Discrete Applied Mathematics 236 (2018) 7–22

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Sublinear approximation algorithms for boxicity and related problems☆

Abhijin Adiga [a], Jasine Babu [b,*], L. Sunil Chandran [c]

[a] *Virginia Tech., USA*
[b] *Indian Institute of Technology, Palakkad, India*
[c] *Indian Institute of Science, Bangalore, India*

ARTICLE INFO

Article history:
Received 9 August 2015
Received in revised form 20 October 2017
Accepted 30 October 2017
Available online 1 December 2017

Keywords:
Boxicity
Approximation algorithm
Partial order dimension
Threshold dimension
Parameterization

ABSTRACT

The boxicity of a graph $G(V, E)$ is the minimum integer k such that G can be represented as the intersection graph of axis parallel boxes in $\mathbb{R}^k$. Cubicity is a variant of boxicity, where the axis parallel boxes in the intersection representation are restricted to be of unit length sides. Deciding whether the boxicity (resp. cubicity) of a graph is at most k is NP-hard, even for $k = 2$ or 3. Computing these parameters is inapproximable within $O(n^{1-\epsilon})$-factor, for any $\epsilon > 0$ in polynomial time unless NP $=$ ZPP, even for many simple graph classes.

In this paper, we give a polynomial time $\kappa(n)$ factor approximation algorithm for computing boxicity and a $\kappa(n)\lceil\log\log n\rceil$ factor approximation algorithm for computing the cubicity, where $\kappa(n) = 2\left\lceil n\sqrt{\log\log n}\,/\sqrt{\log n}\right\rceil$. These $o(n)$ factor approximation algorithms also produce the corresponding box (resp. cube) representations. As a special case, this resolves the question posed by Spinrad (2003) about polynomial time construction of $o(n)$ dimensional box representations for boxicity 2 graphs. Other consequences of our approximation algorithm include $O(\kappa(n))$ factor approximation algorithms for computing the following parameters: the partial order dimension (poset dimension) of finite posets, the interval dimension of finite posets, minimum chain cover of bipartite graphs, Ferrers dimension of digraphs and threshold dimension of split graphs and co-bipartite graphs. Each of these parameters is inapproximable within an $O(n^{1-\epsilon})$-factor, for any $\epsilon > 0$ in polynomial time unless NP $=$ ZPP and the algorithms we derive seem to be the first $o(n)$ factor approximation algorithms known for all these problems. We note that obtaining a $o(n)$ factor approximation for poset dimension was also mentioned as an open problem by Felsner et al. (2017).

In the second part of this paper, parameterized approximation algorithms for boxicity using various edit distance parameters are derived. We also present a parameterized approximation scheme for cubicity, using minimum vertex cover number as the parameter.

1. Introduction

Let $G(V, E)$ be a graph. If $I_1, I_2, \cdots, I_k$ are (unit) interval graphs on the vertex set V such that $E(G) = E(I_1) \cap E(I_2) \cap \cdots \cap E(I_k)$, then $\{I_1, I_2, \cdots, I_k\}$ is called a box (cube) representation of G of dimension k. Boxicity (cubicity) of a non-complete graph G, denoted by box(G) (respectively, cub(G)), is defined as the minimum integer k such that G has a box (cube) representation

☆ A part of this work was done at IISc Bangalore. A preliminary version of this work appeared in IPEC 2012.

* Corresponding author.
E-mail addresses: abhijin@vbi.vt.edu (A. Adiga), jasine@iitpkd.ac.in (J. Babu), sunil@csa.iisc.ernet.in (L.S. Chandran).

https://doi.org/10.1016/j.dam.2017.10.031
0166-218X/

Table 1
Parameterized approximations for computing boxicity and corresponding box representations using Theorems 3 and 4.

Parameter k	Approximation guarantee	Running time
Interval completion No.	Additive 2	$2^{O(k^2 \log k)} n^{O(1)}$
Feedback vertex set size	$2 + \frac{2}{\text{box}(G)}$ factor	$2^{O(k^2 \log k)} n^{O(1)}$
Proper interval vertex deletion No.	$2 + \frac{1}{\text{box}(G)}$ factor	$2^{O(k^2 \log k)} n^{O(1)}$
Proper interval edge deletion No.	Additive 2	$2^{O(k^2 \log k)} n^{O(1)}$
Planar vertex deletion No.	$2 + \frac{3}{\text{box}(G)}$ factor	$f(k) n^{O(1)}$
Crossing number	Additive 6	$f(k) n^{O(1)}$
Planar edge deletion No.	Additive 6	$f(k) n^{O(1)}$
Max leaf No.	Additive 2	$2^{O(k^3 \log k)} n^{O(1)}$

of dimension k. For a complete graph, it is defined to be zero. Equivalently, boxicity (cubicity) is the minimum integer k such that G can be represented as the intersection graph of axis parallel boxes (cubes) in $\mathbb{R}^k$. Boxicity was introduced by Roberts [29] in 1969 for modeling problems in social sciences and ecology. Some well known NP-hard problems like the max-clique problem are polynomial time solvable, if low dimensional box representations are known [30].

For any graph G on n vertices, $\text{box}(G) \le \left\lfloor \frac{n}{2} \right\rfloor$ and $\text{cub}(G) \le \left\lfloor \frac{2n}{3} \right\rfloor$. Upper bounds of boxicity in terms of parameters like maximum degree [2] and tree-width [12] are known. It was shown by Scheinerman [31] in 1984 that the boxicity of outer planar graphs is at most two. In 1986, Thomassen [34] proved that the boxicity of planar graphs is at most 3.

Computation of boxicity is a notoriously hard problem. Even for $k = 2$ or 3, deciding whether boxicity (resp. cubicity) of a graph is at most k is NP-complete [37,24,6]. Recently, Chalermsook et al. [10] proved that no polynomial time algorithm for approximating boxicity of bipartite graphs with approximation factor within $O(n^{1-\epsilon})$ for any $\epsilon > 0$ is possible unless NP = ZPP. Same non-approximability holds in the case of split graphs and co-bipartite graphs too. Since cubicity and boxicity are equal for co-bipartite graphs, these hardness results extend to cubicity as well.

Boxicity is also closely related to other dimensional parameters like poset dimension, interval dimension, threshold dimension, minimum chain cover number of bipartite graphs, and Ferrers dimension of digraphs [13,25,37]. These parameters also have $O(n^{1-\epsilon})$ approximation hardness results for $\epsilon > 0$, assuming NP $\neq$ ZPP. Further, unless NP $\subseteq$ ZPTIME$(n^{\text{poly} \log n})$, for any $\gamma > 0$ there is no $n / 2^{(\log n)^{3/4+\gamma}}$ factor approximation algorithm for any of these problems including boxicity and cubicity [10] (for more details, see Section 5.1).

