^*$, that is, $I_r \notin \mathcal{S}^*$ and $\text{right}(I_{i'}) \leq \text{right}(I_r)$ for any $I_{i'} \in \mathcal{I}_i \setminus \mathcal{S}^*$;
- the interval I_ℓ has the smallest left endpoint among all intervals in $\overline{\mathcal{I}}_i \setminus \mathcal{S}^*$, that is, $I_\ell \notin \mathcal{S}^*$ and $\text{left}(I_\ell) \leq \text{left}(I_{i'})$ for any $I_{i'} \in \overline{\mathcal{I}}_i \setminus \mathcal{S}^*$; and
- $|\mathcal{S}_i| = k'$.

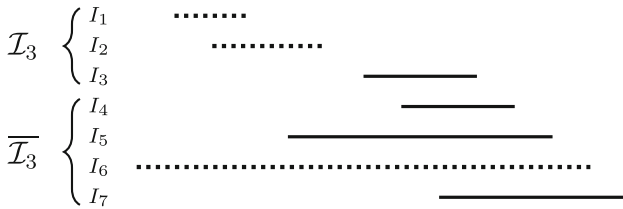


Fig. 5 For the above interval representation $\mathcal{I} = \{I_1, I_2, \dots, I_7\}$, if $k = 3$, the optimal solution \mathcal{S}^* is $\{I_1, I_2, I_6\}$, which is drawn with dotted line segments. Then, for $i = 3$, the interval I_3 (resp. I_5) has the largest right (resp. the smallest left) endpoint among all intervals in $\mathcal{I}_i \setminus \mathcal{S}^*$ (resp. $\overline{\mathcal{I}}_i \setminus \mathcal{S}^*$), and $|\mathcal{S}'_i| = |\mathcal{S}^* \cap \mathcal{I}_i| = 2$. Therefore, a quadruple $(3, 3, 5, 2)$ is compatible with \mathcal{S}^*

We say that a quadruple (i, r, ℓ, k') of integers is *compatible* with \mathcal{S}^* if i, r, ℓ , and k' satisfy the above three conditions (see Fig. 5). Clearly, if (i, r, ℓ, k') is compatible with \mathcal{S}^* , then $0 \leq r \leq i < \ell \leq n + 1$ and $0 \leq k' \leq k$ hold. For integers i, r , and ℓ , let $\overrightarrow{\mathcal{I}}_r = \{I_{i'} \in \mathcal{I}_i \mid \text{right}(I_r) < \text{right}(I_{i'})\}$ and let $\mathcal{I}_{i,\ell}$ be a shorthand for $\mathcal{I}_i \cup \{I_\ell\}$ (See again Fig. 5 as an example. If $i = 3$, we have $\overrightarrow{\mathcal{I}}_r = \emptyset$ because $r = 3$, and if $i = 6$, we have $\overrightarrow{\mathcal{I}}_r = \{I_6\}$ because $r = 5$). We then obtain the following lemma.

Lemma 4 *For an interval representation \mathcal{I} and an integer k , let \mathcal{S}^* be an optimal solution of $\mathcal{I} \setminus \{I_0, I_{n+1}\}$ such that $|\mathcal{S}^*| = k$. Suppose that a quadruple (i, r, ℓ, k') of integers is compatible with \mathcal{S}^* and $\mathcal{S}'_i \subseteq \mathcal{I}_{i,\ell} \setminus \{I_0, I_r, I_\ell\}$ is an optimal solution of $\mathcal{I}_{i,\ell}$ such that $\overrightarrow{\mathcal{I}}_r \subseteq \mathcal{S}'_i$ and $|\mathcal{S}'_i| = k'$. Then, there is an optimal solution \mathcal{S}^* of $\mathcal{I} \setminus \{I_0, I_{n+1}\}$ such that $\mathcal{S}'_i = \mathcal{S}^* \cap \mathcal{I}_i$.*

Proof Let $\mathcal{S}_i = \mathcal{S}^* \cap \mathcal{I}_i$ and $\mathcal{S}^* = (\mathcal{S}^* \setminus \mathcal{S}_i) \cup \mathcal{S}'_i$. Note that $\overline{\mathcal{I}}_i \setminus \mathcal{S}^* = \overline{\mathcal{I}}_i \setminus \mathcal{S}^*$ holds. We then show that the following three inequalities:

- (I) $|H(\mathcal{I}; \mathcal{S}^*) \cap \mathcal{I}_{i,\ell}| \leq |H(\mathcal{I}_{i,\ell}; \mathcal{S}_i)|$;
- (II) $|H(\mathcal{I}_{i,\ell}; \mathcal{S}'_i)| \leq |H(\mathcal{I}; \mathcal{S}^*) \cap \mathcal{I}_{i,\ell}|$; and
- (III) $|H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}_{i,\ell}| \leq |H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}_{i,\ell}|$.

Since $|H(\mathcal{I}_{i,\ell}; \mathcal{S}_i)| \leq |H(\mathcal{I}_{i,\ell}; \mathcal{S}'_i)|$ from the maximality of \mathcal{S}'_i , we have $|H(\mathcal{I}; \mathcal{S}^*)|$ from the inequalities (I)–(III) as follows:

$$\begin{aligned} |H(\mathcal{I}; \mathcal{S}^*)| &= |H(\mathcal{I}; \mathcal{S}^*) \cap \mathcal{I}_{i,\ell}| + |H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}_{i,\ell}| \\ &\leq |H(\mathcal{I}_{i,\ell}; \mathcal{S}_i)| + |H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}_{i,\ell}| \\ &\leq |H(\mathcal{I}_{i,\ell}; \mathcal{S}'_i)| + |H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}_{i,\ell}| \\ &\leq |H(\mathcal{I}; \mathcal{S}^*) \cap \mathcal{I}_{i,\ell}| + |H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}_{i,\ell}| \\ &= |H(\mathcal{I}; \mathcal{S}^*)|. \end{aligned}$$

Namely, \mathcal{S}^* is also the optimal solution of $\mathcal{I} \setminus \{I_0, I_{n+1}\}$ from the maximality of $|H(\mathcal{I}; \mathcal{S}^*)|$.

We first show the inequality (I). Consider an interval representation \mathcal{I} of an interval graph G on Lemma 3, and set $\mathcal{I}' = \mathcal{I} \setminus \mathcal{I}_{i,\ell}$ and $\mathcal{S} = \mathcal{S}^*$. Then, we have

$$H(\mathcal{I}; \mathcal{S}^*) \cap \mathcal{I}_{i,\ell} = H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}'$$

$$\begin{aligned} &\subseteq H(\mathcal{I} \setminus \mathcal{I}'; \mathcal{S}^* \setminus \mathcal{I}') \\ &= H(\mathcal{I}_{i,\ell}; \mathcal{S}^* \cap \mathcal{I}_{i,\ell}). \end{aligned}$$

Since $\mathcal{S}_i = \mathcal{S}^* \cap \mathcal{I}_i$ and $I_\ell \notin \mathcal{S}^*$, we have $|H(\mathcal{I}; \mathcal{S}^*) \cap \mathcal{I}_{i,\ell}| \leq |H(\mathcal{I}_{i,\ell}; \mathcal{S}^* \cap \mathcal{I}_{i,\ell})| = |H(\mathcal{I}_{i,\ell}; \mathcal{S}_i)|$.

We next show the inequality (II). Consider an interval $I_j \in H(\mathcal{I}_{i,\ell}; \mathcal{S}'_i)$. Then, $I_j \in \mathcal{S}'_i$ and I_j intersects with none of intervals in $\mathcal{I}_{i,\ell} \setminus \mathcal{S}'_i = \mathcal{I}_{i,\ell} \setminus (\mathcal{S}^* \cap \mathcal{I}_i) = \mathcal{I}_{i,\ell} \setminus \mathcal{S}^*$. In particular, $\text{right}(I_j) < \text{left}(I_\ell)$ holds. For any interval $I_{j'} \in \overline{\mathcal{I}}_i \setminus \mathcal{S}^*$, equivalently $I_{j'} \in \overline{\mathcal{I}}_i \setminus \mathcal{S}^*$, we also have $\text{left}(I_\ell) \leq \text{left}(I_{j'})$ from the definition of I_ℓ . Thus, $\text{right}(I_j) < \text{left}(I_{j'})$ holds and I_j has no intersection with $I_{j'}$. As a consequence, I_j intersects with none of intervals in $\overline{\mathcal{I}} \setminus \mathcal{S}^*$, that is, $I_j \in H(\mathcal{I}; \mathcal{S}^*) \cap \mathcal{I}_{i,\ell}$. This means that $|H(\mathcal{I}_{i,\ell}; \mathcal{S}'_i)| \leq |H(\mathcal{I}; \mathcal{S}^*) \cap \mathcal{I}_{i,\ell}|$.