Main results

1. If G is a graph on n vertices, containing a clique of size $n - k$ or more, then $\text{box}(G)$ and an optimal box representation of G can be computed in time $n^2 2^{O(k^2 \log k)}$.
2. Using the above result, we derive a polynomial time $2\left\lceil n\sqrt{\log\log n} / \sqrt{\log n} \right\rceil$ factor approximation algorithm for computing boxicity and a $2\left\lceil n(\log\log n)^{\frac{3}{2}} / \sqrt{\log n} \right\rceil$ factor approximation algorithm for computing the cubicity. To our knowledge, no approximation algorithms for approximating boxicity and cubicity of general graphs within $o(n)$ factor were known previously.
3. The above algorithms also give us the corresponding box (resp. cube) representations. As a special case, this answers the question posed by Spinrad [33] about polynomial time construction of $o(n)$ dimensional box representations for boxicity 2 graphs in the affirmative.
4. As a consequence of our $o(n)$ factor approximation algorithm for boxicity, we derive polynomial time $o(n)$ factor approximation algorithms for computing several related parameters: poset dimension, interval dimension of finite posets, minimum chain cover of bipartite graphs, Ferrers dimension of digraphs, and threshold dimension of split graphs and co-bipartite graphs. These algorithms seem to be the first $o(n)$ factor approximation algorithms known for each of these problems. We note that obtaining an $o(n)$ factor approximation algorithm for poset dimension was described as an open problem in Felsner et al. [21].
5. In the second half of this paper, we discuss a general method for obtaining some parameterized approximation algorithms for boxicity, using various edit distance parameters. A summary of the results derived using this method are given in Table 1. Notice that, if the parameter value is below $\sqrt{\log n} / \sqrt{\log\log n}$, the corresponding algorithms run in time polynomial in n, in the case of the first four parameters mentioned in the table.
6. A $(1 + \epsilon)$-factor FPT approximation scheme for computing the cubicity of graphs using vertex cover number as the parameter is derived, for any $\epsilon > 0$, by allowing the running time of the algorithm to vary depending on ϵ and k.

2. Outline

In Section 3, we discuss some basic properties of boxicity which will be required in later sections. In Section 4, an algorithm for computing the boxicity and optimal box representations of graphs with large cliques is developed. This algorithm is one of the main tools that is used in deriving many of our other algorithms in this paper. In Section 5, $o(n)$ factor approximation algorithms for boxicity and cubicity are developed and some corollaries of these results are derived. In Section 6, we develop two algorithms which provide general frameworks for obtaining parameterized approximation algorithms for boxicity with vertex (edge) edit distance parameters. Various parameterized approximation algorithms for boxicity which can be derived as corollaries of these two general algorithms are also discussed in this section. In Section 7, we discuss parameterized approximations for cubicity, using minimum vertex cover number as the parameter.

3. Prerequisites

In this section, we give some basic facts necessary for the later part of this paper. For a vertex $v \in V$ of a graph G, we use $N_G(v)$ to denote the set of neighbors of v in G. We use $G[S]$ to denote the induced subgraph of $G(V, E)$ on the vertex set $S \subseteq V$. Unless specified otherwise, graphs discussed in this paper are assumed to be undirected and without self-loops or parallel edges. The *intersection of two graphs* $G_1(V, E_1)$, $G_2(V, E_2)$ is the graph with vertex set V and edge set $E_1 \cap E_2$ and is denoted as $G_1 \cap G_2$. Similarly, the *union of two graphs* $G_1(V, E_1)$, $G_2(V, E_2)$ is the graph with vertex set V and edge set $E_1 \cup E_2$ and is denoted as $G_1 \cup G_2$. If $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$ are two graphs with $V_1 \subseteq V_2$ and $E_1 \subseteq E_2$, then G_1 is a *subgraph* of G_2 and G_2 is a *supergraph* of G_1. We use the notation $G_1 \subseteq G_2$ to denote G_1 is a subgraph of G_2 and use $G_1 \supseteq G_2$ to denote G_1 is a supergraph of G_2.

If I is an interval representation of an interval graph $G(V, E)$, we use $l_v(I)$ and $r_v(I)$, respectively, to denote the left and right end points of the interval corresponding to $v \in V$ in I. The interval corresponding to v is denoted as $\left[l_v(I), r_v(I)\right]$. We interchangeably use the same symbol for representing an interval graph and its interval representation, when the meaning of the usage is clear from the context. We use $\tau(n)$ to denote $n\sqrt{\log\log n}\,/\sqrt{\log n}$.

From an interval representation I of an interval graph G, it is easy to locally modify the interval end points and derive another interval representation of G in which the $2|V|$ interval end points are all distinct. Hence, every interval graph has an interval representation in which all $2|V|$ interval end points are distinct. Moreover, it is easy to observe that any interval representation of an interval graph with distinct end points induces an ordering of the $2|V|$ end points. From these observations, it can be concluded that there are at most $(2n)! = 2^{O(n \log n)}$ interval graphs on n vertices and hence, there are at most $\binom{(2n)!}{b} = 2^{O(nb \log n)}$ distinct b-dimensional box representations of a graph G on n vertices. Further, all these box representations can be enumerated in time $2^{O(nb \log n)}$ ensuring that in every box representation $\mathcal{B}$ of G enumerated, in the interval representation of each interval graph in $\mathcal{B}$, the $2n$ interval end points are distinct [4, Proposition 1]. This enumeration gives a brute force method to obtain an optimal box representation of G. In linear time, it is also possible to check whether a given graph is a unit interval graph and if so, generate a unit interval representation of it [5]. Hence, we can generalize [4, Proposition 1] as stated below.

Proposition 1. *If $G(V, E)$ is a graph on n vertices and boxicity b, an optimal box representation of G can be computed in $2^{O(nb \log n)}$ time, ensuring that in the interval representation of each interval graph in the box representation, the $2n$ interval end points are all distinct. If $G(V, E)$ is a graph on n vertices and cubicity b, an optimal cube representation of G can be computed in $2^{O(nb \log n)}$ time.*

Lemma 1 (*Roberts [29]*). *Let $G(V, E)$ be any graph. For any $x \in V$, $\mathrm{box}(G) \leq 1 + \mathrm{box}(G \setminus \{x\})$.*

The following lemma is based on a well-known technique of producing box representations.

Lemma 2. *Let $G(V, E)$ be a graph on n vertices. Let $S \subseteq V$ be such that $\forall v \in V \setminus S$ and $u \in V$ such that $u \neq v$, $(u, v) \in E$. If a k-dimensional box representation $\mathcal{B}_S$ of $G[S]$ is known, then, in $O(kn)$ time we can construct a box representation $\mathcal{B}$ of G of dimension $|\mathcal{B}_S|$. Moreover, $\mathrm{box}(G) = \mathrm{box}(G[S])$.*

We include a proof of this lemma, for the sake of completeness.

Proof. Let $\mathcal{B}_S = \{I_1, I_2, \ldots, I_p\}$ be a box representation of $G[S]$. For $1 \leq i \leq p$, let $l_i = \min_{u \in S} l_u(I_i)$ and $r_i = \max_{u \in S} r_u(I_i)$. For $1 \leq i \leq p$ let I'_i be the interval graph on vertex set V obtained by assigning to each vertex $v \in V$ the interval

$$\left[l_v(I'_i), r_v(I'_i)\right] = \begin{cases} [l_v(I_i), r_v(I_i)] & \text{if } v \in S, \\ [l_i, r_i] & \text{if } v \in V \setminus S. \end{cases}$$

It is easy to see that $\mathcal{B}_2 = \{I'_1, I'_2, \ldots, I'_p\}$ is a box representation of G and $\mathrm{box}(G) \leq \mathrm{box}(G[S])$. Since $G[S]$ is an induced subgraph of G, we also have $\mathrm{box}(G) \geq \mathrm{box}(G[S])$. The whole construction can be done in $O(kn)$ time. □

Lemma 3. *Let $G(V, E)$ be a graph on n vertices and let $A \subseteq V$. Let $G_1(V, E_1)$ be a supergraph of G with $E_1 = E \cup \{(x, y) \mid x, y \in A, x \neq y\}$. If a box representation $\mathcal{B}$ of G is known, then in $O(n|\mathcal{B}|)$ time we can construct a box representation $\mathcal{B}_1$ of G_1 of dimension $2|\mathcal{B}|$. In particular, $\mathrm{box}(G_1) \leq 2\,\mathrm{box}(G)$.*

Proof. Let $\mathcal{B} = \{I_1, I_2, \ldots, I_b\}$ be a box representation of G. For each $1 \le i \le b$, let $l_i = \min_{u \in V} l_u(I_i)$ and $r_i = \max_{u \in V} r_u(I_i)$. For $1 \le i \le b$, let I_{i_1} be the interval graph on vertex set V obtained by assigning to each vertex $v \in V$ the interval

$$\left[l_v(I_{i_1}), r_v(I_{i_1})\right] = \begin{cases} [l_i, r_v(I_i)] & \text{if } v \in A, \\ [l_v(I_i), r_v(I_i)] & \text{if } v \in V \setminus A, \end{cases}$$

and let I_{i_2} be the interval graph on vertex set V obtained by assigning to each vertex $v \in V$ the interval

$$\left[l_v(I_{i_2}), r_v(I_{i_2})\right] = \begin{cases} [l_v(I_i), r_i] & \text{if } v \in A, \\ [l_v(I_i), r_v(I_i)] & \text{if } v \in V \setminus A. \end{cases}$$

It is easy to see that this construction can be done in $O(nb)$ time. For an illustration of the above construction, refer to Fig. 1.

Note that, in constructing I_{i_1} and I_{i_2} we have only extended some of the intervals of I_i and therefore, I_{i_1} and I_{i_2} are supergraphs of I_i and in turn of G. By construction, A induces cliques in both I_{i_1} and I_{i_2}, and thus they are supergraphs of G_1 too.

Now, consider $(u, v) \notin E$ with $u \in V \setminus A$, $v \in A$. Then $\exists i \in \{1, 2, \ldots, b\}$ such that either $r_v(I_i) < l_u(I_i)$ or $r_u(I_i) < l_v(I_i)$. If $r_v(I_i) < l_u(I_i)$, then clearly the intervals $[l_i, r_v(I_i)]$ and $[l_u(I_i), r_u(I_i)]$ do not intersect and thus $(u, v) \notin E(I_{i_1})$. Similarly, if $r_u(I_i) < l_v(I_i)$, then $(u, v) \notin E(I_{i_2})$. If both $u, v \in V \setminus A$ and $(u, v) \notin E$, then $\exists i$ such that $(u, v) \notin E(I_i)$ for some $1 \le i \le b$ and clearly by construction, $(u, v) \notin E(I_{i_1})$ and $(u, v) \notin E(I_{i_2})$.

It follows that $G_1 = \bigcap_{1 \le i \le b} I_{i_1} \cap I_{i_2}$ and $\mathcal{B}_1 = \{I_{1_1}, I_{1_2}, I_{2_1}, I_{2_2}, \ldots, I_{b_1}, I_{b_2}\}$ is a box representation of G_1 of dimension $2b$. If $|\mathcal{B}| = \operatorname{box}(G)$ to start with, then we get $|\mathcal{B}'| \le 2\operatorname{box}(G)$. Therefore, $\operatorname{box}(G_1) \le 2\operatorname{box}(G)$. □

If $S \subseteq V$ induces a clique in G, then it is easy to see that the intersection of all the intervals in I corresponding to vertices of S is nonempty. This property is referred to as the *Helly property of intervals* and we refer to this common region of intervals as the *Helly region* of the clique S.

Definition 1. Let $G(V, E)$ be a graph in which $S \subseteq V$ induces a clique in G. Let $H(V, E')$ be an interval supergraph of G. Let p be a point on the real line. If H has an interval representation I satisfying the following conditions:

(1) p belongs to the Helly region of S in I.
(2) The end points of intervals corresponding to vertices of $V \setminus S$ are all distinct in I.
(3) For each $v \in S$,

$$l_v(I) = \min\left(p, \min_{u \in N_G(v) \cap (V \setminus S)} r_u(I)\right) \text{ and}$$
$$r_v(I) = \max\left(p, \max_{u \in N_G(v) \cap (V \setminus S)} l_u(I)\right)$$

then we call I a *nice interval representation of* H *with respect to* S *and* p. If H has a nice interval representation with respect to clique S and some point p, then H is called a *nice interval supergraph of* G *with respect to clique* S.

Fig. 2 shows some nice interval supergraphs of a graph along with their corresponding nice interval representations.