Finally, we show the inequality (III). Consider an interval $I_j \in H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}_{i,\ell}$. Then, $I_j \in \mathcal{S}^* \setminus \mathcal{I}_{i,\ell}$ and I_j intersects with none of intervals in $\overline{\mathcal{I}} \setminus \mathcal{S}^*$. In particular, $\text{right}(I_r) < \text{left}(I_j)$ holds. Recall that $\overline{\mathcal{I}}_i \setminus \mathcal{S}^* = \overline{\mathcal{I}}_i \setminus \mathcal{S}^*$. This means that $I_j \in \mathcal{S}^* \setminus \mathcal{I}_{i,\ell}$ and I_j intersects with none of intervals in $\overline{\mathcal{I}}_i \setminus \mathcal{S}^*$. Moreover, from the assumptions that $\overrightarrow{\mathcal{I}}_r \subseteq \mathcal{S}'_i$ and $\mathcal{S}^* = (\mathcal{S}^* \setminus \mathcal{S}_i) \cup \mathcal{S}'_i$, it holds that $\text{right}(I_{j'}) \leq \text{right}(I_r)$ for any interval $I_{j'}$ in $\overline{\mathcal{I}}_i \setminus \mathcal{S}'_i = \overline{\mathcal{I}}_i \setminus \mathcal{S}^*$. Combined with $\text{right}(I_r) < \text{left}(I_j)$, we have $\text{right}(I_{j'}) < \text{left}(I_j)$, that is, I_j has no intersection with $I_{j'} \in \overline{\mathcal{I}}_i \setminus \mathcal{S}^*$. Therefore, I_j intersects with none of intervals in $\overline{\mathcal{I}} \setminus \mathcal{S}^*$ and thus $I_j \in H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}_{i,\ell}$. We conclude that $|H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}_{i,\ell}| \leq |H(\mathcal{I}; \mathcal{S}^*) \setminus \mathcal{I}_{i,\ell}|$. \square

Lemma 4 suggests that, in order to compute a partial solution of \mathcal{I}_i for each $i \in \{0, \dots, n\}$, it suffices to guess a quadruple (i, r, ℓ, k') of integers that is compatible with \mathcal{S}^* and find \mathcal{S} such that $|H(\mathcal{I}_{i,\ell}; \mathcal{S})|$ is the maximum among all subsets $\mathcal{S} \subseteq \mathcal{I}_{i,\ell} \setminus \{I_0, I_r, I_\ell\}$ that satisfies $\overrightarrow{\mathcal{I}}_r \subseteq \mathcal{S}$ and $|\mathcal{S}| = k'$.

4.1 Algorithm

Our algorithm employs the following main function $f(\mathcal{I}_{i,\ell}; r, k')$ and the subfunction $f_{\max}(\mathcal{I}_{i,\ell}; k')$, where i, r, ℓ , and k' are integers such that $0 \leq i < \ell \leq n+1$, $\max\{0, i - k'\} \leq r \leq i$, and $0 \leq k' \leq k$;

- $f(\mathcal{I}_{i,\ell}; r, k')$ returns the maximum of $|H(\mathcal{I}_{i,\ell}; \mathcal{S})|$ among all subsets $\mathcal{S} \subseteq \mathcal{I}_{i,\ell} \setminus \{I_0, I_r, I_\ell\}$ such that $\overrightarrow{\mathcal{I}}_r \subseteq \mathcal{S}$ and $|\mathcal{S}| = k'$;
- $f_{\max}(\mathcal{I}_{i,\ell}; k')$ returns the maximum of $|H(\mathcal{I}_{i,\ell}; \mathcal{S})|$ among all subsets $\mathcal{S} \subseteq \mathcal{I}_{i,\ell} \setminus \{I_0, I_\ell\}$ such that $|\mathcal{S}| = k'$.

We let $f(\mathcal{I}_{i,\ell}; r, k') = -\infty$ and $f_{\max}(\mathcal{I}_{i,\ell}; k') = -\infty$ if there exists no subset \mathcal{S} that satisfies all the prescribed conditions for f and f_{\max} , respectively. We remark that the integer r must be at least $i - k'$; otherwise it implies $|\mathcal{S}| > k'$ and violates the condition $|\mathcal{S}| = k'$. We will compute values $f(\mathcal{I}_{i,\ell}; r, k')$ and $f_{\max}(\mathcal{I}_{i,\ell}; k')$ by means of dynamic programming in accordance with the lexicographical order of a quadruple (i, r, ℓ, k') . We lastly obtain the value $f_{\max}(\mathcal{I}_{n,n+1}; k)$, which is the maximum size of $H(\mathcal{I}; \mathcal{S})$ such that $\mathcal{S} \subseteq \mathcal{I} \setminus \{I_0, I_{n+1}\}$ and $|\mathcal{S}| = k$.

For each triple (i, ℓ, k') of integers, it is easy to see that $f_{\max}(\mathcal{I}_{i,\ell};k')$ is obtained as follows:

$$f_{\max}(\mathcal{I}_{i,\ell};k') = \max_{\max\{0,i-k'\} \leq r \leq i} f(\mathcal{I}_{i,\ell};r,k').$$

We compute $f(\mathcal{I}_{i,\ell};r,k')$ by dividing two cases (i) $r < i$ and (ii) $r = i$.

Case (i): $r < i$

If $i = 0$, then we have $f(\mathcal{I}_{i,\ell};r,k') = -\infty$ because there exists no integer r such that $\max\{0, i - k'\} \leq r < i$. If $k' = 0$, then we also have $f(\mathcal{I}_{i,\ell};r,k') = -\infty$ for any i, ℓ , and r because there is no subset \mathcal{S} such that $I_i \in \mathcal{S}$ and $|\mathcal{S}| = 0$.

Suppose that $i > 0$ and $k' > 0$. Let \mathcal{S} be a subset of $\mathcal{I}_{i,\ell} \setminus \{I_0, I_r, I_\ell\}$ such that $\vec{\mathcal{I}}_r \subseteq \mathcal{S}$, $|\mathcal{S}| = k'$, and $|H(\mathcal{I}_{i,\ell};\mathcal{S})|$ is maximized. Especially, $I_i \in \mathcal{S}$ in this case. From Lemma 3, it holds that $H(\mathcal{I}_{i,\ell};\mathcal{S}) \setminus \{I_i\} = H(\mathcal{I}_{i-1,\ell};\mathcal{S} \setminus \{I_i\})$. We thus compute $f(\mathcal{I}_{i,\ell};r,k')$ from $f(\mathcal{I}_{i-1,\ell};r,k' - 1)$ by deciding whether the interval I_i is happy with respect to \mathcal{S} . If I_i intersects with the interval $I_r \notin \mathcal{S}$ or $I_\ell \notin \mathcal{S}$, then I_i is unhappy. Conversely, assume that I_i intersects with neither I_r nor I_ℓ . Let $\mathcal{I}'_{i,\ell}$ be a set of intervals of $\mathcal{I}_{i,\ell}$ that intersect with I_i . Since the intervals of $\mathcal{I}_{i,\ell}$ are sorted in increasing order of the right endpoints, we have $\mathcal{I}'_{i,\ell} \subseteq \vec{\mathcal{I}}_r$ from the definition of I_r . Combined with $\vec{\mathcal{I}}_r \subseteq \mathcal{S}$, we have $\mathcal{I}'_{i,\ell} \subseteq \mathcal{S}$ and hence I_i is happy. Therefore, it suffices to check that I_i intersects with I_r or I_ℓ to decide whether I_i is happy, and we have

$$f(\mathcal{I}_{i,\ell};r,k') = \begin{cases} f(\mathcal{I}_{i-1,\ell};r,k' - 1) & \text{if } I_i \cap I_r \neq \emptyset \text{ or } I_i \cap I_\ell \neq \emptyset, \\ f(\mathcal{I}_{i-1,\ell};r,k' - 1) + 1 & \text{otherwise.} \end{cases}$$

Case (ii): $r = i$

If $i = 0$, an interval representation $\mathcal{I}_{i,\ell}$ consists of the two disjoint intervals I_0 and I_ℓ . Only $\mathcal{S} = \emptyset$ satisfies the prescribed conditions for $f(\mathcal{I}_{i,\ell};i,k')$. Thus, for any integer $\ell > 0$, we have $f(\mathcal{I}_{i,\ell};i,k') = 0$ if $k' = 0$; otherwise $f(\mathcal{I}_{i,\ell};i,k') = -\infty$. Suppose that $i > 0$. To calculate $f(\mathcal{I}_{i,\ell};i,k')$ for each integers i, ℓ , and k' , we give the following lemma.