Lemma 4. *Let $G(V, E)$ be a graph in which $S \subseteq V$ induces a clique in G. For every interval supergraph H of G, we can derive a graph H' such that $H \supseteq H' \supseteq G$ and H' a nice interval supergraph of G with respect to S.*

Proof. Without loss of generality, we can assume that all $2|V|$ interval end points are distinct in (the interval representation of) H. (Otherwise, we can always alter the end points locally and make them distinct.) Let $p \in \mathbb{R}$ be a point belonging to the Helly region corresponding to S in H. Let H' be the interval graph on vertex set V obtained by assigning to each vertex $v \in V$ the interval

$$\left[l_v(H'), r_v(H')\right] = \begin{cases} [l_v(H), r_v(H)] & \text{if } v \in V \setminus S, \\ [l'_v, r'_v] & \text{if } v \in S, \end{cases}$$

where $l'_v = \min\left(p, \min_{u \in N_G(v) \cap (V \setminus S)} r_u(H)\right)$ and $r'_v = \max\left(p, \max_{u \in N_G(v) \cap (V \setminus S)} l_u(H)\right)$. Note that since $l'_v \le p \le r'_v$, the interval $[l'_v, r'_v]$ is well-defined.

We claim that $H \supseteq H' \supseteq G$. Since for any vertex $v \in V$, the interval of v in H contains the interval of v in H', we have $H \supseteq H'$. It directly follows from the definition of H' that $H'[V \setminus S] = H[V \setminus S]$. For any $(u, v) \in E(G)$, with $u \in V \setminus S$ and $v \in S$, the interval of v intersects the interval of u in H', by the definition of $[l'_v, r'_v]$. Intervals of vertices of S share the common point p in H'. Thus, $H \supseteq H' \supseteq G$. Now, from the definition of H' it follows that it is a nice interval supergraph of G with respect to the clique S and point p. □

Fig. 2 gives an illustration of the above construction.

Corollary 1. *If $G(V, E)$ has a box representation $\mathcal{B}$ of dimension b and $S \subseteq V(G)$ induces a clique in G, then G also has a box representation $\mathcal{B}'$ of the same dimension, in which $\forall H' \in \mathcal{B}'$, H' is a nice interval supergraph of G with respect to S.*

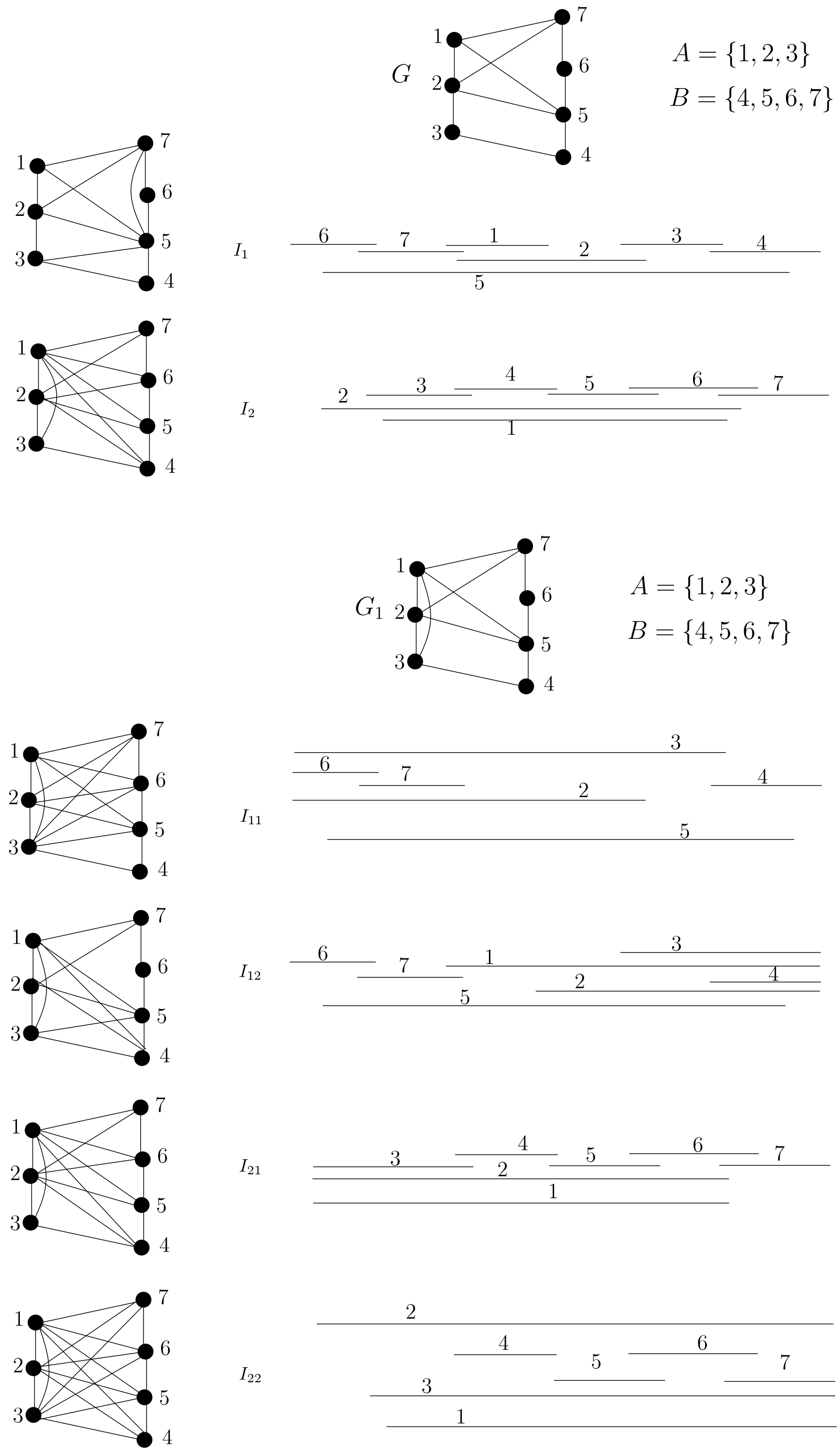

Fig. 1. A graph G, and a supergraph G_1 of G are shown. A two-dimensional box representation $\mathcal{B} = \{I_1, I_2\}$ of G and a four-dimensional box representation $\mathcal{B}' = \{I_{11}, I_{12}, I_{21}, I_{22}\}$ of G_1 derived from $\mathcal{B}$, using the construction given in the proof of Lemma 3 are also shown.