Lemma 5 *Let \mathcal{S} be a subset of $\mathcal{I}_{i,\ell} \setminus \{I_0, I_i, I_\ell\}$ such that $|\mathcal{S}| = k'$. In addition, let $\ell' \in \{i, \ell\}$ be an integer such that $\text{left}(I_{\ell'}) \leq \min\{\text{left}(I_i), \text{left}(I_\ell)\}$. Then, $H(\mathcal{I}_{i,\ell};\mathcal{S}) = H(\mathcal{I}_{i-1,\ell'};\mathcal{S})$.*

Proof We first show that $H(\mathcal{I}_{i,\ell};\mathcal{S}) \subseteq H(\mathcal{I}_{i-1,\ell'};\mathcal{S})$. Let $I' \in \{I_i, I_\ell\}$ be the other interval that is distinct from $I_{\ell'}$. Since $I' \notin \mathcal{S}$, we have $H(\mathcal{I}_{i,\ell};\mathcal{S}) = H(\mathcal{I}_{i,\ell};\mathcal{S}) \setminus \{I'\}$. Moreover, from Lemma 3, it holds that

$$H(\mathcal{I}_{i,\ell};\mathcal{S}) \setminus \{I'\} \subseteq H(\mathcal{I}_{i,\ell} \setminus \{I'\};\mathcal{S} \setminus \{I'\}) = H(\mathcal{I}_{i-1,\ell'};\mathcal{S}).$$

Thus, we have $H(\mathcal{I}_{i,\ell};\mathcal{S}) \subseteq H(\mathcal{I}_{i-1,\ell'};\mathcal{S})$.

We next show that $H(\mathcal{I}_{i-1,\ell'};\mathcal{S}) \subseteq H(\mathcal{I}_{i,\ell};\mathcal{S})$. Since $I_{\ell'} \notin \mathcal{S}$, we have $\text{right}(I_j) < \text{left}(I_{\ell'})$ for any interval $I_j \in H(\mathcal{I}_{i-1,\ell'};\mathcal{S})$. In addition, $\text{left}(I_{\ell'}) \leq \min\{\text{left}(I_i), \text{left}(I_\ell)\}$ holds from the definition of $I_{\ell'}$. Therefore, I_j intersects with

neither I_j nor I_ℓ , which implies that $I_j \in H(\mathcal{I}_{i,\ell}; \mathcal{S})$. We conclude that $H(\mathcal{I}_{i-1,\ell}; \mathcal{S}) \subseteq H(\mathcal{I}_{i,\ell}; \mathcal{S})$. □

Lemma 5 immediately leads to the following equation:

$$f(\mathcal{I}_{i,\ell}; i, k') = \begin{cases} f_{\max}(\mathcal{I}_{i-1,i}; k') & \text{if } \text{left}(I_i) < \text{left}(I_\ell), \\ f_{\max}(\mathcal{I}_{i-1,\ell}; k') & \text{otherwise.} \end{cases}$$

4.2 Running Time

We bound the running time of our algorithm for interval graphs. Given an n -vertex interval graph G and an integer k , we first convert G to its interval representation \mathcal{I} in linear time and add the dummy intervals I_0 and I_{n+1} . Recall that each endpoint of intervals can be assigned to an integer between -1 to $2n+2$. We can sort intervals of \mathcal{I} in increasing order of their right endpoints by an $O(n)$ -time integer sorting algorithm. For each integer $i \in \{0, \dots, n+1\}$, we compute \mathcal{I}_i and $\overline{\mathcal{I}}_i$ in $O(n^2)$ time.

We then compute values $f(\mathcal{I}_{i,\ell}; r, k')$ and $f_{\max}(\mathcal{I}_{i,\ell}; k')$ for each integers i, r, ℓ , and k' with $0 \leq i < \ell \leq n+1$, $\max\{0, i - k'\} \leq r \leq i$, and $0 \leq k' \leq k$. In fact, for each i , the interval I_ℓ can be chosen from a set $\mathcal{I}' \subseteq \overline{\mathcal{I}}_i$ of size at most $k+1$, where \mathcal{I}' consists of the first $k+1$ intervals of $\overline{\mathcal{I}}_i$ sorted in increasing order of their *left* endpoints; otherwise, from the definition of I_ℓ , an optimal solution \mathcal{S}^* of \mathcal{I} has size at least $k+1$, a contradiction.

Let (i, r, ℓ, k') be a quadruple of integers. In the case of $r < i$, our algorithm computes $f(\mathcal{I}_{i,\ell}; r, k')$ in $O(1)$ time by checking whether $I_i \cap I_r \neq \emptyset$ or $I_i \cap I_\ell \neq \emptyset$. In the case of $r = i$, our algorithm computes $f(\mathcal{I}_{i,\ell}; i, k')$ in $O(1)$ time by checking whether $\text{left}(I_i) < \text{left}(I_\ell)$. Thus, for all quadruples (i, r, ℓ, k') , the value in each case can be computed in $O(k^3n)$ time. Obviously, for all triples (i, ℓ, k') , $f_{\max}(\mathcal{I}_{i,\ell}; k')$ can be computed in $O(k^3n)$ time.

Finally, by taking the value $f_{\max}(\mathcal{I}_{n,n+1}; k)$ in $O(1)$ time, we determine the maximum number of happy vertices for the given interval graph G and integer k . The total running time of our algorithm is $O(n^2 + k^3n)$, as claimed in Theorem 2. □

5 Extension of Algorithmic Techniques

In this section, we demonstrate that our idea for interval graphs is applicable to other subclasses of co-comparability graphs.

5.1 Permutation Graphs

Let $\pi = (\pi(1), \pi(2), \dots, \pi(n))$ be a permutation of the integers from 1 to n . We denote by $\pi^{-1}(i)$ the position of the number i in π . Consider two horizontal parallel lines on the plane, where the upper line has distinct n points labeled $1, 2, \dots, n$, and the lower line has distinct n points labeled $\pi(1), \pi(2), \dots, \pi(n)$ in the order from left

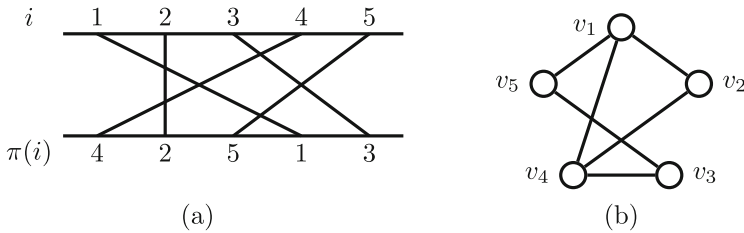


Fig. 6 **a** The line representation and **b** its corresponding permutation graph, where $\pi = (4, 2, 5, 1, 3)$

to right, respectively. For each integer $i \in \{1, 2, \dots, n\}$, let L_i be a line segment from a label i on the upper line to a label i on the lower line. A graph $G = (V, E)$ with vertices v_1, v_2, \dots, v_n is called a *permutation graph* if there exists a permutation π between 1 and n such that for any two distinct integers $i, j \in \{1, \dots, n\}$, $v_i v_j \in E$ if and only if L_i intersects with L_j . A set \mathcal{L} of line segments corresponding to G is called a *line representation* of G . We illustrate an example of the line representation and its corresponding permutation graph in Fig. 6.

Theorem 6 *Given an n -vertex permutation graph G and a non-negative integer k , MAXIMUM HAPPY SET is solvable in $O(n^2 + k^3n)$ time.*

The algorithmic approach for permutation graphs is basically the same as for interval graphs in Sect. 4. Given a permutation graph G , a line representation \mathcal{L} of G can be constructed in linear time [13]. We say that a line segment $L \in \mathcal{L}$ is *happy with respect to \mathcal{S}* if $L' \in \mathcal{S}$ for every line segment L' such that $L \cap L' \neq \emptyset$. MAXIMUM HAPPY SET on \mathcal{L} is the problem of finding $\mathcal{S} \subseteq \mathcal{L}$ such that the size of a set $H(\mathcal{L}; \mathcal{S})$ of happy line segments with respect to \mathcal{S} is maximized. We add a dummy line segment L_0 (resp. L_{n+1}) that connects a label 0 (resp. $n + 1$) on the upper line and a label $\pi(0) = 0$ (resp. $\pi(n + 1) = n + 1$) on the lower line. Both L_0 and L_{n+1} intersect with none of the other line segments in \mathcal{L} . For an integer $i \in \{0, \dots, n + 1\}$, we let $\mathcal{L}_i = \{L_0, L_1, \dots, L_i\}$ and $\overline{\mathcal{L}}_i = \mathcal{L} \setminus \mathcal{L}_i$.