Proof. Let $\mathcal{B} = \{H_1, H_2, \ldots, H_b\}$ be a box representation of G. For each $1 \le i \le b$, let H_i' be the nice interval supergraph of G with respect to S, derived from H_i, as stated in Lemma 4. Since, by Lemma 4 we have $H_i \supseteq H_i' \supseteq G$, for each $1 \le i \le b$, it

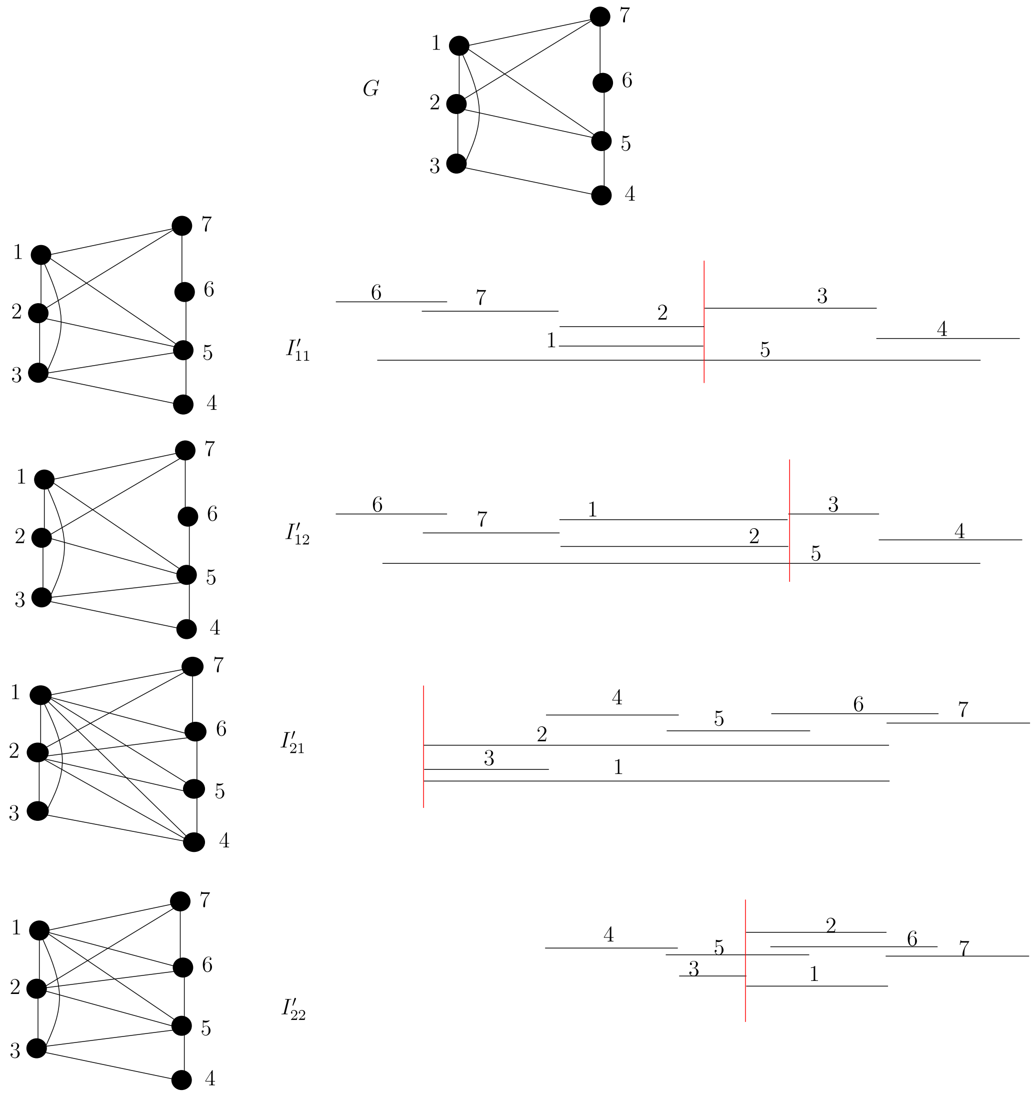

Fig. 2. $I'_{11}, I'_{12}, I'_{21}, I'_{22}$ are nice interval supergraphs of G with respect to the clique $S = \{1, 2, 3\}$. These nice interval supergraphs are, respectively, derived from the interval supergraphs $I_{11}, I_{12}, I_{21}, I_{22}$ of G given in Fig. 1 using the construction given in the proof of Lemma 4. In each interval representation, a point belonging to the Helly region of clique S is indicated by a vertical line. $\mathcal{B}' = \{I'_{11}, I'_{12}, I'_{21}, I'_{22}\}$ is a box representation of G.

follows that $\mathcal{B}' = \{H'_1, H'_2, \dots, H'_b\}$ is also a box representation of G. Notice that $\mathcal{B}'$ satisfies our requirement. An illustration of this corollary is given in Fig. 2. $\square$

Lemma 5. *Let G be a graph on n vertices, with its vertices arbitrarily labeled as $1, 2, \dots, n$. If G contains a clique of size $n-k$ or more, then:*

(a) *A subset $A \subseteq V$ such that $|A| \leq k$ and $G[V \setminus A]$ is a clique, can be computed in $O(n2^k + n^2)$ time.*
(b) *There are at most $2^{O(k \log k)}$ nice interval supergraphs of G with respect to the clique $V \setminus A$. These can be enumerated in $n^2 2^{O(k \log k)}$ time.*

Proof.

(a) We know that, if G contains a clique of size $n-k$ or more, then the complement graph $\overline{G}$ has a vertex cover of size at most k. We can compute $\overline{G}$ in $O(n^2)$ time and a minimum vertex cover A of $\overline{G}$ in $O(n2^k)$ time [27]. We have $|A| \leq k$ and $G[V \setminus A]$ is a clique because $V \setminus A$ is an independent set in $\overline{G}$.
(b) Let H be any nice interval supergraph of G with respect to $V \setminus A$. Let I be a nice interval representation of H with respect to $V \setminus A$ and a point p. Let P be the set of end points (both left and right) of the intervals corresponding to vertices of A in

H. Clearly $|P| = 2|A| \leq 2k$. The order of end points of vertices of A in I from left to right corresponds to a permutation of elements of P and therefore, there are at most $(2k)!$ possibilities for this ordering. Moreover, note that the points of P divide the real line into $|P| + 1$ regions and that p can belong to any of these regions. From the definition of nice interval representation, it is clear that, once the point p and the end points of vertices of A are fixed, the end points of vertices in $V \setminus A$ get automatically decided.