Let \mathcal{S}^* be an optimal solution for \mathcal{L} and let $\mathcal{S}_i = \mathcal{S}^* \cap \mathcal{L}_i$. To characterize a partial solution \mathcal{S}_i of \mathcal{L}_i , we seek integers r, ℓ , and k' that satisfy the following three conditions for \mathcal{S}^* :

- $L_r \in \mathcal{L}_i \setminus \mathcal{S}^*$ and $\pi^{-1}(i') \leq \pi^{-1}(r)$ for any $L_{i'} \in \mathcal{L}_i \setminus \mathcal{S}^*$;
- $L_\ell \in \overline{\mathcal{L}}_i \setminus \mathcal{S}^*$ and $\pi^{-1}(\ell) \leq \pi^{-1}(i')$ for any $L_{i'} \in \overline{\mathcal{L}}_i \setminus \mathcal{S}^*$; and
- $|\mathcal{S}_i| = k'$.

We say that a quadruple (i, r, ℓ, k') of integers is *compatible* with \mathcal{S}^* if i, r, ℓ , and k' satisfy the above three conditions (see Fig. 7). For integers i, r , and ℓ , let $\mathcal{L}_{i,\ell}$ be a shorthand for $\mathcal{L}_i \cup \{L_\ell\}$ and let $\overrightarrow{\mathcal{L}}_r = \{L_{i'} \in \mathcal{L}_i \mid \pi^{-1}(r) < \pi^{-1}(i')\}$. For example, if $i = 4$ in Fig. 7, we have $\overrightarrow{\mathcal{L}}_r = \{L_1, L_4\}$ because $r = 2$. Analogous to Lemma 4, we give the following lemma.

Lemma 7 *For a line representation \mathcal{L} and an integer k , let \mathcal{S}^* be an optimal solution of $\mathcal{L} \setminus \{L_0, L_{n+1}\}$ such that $|\mathcal{S}^*| = k$. Suppose that a quadruple (i, r, ℓ, k') of integers is compatible with \mathcal{S}^* and $\mathcal{S}'_i \subseteq \mathcal{L}_{i,\ell} \setminus \{L_0, L_r, L_\ell\}$ is an optimal solution of $\mathcal{L}_{i,\ell}$ such*

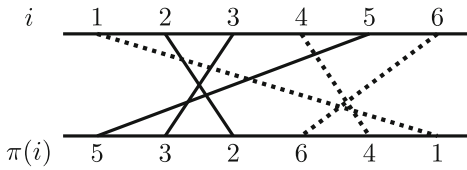


Fig. 7 For the above line representation $\mathcal{L} = \{L_1, L_2, \dots, L_6\}$, if $k = 3$, the optimal solution \mathcal{S}^* is $\{L_1, L_4, L_6\}$, which is drawn with dotted line segments. For $i = 4$, $\mathcal{L}_i \setminus \mathcal{S}^* = \{L_2, L_3\}$ with $\pi^{-1}(3) < \pi^{-1}(2)$, $\overline{\mathcal{L}}_i \setminus \mathcal{S}^* = \{L_5\}$, and $|\mathcal{S}_i| = |\mathcal{S}^* \cap \mathcal{L}_i| = 2$. Thus, a quadruple $(4, 2, 5, 2)$ is compatible with \mathcal{S}^*

that $\overrightarrow{\mathcal{L}}_r \subseteq \mathcal{S}'_i$ and $|\mathcal{S}'_i| = k'$. Then, there is an optimal solution \mathcal{S}^* of $\mathcal{L} \setminus \{L_0, L_{n+1}\}$ such that $\mathcal{S}'_i = \mathcal{S}^* \cap \mathcal{L}_i$.

Proof We replace the words and notations in the proof of Lemma 4 as follows:

- an interval $I \rightarrow$ a line segment L ;
- an interval representation $\mathcal{I} \rightarrow$ a line representation \mathcal{L} ; and
- left and right $\rightarrow \pi^{-1}$.

One can verify the correctness of Lemma 7 by the proof after replacing them. □

5.1.1 Algorithm

Given a line representation \mathcal{L} and an integer k , suppose that i, r, ℓ , and k' are integers such that $0 \leq r \leq i < \ell \leq n + 1$ and $0 \leq k' \leq k$. Our algorithm for permutation graphs employs the following three functions:

- $g_{in}(\mathcal{L}_{i,\ell}; r, k')$ returns the maximum of $|H(\mathcal{L}_{i,\ell}; \mathcal{S})|$ among all subsets $\mathcal{S} \subseteq \mathcal{L}_{i,\ell} \setminus \{L_0, L_r, L_\ell\}$ such that $\overrightarrow{\mathcal{L}}_r \cup \{L_i\} \subseteq \mathcal{S}$ and $|\mathcal{S}| = k'$;
- $g_{out}(\mathcal{L}_{i,\ell}; r, k')$ returns the maximum of $|H(\mathcal{L}_{i,\ell}; \mathcal{S})|$ among all subsets $\mathcal{S} \subseteq \mathcal{L}_{i,\ell} \setminus \{L_0, L_r, L_i, L_\ell\}$ such that $\overrightarrow{\mathcal{L}}_r \subseteq \mathcal{S}$ and $|\mathcal{S}| = k'$; and
- $g_{max}(\mathcal{L}_{i,\ell}; k')$ returns the maximum of $|H(\mathcal{L}_{i,\ell}; \mathcal{S})|$ among all subsets $\mathcal{S} \subseteq \mathcal{L}_{i,\ell} \setminus \{L_0, L_\ell\}$ such that $|\mathcal{S}| = k'$.

If there exists no subset \mathcal{S} that satisfies the prescribed conditions, we set $g_{in}(\mathcal{L}_{i,\ell}; r, k') = -\infty$, $g_{out}(\mathcal{L}_{i,\ell}; r, k') = -\infty$, and $g_{max}(\mathcal{L}_{i,\ell}; k') = -\infty$. It is easy to see that a value $g_{max}(\mathcal{L}_{i,\ell}; k')$ can be obtained as follows:

$$g_{max}(\mathcal{L}_{i,\ell}; k') = \max_{0 \leq r \leq i} \{g_{in}(\mathcal{L}_{i,\ell}; r, k'), g_{out}(\mathcal{L}_{i,\ell}; r, k')\}.$$

In the remainder of this subsection, we will explain the computation of the functions $g_{in}(\mathcal{L}_{i,\ell}; r, k')$ and $g_{out}(\mathcal{L}_{i,\ell}; r, k')$.

Computation of the function g_{in} : If $i = 0$, we have $g_{in}(\mathcal{L}_{i,\ell}; r, k') = -\infty$ for any r, ℓ , and k' because there exists no subset $\mathcal{S} \subseteq \mathcal{L}_{i,\ell}$ that satisfies $L_i \in \mathcal{S}$ and $L_0 \notin \mathcal{S}$. Similarly, if $i = r$, we have $g_{in}(\mathcal{L}_{i,\ell}; r, k') = -\infty$ for any r, ℓ , and k' ; and if $k' = 0$, we have $g_{in}(\mathcal{L}_{i,\ell}; r, k') = -\infty$ for any i, r , and ℓ . In the remainder of this subsection, we suppose that $i > 0, i \neq r$, and $k' > 0$.

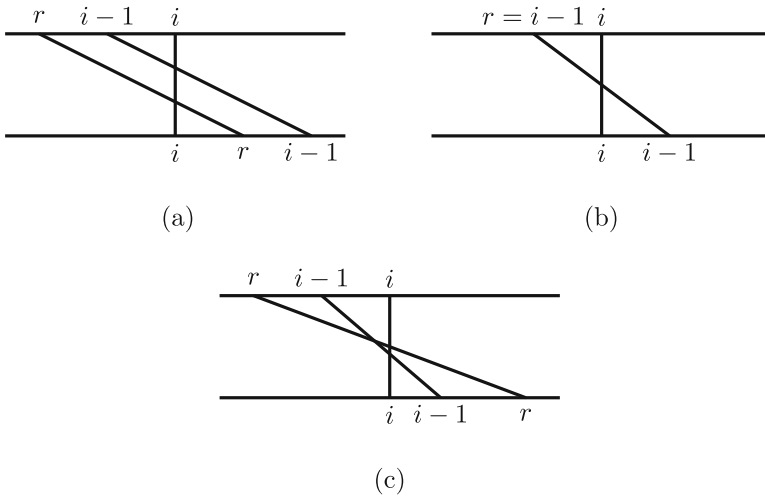


Fig. 8 The line representations of the case where $\pi^{-1}(i) < \pi^{-1}(r)$ and **a** $\pi^{-1}(i - 1) > \pi^{-1}(r)$, **b** $\pi^{-1}(i - 1) = \pi^{-1}(r)$, and **c** $\pi^{-1}(i - 1) < \pi^{-1}(r)$, respectively