Thus, to enumerate every nice interval supergraph H of G with respect to clique $V \setminus A$, it is enough to enumerate all the $(2k)! = 2^{O(k \log k)}$ permutations of elements of P and consider $|P| + 1 \leq 2k + 1$ possible placements of p in each of them. Some of these orderings may not produce an interval supergraph of G though. In $O(n^2)$ time, we can check whether the resultant graph is an interval supergraph of G and output the interval representation. The number of supergraphs enumerated is only $(2k + 1)2^{O(k \log k)} = 2^{O(k \log k)}$. □

It may be noted that since the vertices of G are labeled in Lemma 5, we can retain the same labeling of vertices in the definition and construction of nice interval supergraphs of G, while using the method described in the proof of Lemma 5. Therefore, we have the following:

Remark 1. By construction, vertices of the nice interval supergraphs obtained by the proof of Lemma 5(b) retain their original labels as in G.

4. Boxicity of graphs with large cliques

One of the central ideas in this paper is the following theorem about computing the boxicity of graphs which contain very large cliques. Using this theorem, in Section 5 we derive $o(n)$ factor approximation algorithms for computing the boxicity and cubicity of graphs.

Theorem 1. *Let G be a graph on n vertices, containing a clique of size $n - k$ or more. Then,* $\text{box}(G) \leq k$ *and an optimal box representation of G can be found in time $n^2 2^{O(k^2 \log k)}$.*

Proof. Let $G(V, E)$ be a graph on n vertices containing a clique of size $n - k$ or more. We can assume that G is not a complete graph; otherwise, the problem becomes trivial. Arbitrarily label the vertices of G as $1, 2, \ldots, n$. Using part (a) of Lemma 5, we can compute in $O(n2^k + n^2)$ time, $A \subseteq V$ such that $|A| \leq k$ and $G[V \setminus A]$ is a clique. It is easy to infer from Lemma 1 that $\text{box}(G) \leq \text{box}(G \setminus A) + |A| = k$, since $\text{box}(G \setminus A) = 0$ by definition.

Let $\mathcal{F}$ be the family of all nice interval supergraphs of G with respect to the clique $V \setminus A$. By Corollary 1, if $\text{box}(G) = b$, then there exists a b-dimensional nice box representation of G, i.e., a box representation $\mathcal{B}' = \{I'_1, I'_2, \ldots, I'_b\}$ of G in which $I'_i \in \mathcal{F}$, for each $1 \leq i \leq b$. By part (b) of Lemma 5, $|\mathcal{F}| = 2^{O(k \log k)}$ and all graphs in $\mathcal{F}$ can be enumerated in $n^2 2^{O(k \log k)}$ time. Given an integer d, $1 \leq d \leq b$, verifying whether there exists a d-dimensional nice box representation of G, and producing if one exists, can be done in $n^2 2^{O(k \cdot d \log k)}$ time, as follows: consider every subfamily $\mathcal{F}' \subseteq \mathcal{F}$ with $|\mathcal{F}'| = d$ and check if $\mathcal{F}'$ gives a valid box representation of G (this validation is straightforward because vertices of supergraphs in $\mathcal{F}'$ retain their original labels as in G, as explained in Remark 1). We might have to repeat this process for $1 \leq d \leq b$ in that order, to identify the optimum dimension b. Hence the total time required to compute an optimal box representation of G is $bn^2 2^{O(k \cdot b \log k)}$, which is $n^2 2^{O(k^2 \log k)}$, because $b \leq k$ by the first part of this theorem. □

5. Approximation algorithms for computing boxicity and cubicity

In this section, we use Theorem 1 and derive an $o(n)$ factor approximation algorithms for boxicity and cubicity. Let $G(V, E)$ be the given graph with $|V| = n$. Without loss of generality, we can assume that G is connected. Recall the notation, $\tau(n) = n\sqrt{\log \log n}/\sqrt{\log n}$. Let $k = \lceil n/\tau(n) \rceil$ and $t = \lceil \tau(n) \rceil \geq \left\lceil \frac{n}{k} \right\rceil$. The algorithm proceeds by defining t supergraphs of G and computing their optimal box representations. Let the vertex set V be partitioned arbitrarily into t sets $V_1, V_2, \ldots, V_t$ where $|V_i| \leq k$, for each $1 \leq i \leq t$. We define supergraphs $G_1, G_2, \ldots, G_t$ of G with $G_i(V, E_i)$ defined by setting $E_i = E \cup \{(x, y) \mid x, y \in V \setminus V_i \text{ and } x \neq y\}$, for $1 \leq i \leq t$.

Lemma 6. *Let G_i be as defined above, for $1 \leq i \leq t$. An optimal box representation $\mathcal{B}_i$ of G_i can be computed in $n^{O(1)}$ time, where $n = |V|$.*

Proof. Noting that $G[V \setminus V_i]$ is a clique and $|V_i| \leq k$, by Theorem 1, we can compute an optimal box representation $\mathcal{B}_i$ of G_i in $n^2 2^{O(k^2 \log k)}$ time, where $n = |V|$. By the definition of k, we have $n^2 2^{O(k^2 \log k)} = n^{O(1)}$. □

Lemma 7. *Let $\mathcal{B}_i$ be as computed above, for $1 \leq i \leq t$. Then, $\mathcal{B} = \bigcup_{1 \leq i \leq t} \mathcal{B}_i$ is a valid box representation of G such that $|\mathcal{B}| \leq t' \,\text{box}(G)$, where t' is $2\lceil \tau(n) \rceil$. The box representation $\mathcal{B}$ is computable in $n^{O(1)}$ time.*

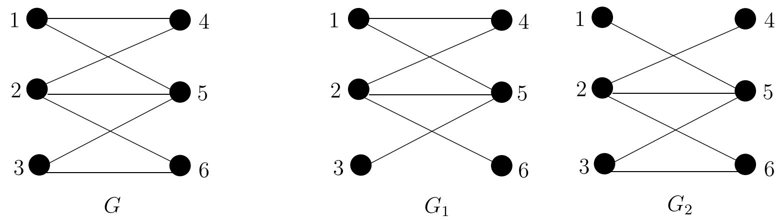

Fig. 3. A bipartite graph G and its chain cover.

Proof. We can compute optimal box representations $\mathcal{B}_i$ of G_i, for $1 \le i \le t = \lceil \tau(n) \rceil$ as explained in Lemma 6 in total $n^{O(1)}$ time. Observe that $E(G) = E(G_1) \cap E(G_2) \cap \cdots \cap E(G_t)$. Therefore, it is a trivial observation that the union $\mathcal{B} = \bigcup_{1 \le i \le t} \mathcal{B}_i$ gives us a valid box representation of G.

We will prove that this representation gives the approximation ratio as required. By Lemma 3 we have, $|\mathcal{B}_i| = \mathrm{box}(G_i) \le 2\,\mathrm{box}(G)$. Hence, $|\mathcal{B}| = \sum_{i=1}^{t} |\mathcal{B}_i| \le 2t\,\mathrm{box}(G)$. □

The box representation $\mathcal{B}$ obtained from Lemma 7 can be extended to a cube representation $\mathcal{C}$ of G as stated in the following lemma.

Lemma 8. *A cube representation $\mathcal{C}$ of G, such that $|\mathcal{C}| \le t'\,\mathrm{cub}(G)$, where t' is $2\lceil \tau(n) \log\log n \rceil$, can be computed in $n^{O(1)}$ time.*

Proof. We can compute optimal box representations $\mathcal{B}_i$ of G_i, for $1 \le i \le t = \lceil \tau(n) \rceil$ as explained in Lemma 6 in $O(n^4)$ time. By [3, Corollary 1] we know that, from the optimal box representation $\mathcal{B}_i$ of G_i we can construct a cube representation $\mathcal{C}_i$ of G_i of dimension $\mathrm{box}(G_i)\lceil \log \alpha(G_i) \rceil$, where $\alpha(G_i)$ is the independence number of G_i which is at most $|V_i|$. The construction of cube representation given in [3] can be done in $n^{O(1)}$ time.

(Recall the assumption that G is connected.)

It is easy to see that $\mathcal{C} = \bigcup_{1 \le i \le t} \mathcal{C}_i$ gives us a valid cube representation of G. We will prove that this cube representation gives the approximation ratio as required. We have,

$$|\mathcal{C}| = \sum_{i=1}^{t} |\mathcal{C}_i| \le \sum_{i=1}^{t} |\mathcal{B}_i| \lceil \log \alpha(G_i) \rceil \le \sum_{i=1}^{t} |\mathcal{B}_i| \lceil \log k \rceil \le 2t\,\mathrm{box}(G) \log\log n \le 2t \log\log n\,\mathrm{cub}(G). \quad \square$$

Combining Lemmas 7 and 8, we get the following theorem which gives $o(n)$ factor approximation algorithms for computing boxicity and cubicity.

Theorem 2. *Let $G(V, E)$ be a graph on n vertices. Then a box representation $\mathcal{B}$ of G, such that $|\mathcal{B}| \le t\,\mathrm{box}(G)$, where t is $2\lceil \tau(n) \rceil$, can be computed in polynomial time. Further, a cube representation $\mathcal{C}$ of G, such that $|\mathcal{C}| \le t'\,\mathrm{cub}(G)$, where t' is $2\lceil \tau(n) \log\log n \rceil$, can also be computed in polynomial time.*

5.1. Consequences of Theorem 2

Now, we describe how Theorem 2 can be used to derive sublinear approximation algorithms for some well-known problems whose computational complexity is closely related to that of boxicity.

Chain cover of bipartite graphs. A bipartite graph is a *chain graph*, if it does not contain an induced matching of size 2. Given a bipartite graph $G(V, E)$, the *minimum chain cover number* of G, denoted by $\mathrm{ch}(G)$ is the smallest number of chain graphs on the vertex set V such that the union of their edge sets is $E(G)$. Fig. 3 shows a bipartite graph G and two chain graphs G_1 and G_2 such that $E(G_1) \cup E(G_2) = E(G)$. Therefore, $\mathrm{ch}(G) \le 2$. Since G contains an induced matching of size 2, $\mathrm{ch}(G) > 1$ and from this, it follows that $\mathrm{ch}(G) = 2$.

It is well-known that G is a chain graph if and only if its complement is a co-bipartite interval graph [37, Lemma 4]. From this, it immediately follows that $\mathrm{ch}(G) = \mathrm{box}(\overline{G})$ [37, Lemma 4]. For an illustration of this fact, see Figs. 3 and 4.

Corollary 2. *There is a polynomial time $2\lceil \tau(n) \rceil$ factor approximation algorithm to compute the minimum chain cover number of an n-vertex bipartite graph.*

Threshold dimension of split graphs. The concept of threshold graphs and threshold dimension was introduced by Chvátal and Hammer [14] while studying some set-packing problems. A graph $G(V, E)$ is called a *threshold graph* if there exists $s \in \mathbb{R}$ and a labeling of vertices $w : V \mapsto \mathbb{R}$ such that $\forall u, v \in V, (u, v) \in E \Leftrightarrow w(u) + w(v) \ge s$. Threshold graphs are also characterized

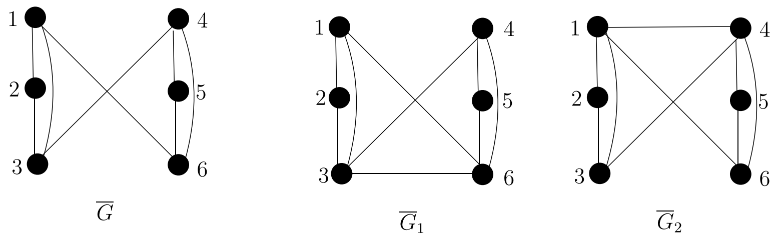

Fig. 4. Co-bipartite graphs $\overline{G}$, $\overline{G}_1$, $\overline{G}_2$ are the respective complements of graphs G, G_1, G_2 of Fig. 3. It is easy to verify that $\overline{G}_1$, $\overline{G}_2$ are interval graphs and $\{\overline{G}_1, \overline{G}_2\}$ is a box representation of $\overline{G}$.

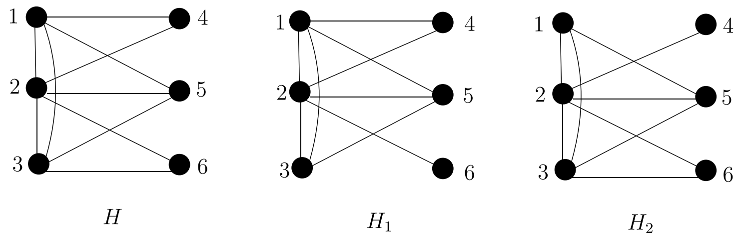

Fig. 5. A split graph H and its threshold cover. Note that bipartite graphs G, G_1, and G_2 of Fig. 3 are obtained, respectively, from split graphs H, H_1 and H_2, by converting the clique side of the split graphs into independent sets.