Let \mathcal{S} be a subset of $\mathcal{L}_{i,\ell} \setminus \{L_0, L_r, L_\ell\}$ such that $|H(\mathcal{L}_{i,\ell}; \mathcal{S})|$ is the maximum among all subsets $\mathcal{S} \subseteq \mathcal{L}_{i,\ell} \setminus \{L_0, L_r, L_\ell\}$ with $\vec{\mathcal{L}}_r \cup \{L_i\} \subseteq \mathcal{S}$ and $|\mathcal{S}| = k'$. From Lemma 3, it holds that $H(\mathcal{L}_{i,\ell}; \mathcal{S}) \setminus \{L_i\} = H(\mathcal{L}_{i-1,\ell}; \mathcal{S} \setminus \{L_i\})$. We thus take values $g_{\text{in}}(\mathcal{L}_{i-1,\ell}; r, k' - 1)$ and $g_{\text{out}}(\mathcal{L}_{i-1,\ell}; r, k' - 1)$ to compute $g_{\text{in}}(\mathcal{L}_{i,\ell}; r, k')$. However, L_i may be unhappy with respect to \mathcal{S} . To determine whether $L_i \in H(\mathcal{L}_{i,\ell}; \mathcal{S})$, we consider whether L_i intersects with L_r or L_ℓ . If L_i intersects with L_r or L_ℓ , then L_i is clearly unhappy. Conversely, suppose that L_i intersects with neither L_r nor L_ℓ . Let $\mathcal{L}'_{i,\ell}$ be a set of line segments of $\mathcal{L}_{i,\ell}$ that intersect with L_i . Then, $\pi^{-1}(i) < \pi^{-1}(j)$ holds for any $L_j \in \mathcal{L}'_{i,\ell}$. Since L_i does not intersect with L_r , we have $\pi^{-1}(r) < \pi^{-1}(i)$ and hence $\mathcal{L}'_{i,\ell} \subseteq \vec{\mathcal{L}}_r \subseteq \mathcal{S}$ holds from the definition of L_r , which means that L_i is happy. Thus, to determine that L_i is happy, it suffice to confirm whether L_i intersects with L_r or L_ℓ .

In addition, we must pay attention to whether $L_{i-1} \in \mathcal{S}$ or not to compute $g_{\text{in}}(\mathcal{L}_{i,\ell}; r, k')$ (see also Fig. 8). If $\pi^{-1}(i - 1) > \pi^{-1}(r)$, then $L_{i-1} \in \vec{\mathcal{L}}_r$ and hence $L_{i-1} \in \mathcal{S}$. We thus have $g_{\text{in}}(\mathcal{L}_{i,\ell}; r, k')$ from $g_{\text{in}}(\mathcal{L}_{i-1,\ell}; r, k' - 1)$. If $\pi^{-1}(i - 1) = \pi^{-1}(r)$, then we take $g_{\text{out}}(\mathcal{L}_{i-1,\ell}; r, k' - 1)$ because $L_{i-1} = L_r$ and $L_r \notin \mathcal{S}$. If $\pi^{-1}(i - 1) < \pi^{-1}(r)$, then we compare values $g_{\text{in}}(\mathcal{L}_{i-1,\ell}; r, k' - 1)$ and $g_{\text{out}}(\mathcal{L}_{i-1,\ell}; r, k' - 1)$, and take the larger one. In summary, we obtain the following formula:

$$g_{\text{in}}(\mathcal{L}_{i,\ell};r,k') = \begin{cases} g_{\text{in}}(\mathcal{L}_{i-1,\ell};r,k'-1) + c & \text{if } \pi^{-1}(i-1) > \pi^{-1}(r), \\ g_{\text{out}}(\mathcal{L}_{i-1,\ell};r,k'-1) + c & \text{if } \pi^{-1}(i-1) = \pi^{-1}(r), \text{ and} \\ \max\{g_{\text{in}}(\mathcal{L}_{i-1,\ell};r,k'-1), \\ g_{\text{out}}(\mathcal{L}_{i-1,\ell};r,k'-1)\} + c & \text{otherwise,} \end{cases}$$

where

$$c = \begin{cases} 0 & \text{if } L_i \cap L_r \neq \emptyset \text{ or } L_i \cap L_\ell \neq \emptyset \\ 1 & \text{otherwise.} \end{cases}$$

Computation of the function g_{out} : Suppose that $i = 0$. As in Case (ii) of Sect. 4.1, for any integer $\ell > 0$, we have $g_{\text{out}}(\mathcal{L}_{i,\ell};r,k') = 0$ if $r = 0$ and $k' = 0$; otherwise $g_{\text{out}}(\mathcal{L}_{i,\ell};r,k') = -\infty$. For $i > 0$, we give the following lemma, which can be proved in the same way as the proof of Lemma 5.

Lemma 8 *Let \mathcal{S} be a subset of $\mathcal{L}_{i,\ell} \setminus \{L_0, L_r, L_i, L_\ell\}$ such that $|\mathcal{S}| = k'$. In addition, let $\ell' \in \{i, \ell\}$ be an integer such that $\pi^{-1}(\ell') \leq \min\{\pi^{-1}(i), \pi^{-1}(\ell)\}$. Then, $H(\mathcal{L}_{i,\ell};\mathcal{S}) = H(\mathcal{L}_{i-1,\ell'};\mathcal{S})$.*

It should be noted that, in contrast to the case for interval graphs, $L_i \notin \mathcal{S}$ does not always mean $r = i$. It is guaranteed that $\pi^{-1}(i) \leq \pi^{-1}(r)$; otherwise, $L_i \in \mathcal{S}$, contradicting the condition of g_{out} . Therefore, in addition to whether $\pi^{-1}(i) < \pi^{-1}(\ell)$ or not, the relationship between $i, i - 1$, and r must also be carefully considered (see again Fig. 8). We have

$$g_{\text{out}}(\mathcal{L}_{i,\ell};r,k') = \begin{cases} g_{\max}(\mathcal{L}_{i-1,\ell'};k') & \text{if } \pi^{-1}(i) = \pi^{-1}(r), \\ g_{\text{in}}(\mathcal{L}_{i-1,\ell'};r,k') & \text{if } \pi^{-1}(i) < \pi^{-1}(r) \text{ and } \pi^{-1}(i-1) > \pi^{-1}(r), \\ g_{\text{out}}(\mathcal{L}_{i-1,\ell'};r,k') & \text{if } \pi^{-1}(i) < \pi^{-1}(r) \text{ and } \pi^{-1}(i-1) = \pi^{-1}(r), \text{ and} \\ \max\{g_{\text{in}}(\mathcal{L}_{i-1,\ell'};r,k'), \\ g_{\text{out}}(\mathcal{L}_{i-1,\ell'};r,k')\} & \text{otherwise,} \end{cases}$$

where $\ell' \in \{i, \ell\}$ is an integer such that $\pi^{-1}(\ell') \leq \min\{\pi^{-1}(i), \pi^{-1}(\ell)\}$.

5.1.2 Running Time

Recall that a line representation \mathcal{L} can be constructed in linear time from a given n -vertex permutation graph G . We add dummy line segments L_0 and L_{n+1} and we then compute \mathcal{L}_i and $\overline{\mathcal{L}}_i$ in $O(n)$ time for every $i \in \{0, \dots, n + 1\}$. This preprocessing can be done in $O(n^2)$ time.

We bound the running time to compute $g_{\max}(\mathcal{L}_{i,\ell};k')$, $g_{\text{in}}(\mathcal{L}_{i,\ell};r,k')$, and $g_{\text{out}}(\mathcal{L}_{i,\ell};r,k')$ for each integers i, r, ℓ , and k' , where $0 \leq r \leq i < \ell \leq n + 1$ and $0 \leq k' \leq k$. Notice that we do not need to consider an integer r such that

$|\vec{\mathcal{L}}_r| > k'$ because it violates the condition that $|\mathcal{S}| = k'$. Similarly, we can ignore an integer ℓ such that the size of $\{L_{i'} \in \overline{\mathcal{L}}_i \mid \pi^{-1}(i') < \pi^{-1}(\ell)\}$ is at least $k + 1$. Thus, we can compute $g_{\max}(\mathcal{L}_i, \ell; k')$ in $O(k)$ time for each triple (i, ℓ, k') , and compute $g_{\max}(\mathcal{L}_i, \ell; k')$ in $O(k^3 n)$ time for all triples (i, ℓ, k') . Moreover, since both $g_{\text{in}}(\mathcal{L}_i, \ell; r, k')$ and $g_{\text{out}}(\mathcal{L}_i, \ell; r, k')$ are computed in $O(1)$ time for each quadruple (i, r, ℓ, k') , they are computed in $O(k^3 n)$ time for all quadruples (i, r, ℓ, k') . Therefore, the total running time of our algorithm is $O(n^2 + k^3 n)$, as claimed in Theorem 6. \square

5.2 d -Trapezoid Graphs

Our definition of d -trapezoid graphs follows [14]. Let d be a fixed positive integer and consider d horizontal parallel lines L_1, L_2, \dots, L_d on the plane. A polygon T is called d -trapezoid if an interval $[\text{left}_j(T), \text{right}_j(T)]$ on L_j for each $j \in \{1, \dots, d\}$ is defined and T is formed by connecting points $\text{left}_1(T), \text{left}_2(T), \dots, \text{left}_d(T), \text{right}_d(T), \text{right}_{d-1}(T), \dots, \text{right}_1(T), \text{left}_1(T)$. A graph $G = (V, E)$ with vertices v_1, v_2, \dots, v_n is called a d -trapezoid graph if there exists a set $\mathcal{T} = \{T_1, \dots, T_n\}$ of d -trapezoids such that for any two distinct integers $i, j \in \{1, \dots, n\}$, $v_i v_j \in E$ if and only if T_i intersects with T_j . A set \mathcal{T} of d -trapezoids that represents G is called a d -trapezoid diagram of G . Figure 9 shows an example of (a) a 3-trapezoid diagram and (b) its corresponding 3-trapezoid graph. Without loss of generality, for each $j \in \{1, \dots, d\}$, we suppose that $\text{left}_j(T), \text{right}_j(T) \in \{1, \dots, 2n\}$ for any $T \in \mathcal{T}$, and any intervals on L_j have no endpoint in common.

Unfortunately, computing a d -trapezoid diagram of a given graph G is known to be NP-hard for any $d \geq 3$ [15]. We thus assume that a d -trapezoid diagram of G is also provided as part of an input of MAXIMUM HAPPY SET.

Theorem 9 *Given an n -vertex d -trapezoid graph G along with its d -trapezoid diagram and a non-negative integer k , MAXIMUM HAPPY SET is solvable in $O(n^2 + d^2(k + 1)^{3d}n)$ time.*

All interval graphs are 1-trapezoid graphs, and all permutation graphs and trapezoid graphs are 2-trapezoid graphs. Moreover, for $d \in \{1, 2\}$, a d -trapezoid diagram of a given d -trapezoid graph can be obtained in polynomial time [10, 16]. Therefore, Theorem 9 indeed indicates the polynomial-time solvability of MAXIMUM HAPPY SET on interval graphs, permutation graphs, and trapezoid graphs. Notice that Theorem 6 improves the running time of Theorem 9 for $d = 2$ by applying good properties of permutation graphs.

Similar arguments for interval graphs in Sect. 4 are also applicable to d -trapezoid graphs. The only difference of our algorithm for d -trapezoid graphs is that vertices *not* contained in an optimal solution are characterized by a set of d -trapezoids of size at most $2d$, whereas for interval graphs, such vertices are characterized by only two intervals. Thus, we here only describe the notations and the formulae to compute the optimal value without detailed proofs.

We first label d -trapezoids of \mathcal{T} with $1, \dots, n$ such that $\text{right}_1(T_i) < \text{right}_1(T_j)$ for any i, j with $1 \leq i < j \leq n$. We add dummy trapezoids T_0 and T_{n+1} into a d -trapezoid

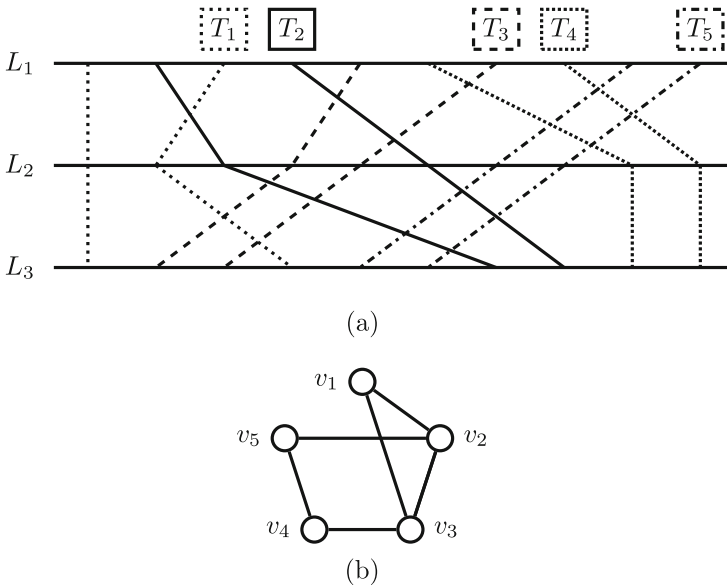


Fig. 9 **a** The 3-trapezoid diagram T . Each T_i surrounded by a solid, dotted, or dashed rectangle is associated with a 3-trapezoid on T formed by the same type of lines. **b** The 3-trapezoid graph corresponding to T

diagram of G , where $\text{right}_j(T_0) < \text{left}_j(T_1)$ and $\text{right}_j(T_n) < \text{left}_j(T_{n+1})$ for every $j \in \{1, \dots, d\}$. Our task is to find an optimal solution $S^* \subseteq T \setminus \{T_0, T_{n+1}\}$ with the maximum set $H(T; S^*)$ of happy d -trapezoids. For an integer $i \in \{0, \dots, n + 1\}$, we let $\mathcal{T}_i = \{T_0, T_1, \dots, T_i\}$ and $\overline{\mathcal{T}}_i = T \setminus \overline{\mathcal{T}}_i$.

We next give the additional notations to define sets corresponding to I_r and I_ℓ in Sect. 4. For non-empty sets $\mathcal{R} \subseteq \mathcal{T}_i$ and $\mathcal{L} \subseteq \overline{\mathcal{T}}_i$, let

- $\vec{\mathcal{R}} = \{T \in \mathcal{T}_i \mid \forall T' \in \mathcal{R} \exists j \in \{1, \dots, d\} : \text{right}_j(T') < \text{right}_j(T)\}$; and
- $\overleftarrow{\mathcal{L}} = \{T \in \overline{\mathcal{T}}_i \mid \forall T' \in \mathcal{L} \exists j \in \{1, \dots, d\} : \text{left}_j(T) < \text{left}_j(T')\}$.

For two non-empty subsets \mathcal{R} and \mathcal{R}' of \mathcal{T}_i , we say that \mathcal{R} and \mathcal{R}' are *equivalent*, denoted by $\mathcal{R} \sim \mathcal{R}'$, if $\vec{\mathcal{R}} = \vec{\mathcal{R}'}$. Similarly, for two non-empty subsets \mathcal{L} and \mathcal{L}' of $\overline{\mathcal{T}}_i$, we let $\mathcal{L} \sim \mathcal{L}'$ if $\overleftarrow{\mathcal{L}} = \overleftarrow{\mathcal{L}'}$. If $\mathcal{R} \subseteq \mathcal{T}_i$ is the inclusion-wise minimal set with respect to the equivalence class $[\mathcal{R}] = \{\mathcal{R}' \mid \mathcal{R} \sim \mathcal{R}'\}$, then \mathcal{R} is called a *representative* of $[\mathcal{R}]$. Notice that such representative is uniquely determined. A representative of $\mathcal{L} \subseteq \overline{\mathcal{T}}_i$ is also defined in the similar way. For sets $\mathcal{R} \subseteq \mathcal{T}_i$ and $\mathcal{L} \subseteq \overline{\mathcal{T}}_i$, the mappings $\text{rep}_i(\mathcal{R})$ and $\text{rep}_i(\mathcal{L})$ return the representatives of $[\mathcal{R}]$ and $[\mathcal{L}]$, respectively. For an optimal solution $S^* \subseteq T$, we let $\mathcal{R}^* = \text{rep}_i(\mathcal{T}_i \setminus S^*)$ and $\mathcal{L}^* = \text{rep}_i(\overline{\mathcal{T}}_i \setminus S^*)$. Note that $\vec{\mathcal{R}}^* \subseteq S^*$ and $\overleftarrow{\mathcal{L}}^* \subseteq S^*$ hold. We guess non-empty sets $\mathcal{R} \subseteq \mathcal{T}_i$ and $\mathcal{L} \subseteq \overline{\mathcal{T}}_i$ such that $\mathcal{R}^* = \mathcal{R}$ and $\mathcal{L}^* = \mathcal{L}$, and then we compute a partial solution of \mathcal{T}_i under the guess.

Let \mathcal{R}_i and \mathcal{L}_i be non-empty families of subsets of \mathcal{T}_i and $\overline{\mathcal{T}}_i$ such that for each $\mathcal{R} \in \mathcal{R}_i$ and $\mathcal{L} \in \mathcal{L}_i$, \mathcal{R} and \mathcal{L} are the representatives of $[\mathcal{R}]$ and $[\mathcal{L}]$, respectively. We write $\mathcal{T}_{i,\mathcal{L}} = \mathcal{T}_i \cup \mathcal{L}$ and define the following function:

- $t(\mathcal{T}_{i,\mathcal{L}}; \mathcal{R}, k')$ returns the maximum of $|H(\mathcal{T}_{i,\mathcal{L}}; \mathcal{S})|$ among all subsets $\mathcal{S} \subseteq \mathcal{T}_{i,\mathcal{L}} \setminus (\{T_0\} \cup \mathcal{R} \cup \mathcal{L})$ such that $\vec{\mathcal{R}} \subseteq \mathcal{S}$ and $|\mathcal{S}| = k'$.

We let $t(\mathcal{T}_{i,\mathcal{L}}; \mathcal{R}, k') = -\infty$ if there exists no subset \mathcal{S} that satisfies all the prescribed conditions. Then, $t(\mathcal{T}_{i,\mathcal{L}}; \mathcal{R}, k')$ is computed as follows: if $T_i \notin \mathcal{R}$,

$$t(\mathcal{T}_{i,\mathcal{L}}; \mathcal{R}, k') = \begin{cases} -\infty & \text{if } i = 0 \text{ or } k' = 0, \\ t(\mathcal{T}_{i-1,\mathcal{L}}; \mathcal{R}, k' - 1) & \text{if } i > 0, k' > 0 \text{ and } \exists T \in \mathcal{R} \cup \mathcal{L} : T_i \cap T \neq \emptyset, \\ t(\mathcal{T}_{i-1,\mathcal{L}}; \mathcal{R}, k' - 1) + 1 & \text{otherwise,} \end{cases}$$

and if $T_i \in \mathcal{R}$,

$$t(\mathcal{T}_{i,\mathcal{L}}; \mathcal{R}, k') = \begin{cases} 0 & \text{if } i = 0, \mathcal{R} = \{T_0\}, \text{ and } k' = 0, \\ \max_{\substack{\mathcal{R}' \in \mathcal{R}_{i-1} \\ \text{rep}_i(\mathcal{R}' \cup \{T_i\}) = \mathcal{R}}} t(\mathcal{T}_{i-1,\mathcal{L}'}; \mathcal{R}', k') & \text{if } i > 0, \\ -\infty & \text{otherwise,} \end{cases}$$

where $\mathcal{L}' = \text{rep}_{i-1}(\mathcal{L} \cup \{T_i\})$.

5.2.1 Running Time

Obviously, the running time of our algorithm crucially depends on \mathcal{R}_i and \mathcal{L}_i . The following lemma bounds the running time to construct \mathcal{R}_i and \mathcal{L}_i , and their sizes.

Lemma 10 *Suppose that, for each $j \in \{1, \dots, d\}$, we have obtained a set of d -trapezoids in a d -trapezoid diagram \mathcal{T} that are sorted in increasing order of the right endpoints of intervals on L_j . Then, for each $i \in \{0, \dots, n\}$, \mathcal{R}_i and \mathcal{L}_i of each size at most $(k + 1)^d$ can be constructed in $O(n + kd^2(k + 1)^{2d})$ time.*

Proof For each $j \in \{1, \dots, d\}$, let $\mathcal{T}_i^{j,k+1}$ be a maximal subset of \mathcal{T}_i of size at most $k + 1$ such that $\text{right}_j(T') < \text{right}_j(T)$ for any $T \in \mathcal{T}_i^{j,k+1}$ and any $T' \in \mathcal{T}_i \setminus \mathcal{T}_i^{j,k+1}$. In particular, if $i \leq k + 1$, we have $\mathcal{T}_i^{j,k+1} = \mathcal{T}_i$.

Recall that, for an optimal solution \mathcal{S}^* of size k , we define $\mathcal{R}^* = \text{rep}_i(\mathcal{T}_i \setminus \mathcal{S}^*)$ such that $\vec{\mathcal{R}}^* \subseteq \mathcal{S}^*$. Then, at least one d -trapezoid of \mathcal{R}^* is contained in $\mathcal{T}_i^{j,k+1}$. Assume for a contradiction that there is no d -trapezoid of \mathcal{R}^* in $\mathcal{T}_i^{j,k+1}$. This implies that $\mathcal{T}_i^{j,k+1} \subseteq \vec{\mathcal{R}}^* \subseteq \mathcal{S}^*$. If $i \leq k + 1$, then $\mathcal{T}_i = \mathcal{T}_i^{j,k+1} \subseteq \mathcal{S}^*$, which contradicts that $T_0 \notin \mathcal{S}^*$ and hence $\mathcal{T}_i \setminus \mathcal{S}^* \neq \emptyset$. If $i \geq k + 1$, then the size of $\mathcal{T}_i^{j,k+1}$ is exactly $k + 1$, which contradicts that the size of \mathcal{S}^* is exactly k .

In addition, for each $T \in \mathcal{R}^*$, there exists $j \in \{1, \dots, d\}$ such that $\text{right}_j(T') < \text{right}_j(T)$ for any $T' \in \mathcal{R}^* \setminus \{T\}$; otherwise, \mathcal{R}^* is not minimal with respect to the equivalence class $[\mathcal{R}^*]$, contradicting that $\mathcal{R}^* = \text{rep}_i(\mathcal{T}_i \setminus \mathcal{S}^*)$. Therefore, we construct

\mathcal{R}_i as follows. We enumerate possible candidates of \mathcal{R}^* by picking a d -trapezoid in $\mathcal{T}_i^{j,k+1}$ for each $j \in \{1, \dots, d\}$ repeatedly. Let \mathcal{R}'_i be a family of the enumerated sets. Then, we remove $\mathcal{R} \in \mathcal{R}'_i$ from \mathcal{R}'_i if it is inappropriate as a candidate because $|\vec{\mathcal{R}}| > k$, or it is redundant, that is, there exists $\mathcal{R}' \in \mathcal{R}'_i$ such that $\mathcal{R} \sim \mathcal{R}'$ and $|\mathcal{R}| \geq |\mathcal{R}'|$. Let \mathcal{R}_i be a family obtained after the reduction.

We bound the running time to construct \mathcal{R}_i . Recall that, for each $j \in \{1, \dots, d\}$, we have sorted d -trapezoids in a d -trapezoid diagram \mathcal{T} in increasing order of the right endpoints of intervals on L_j . For each $i \in \{0, \dots, n\}$, we can obtain $\mathcal{T}_i^{j,k+1}$ in $O(n)$ time. Then, we construct \mathcal{R}'_i . Since the size of $\mathcal{T}_i^{j,k+1}$ is at most $k + 1$ for every $j \in \{1, \dots, d\}$, the size of \mathcal{R}'_i is at most $(k + 1)^d$. To obtain \mathcal{R}_i from \mathcal{R}'_i , we remove inappropriate sets as candidates of \mathcal{R}^* from \mathcal{R}'_i . Notice that for each $\mathcal{R} \in \mathcal{R}'_i$, we can construct $\vec{\mathcal{R}}$ in $O(kd^2)$ time because $|\mathcal{R}| \leq d$ and $\mathcal{T}_i^{j,k+1}$ has the size at most $k + 1$ for each $j \in \{1, \dots, d\}$. After removing all sets $\mathcal{R} \in \mathcal{R}'_i$ with $|\vec{\mathcal{R}}| > k$ from \mathcal{R}'_i , we also remove redundant sets by checking whether $\mathcal{R} \sim \mathcal{R}'$ and $|\mathcal{R}| \geq |\mathcal{R}'|$ for each pair $\mathcal{R}, \mathcal{R}' \in \mathcal{R}'_i$. This reduction of \mathcal{R}'_i can be done in $O(kd^2 \cdot |\mathcal{R}'_i| + (k + d) \cdot |\mathcal{R}'_i|^2) = O(kd^2(k + 1)^{2d})$ time. Consequently, \mathcal{R}_i of size at most $(k + 1)^d$ is obtained in $O(n + kd^2(k + 1)^{2d})$ time for each $i \in \{0, \dots, n\}$. The similar construction of \mathcal{L}_i can also be done in $O(n + kd^2(k + 1)^{2d})$ time for each $i \in \{0, \dots, n\}$. □

We now estimate the running time of our algorithm. First, for each $j \in \{1, \dots, d\}$, we sort d -trapezoids in a d -trapezoid diagram \mathcal{T} in $O(n)$ time in increasing order of the right endpoints of intervals on L_j and store the results of sorting. We then compute $\mathcal{T}_i, \overline{\mathcal{T}}_i, \mathcal{R}_i$, and \mathcal{L}_i for each $i \in \{0, \dots, n\}$. From Lemma 10, this can be done in $O(n^2 + kd^2(k + 1)^{2d}n)$ time.

Consider a quadruple $(i, \mathcal{R}, \mathcal{L}, k')$, where $\mathcal{R} \in \mathcal{R}_i, \mathcal{L} \in \mathcal{L}_i$, and $k' \in \{0, \dots, k\}$. If $T_i \notin \mathcal{R}$, we check whether $\exists T \in \mathcal{R} \cup \mathcal{L} : T_i \cap T \neq \emptyset$, where $|\mathcal{R} \cup \mathcal{L}| \leq 2d$ because $|\mathcal{R}| \leq d$ and $|\mathcal{L}| \leq d$. For a trapezoid $T \in \mathcal{R} \cup \mathcal{L}$, the following observation allows us to check whether $T_i \cap T \neq \emptyset$ in $O(d)$ time.

Observation 11 *Let \mathcal{T} be a d -trapezoid diagram. For two trapezoids $T, T' \in \mathcal{T}$, $T \cap T' = \emptyset$ if and only if*

- (1) $right_j(T) < left_j(T')$ for every $j \in \{1, \dots, d\}$, or
- (2) $right_j(T') < left_j(T)$ for every $j \in \{1, \dots, d\}$.

Thus, if $T_i \notin \mathcal{R}$, the value $t(\mathcal{T}_i, \mathcal{L}; \mathcal{R}, k')$ is computed in $O(d^2)$ time.

If $T_i \in \mathcal{R}$, we first obtain $rep_{i-1}(\mathcal{L} \cup \{T_i\})$ in $O(d^2)$ time by removing a d -trapezoid T from $\mathcal{L} \cup \{T_i\}$ such that $left_j(T_i) < left_j(T)$ for every $j \in \{1, \dots, d\}$. Then, for each $\mathcal{R}' \in \mathcal{R}_{i-1}$, we check in $O(d^2)$ time whether $rep_i(\mathcal{R}' \cup \{T_i\}) = \mathcal{R}$ and obtain a value $t(\mathcal{T}_{i-1}, \mathcal{L}'; \mathcal{R}', k')$ in $O(1)$ time. Thus, if $T_i \in \mathcal{R}$, the value $t(\mathcal{T}_i, \mathcal{L}; \mathcal{R}, k')$ is computed in $O(d^2 + d^2 \cdot |\mathcal{R}_{i-1}|) = O(d^2(k + 1)^d)$ time.

We finally bound the total running time of our algorithm. The preprocessing can be done in $O(n^2 + kd^2(k + 1)^{2d}n)$ time. Since $|\mathcal{R}_i| \leq (k + 1)^d$ and $|\mathcal{L}_i| \leq (k + 1)^d$, if $T_i \notin \mathcal{R}$, our algorithm computes the value $t(\mathcal{T}_i, \mathcal{L}; \mathcal{R}, k')$ in $O(d^2(k + 1)^{2d+1}n)$ time for all quadruples $(i, \mathcal{R}, \mathcal{L}, k')$. Notice that the number of $\mathcal{R} \in \mathcal{R}_i$ such that $T_i \in \mathcal{R}$ is at

most $(k+1)^{d-1}$. Therefore, for all quadruples $(i, \mathcal{R}, \mathcal{L}, k')$ with $T_i \in \mathcal{R}$, our algorithm computes the value $t(\mathcal{T}_i, \mathcal{L}; \mathcal{R}, k')$ in $O(d^2(k+1)^d \cdot (k+1)^{2d}n) = O(d^2(k+1)^{3d}n)$ time. As a conclusion, we obtain an optimal solution of MAXIMUM HAPPY SET on a d -trapezoid graph in $O(n^2 + d^2(k+1)^{3d}n)$ time. This completes the proof of Theorem 9. \square

6 Conclusion

In this paper, we studied the complexity of MAXIMUM HAPPY SET on subclasses of co-comparability graphs; co-bipartite graphs, interval graphs, permutation graphs, and d -trapezoid graphs. Especially, our algorithm for interval graphs improved the best known running time $O(kn^8)$. Our polynomial-time algorithms employ basically the same techniques.

The complexity of MAXIMUM HAPPY SET has been studied for various graph classes. However, the (in)tractability of MAXIMUM HAPPY SET on planar graphs remains open. We note that the complexity of the edge variant [9] of MAXIMUM HAPPY SET is also unknown for planar graphs.

Acknowledgements We are grateful to anonymous referees of the preliminary version [1] and of this journal version for their helpful suggestions. This work is partially supported by JSPS KAKENHI Grant Numbers JP18H04091, JP19K11814, JP20H05793, JP20H05794, JP20K11666, JP21K11755, and JP21K21278, Japan.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Eto, H., Ito, T., Miyano, E., Suzuki, A., Tamura, Y.: Happy set problem on subclasses of co-comparability graphs. In: Mutzel, P., Rahman, M.S., Slamin (eds.) Proceedings of the 16th International Conference and Workshop on Algorithms and Computation (WALCOM 2022). Lecture Notes in Computer Science, vol. 13174, pp. 149–160. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-96731-4_13
2. Easley, D., Kleinberg, J.: Networks, Crowds, and Markets: Reasoning About a Highly Connected World. Cambridge University Press, Cambridge (2010). <https://doi.org/10.1017/CBO9780511761942>
3. Zhang, P., Li, A.: Algorithmic aspects of homophily of networks. Theor. Comput. Sci. **593**, 117–131 (2015). <https://doi.org/10.1016/j.tcs.2015.06.003>
4. Asahiro, Y., Eto, H., Hanaka, T., Lin, G., Miyano, E., Terabaru, I.: Parameterized algorithms for the happy set problem. Discrete Appl. Math. **304**, 32–44 (2021). <https://doi.org/10.1016/j.dam.2021.07.005>
5. Asahiro, Y., Eto, H., Hanaka, T., Lin, G., Miyano, E., Terabaru, I.: Complexity and approximability of the happy set problem. Theor. Comput. Sci. **866**, 123–144 (2021). <https://doi.org/10.1016/j.tcs.2021.03.023>
6. Kratsch, D., Stewart, L.: Domination on cocomparability graphs. SIAM J. Discrete Math. **6**(3), 400–417 (1993). <https://doi.org/10.1137/0406032>
7. Deogun, J.S., Steiner, G.: Polynomial algorithms for Hamiltonian cycle in cocomparability graphs. SIAM J. Comput. **23**(3), 520–552 (1994). <https://doi.org/10.1137/S0097539791200375>
8. Coorg, S.R., Rangan, C.P.: Feedback vertex set on cocomparability graphs. Networks **26**(2), 101–111 (1995). <https://doi.org/10.1002/net.3230260205>

9. Feige, U., Kortsarz, G., Peleg, D.: The dense k -subgraph problem. *Algorithmica* **29**(3), 410–421 (2001). <https://doi.org/10.1007/s004530010050>
10. Booth, K.S., Lueker, G.S.: Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. *J. Comput. Syst. Sci.* **13**(3), 335–379 (1976). [https://doi.org/10.1016/S0022-0000\(76\)80045-1](https://doi.org/10.1016/S0022-0000(76)80045-1)
11. Corneil, D.G., Olariu, S., Stewart, L.: The LBFS structure and recognition of interval graphs. *SIAM J. Discrete Math.* **23**(4), 1905–1953 (2010). <https://doi.org/10.1137/S0895480100373455>
12. Hsu, W.-L., Ma, T.-H.: Fast and simple algorithms for recognizing chordal comparability graphs and interval graphs. *SIAM J. Comput.* **28**(3), 1004–1020 (1998). <https://doi.org/10.1137/S0097539792224814>
13. McConnell, R.M., Spinrad, J.P.: Modular decomposition and transitive orientation. *Discrete Math.* **201**(1), 189–241 (1999). [https://doi.org/10.1016/S0012-365X\(98\)00319-7](https://doi.org/10.1016/S0012-365X(98)00319-7)
14. Bodlaender, H.L., Kloks, T., Kratsch, D., Müller, H.: Treewidth and minimum fill-in on d -trapezoid graphs. *J. Graph Algorithms Appl.* **2**(5), 1–23 (1998). <https://doi.org/10.7155/jgaa.00008>
15. Yannakakis, M.: The complexity of the partial order dimension problem. *SIAM J. Algebr. Discrete Methods* **3**(3), 351–358 (1982). <https://doi.org/10.1137/0603036>
16. Ma, T.-H., Spinrad, J.P.: On the 2-chain subgraph cover and related problems. *J. Algorithms* **17**(2), 251–268 (1994). <https://doi.org/10.1006/jagm.1994.1034>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.