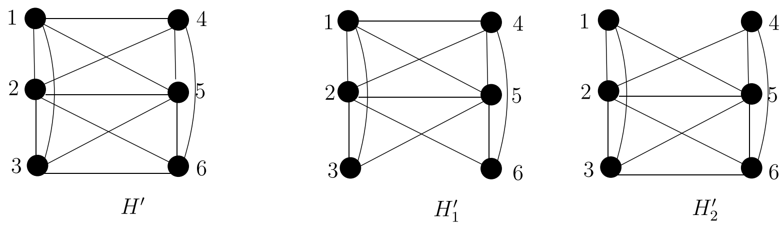

Fig. 6. A co-bipartite graph H' and its threshold cover. Note that split graphs H, H_1, and H_2 of Fig. 5 are obtained, respectively, from co-bipartite graphs H', H'_1 and H'_2, by converting one of the clique sides of the split graphs into an independent set.

as graphs which have neither an induced matching of size 2, nor an induced path on 4 vertices nor an induced cycle on 4 vertices [14].

The *threshold dimension* of G, denoted by $t(G)$ is the minimum number of threshold subgraphs required to cover $E(G)$. Fig. 5 shows a graph H and two threshold graphs H_1 and H_2 that cover $E(H)$. Since H contains an induced path on 4 vertices, H is not a threshold graph and therefore $t(H) = 2$. Even for split graphs, threshold dimension is hard to approximate within an $O(n^{1-\epsilon})$ factor for any $\epsilon > 0$, unless NP $=$ ZPP [10,25].

Corollary 3. *There is a polynomial time $2\lceil \tau(n) \rceil$ factor approximation algorithm to compute the threshold dimension of any split graph on n vertices.*

Proof. Given any split graph G, there is a polynomial time method to construct a bipartite graph H on the same vertex set such that $t(G) = \mathrm{ch}(H)$ [25, Page 149, Lemma 7.3.4]. From the approximation algorithm for computing $\mathrm{ch}(H)$, the result follows. □

Threshold dimension of co-bipartite graphs. Cozzens et al. [16] showed that if G is a co-bipartite graph, an associated split graph G' on the same vertex set can be constructed in polynomial time, such that for any $k \geq 2$, $t(G) \leq k$ if and only if $t(G') \leq k$ (see Fig. 6).

This reduction shows that the hardness result of threshold dimension of split graphs is also applicable for the threshold dimension of co-bipartite graphs. Moreover, we get the following.

Corollary 4. *There is a polynomial time $2\lceil \tau(n) \rceil$ factor approximation algorithm to compute the threshold dimension of any co-bipartite graph on n vertices.*

Partial order dimension. This concept was introduced by Dushnik and Miller in 1941 [17]. A *partially ordered set (poset)* $\mathcal{P} = (X, P)$ consists of a nonempty set X and a binary relation P on X that is reflexive, antisymmetric and transitive. If every pair of

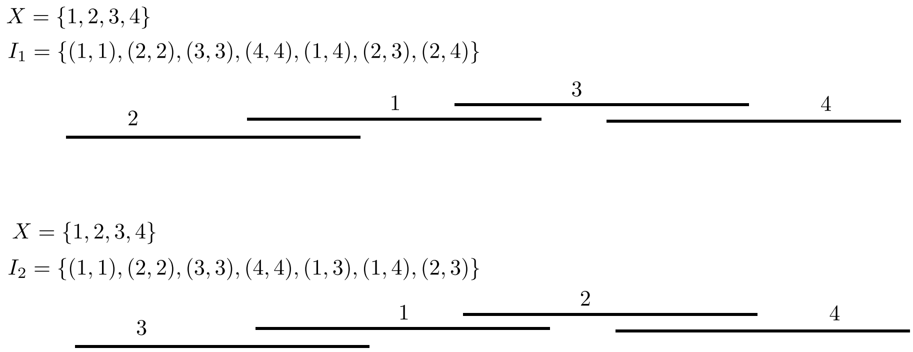

Fig. 7. Interval realization of two interval extensions (X, I_1) and (X, I_2) of the poset (X, P), where $X = \{1, 2, 3, 4\}$ and $P = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 4), (2, 3)\}$.

distinct elements of X are comparable under the relation P, then (X, P) is called a *total order* or a *linear order*. A *linear extension* of a partial order (X, P) is a linear order (X, P') such that $\forall x, y \in X, (x, y) \in P \Rightarrow (x, y) \in P'$. For example, if $X = \{1, 2, 3, 4\}$, $P = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 2), (3, 4)\}$, $P_1 = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (3, 4)\}$ and $P_2 = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 2), (3, 1), (4, 1), (3, 2), (4, 2), (3, 4)\}$, then (X, P) is a partial order and $(X, P_1), (X, P_2)$ are its linear extensions.

The *dimension of a poset* $\mathcal{P} = (X, P)$, denoted by $\dim(\mathcal{P})$ is defined as the smallest integer k such that $\mathcal{P}$ can be expressed as the intersection of k linear extensions $(X, P_1), (X, P_2), \ldots, (X, P_k)$ of $\mathcal{P}$: i.e., if $\forall x, y \in X, (x, y) \in P \Leftrightarrow (x, y) \in P_i$, for each $1 \leq i \leq k$. In the example given in previous paragraph, the partial order (X, P) is not a total order; however, it is the intersection of total orders (X, P_1) and (X, P_2). Therefore, $\dim((X, P)) = 2$.

A *height-two poset* is a poset (X, P) in which all elements of X are either minimal elements or maximal elements under the relation P. Even in the case of height-two posets, partial order dimension is hard to approximate within an $O(n^{1-\epsilon})$ factor for any $\epsilon > 0$, unless NP = ZPP [10]. A height-two poset $\mathcal{P} = (X, P)$ in which X_1 is the set of minimal elements and X_2 is the set of maximal elements can be associated with a bipartite graph $B(\mathcal{P})$ with vertex set X and edge set given by $\{(x, y) : x \in X_1, y \in X_2, (x, y) \notin P\}$ [25, Page 147].

Corollary 5. *There is a polynomial time $O(\tau(n))$ factor approximation algorithm to compute the partial order dimension of a poset $\mathcal{P} = (X, P)$ defined on an n-element set X.*

Proof. Let $\mathcal{P} = (X, P)$ be a poset with $|X| = n$. By a construction given by R. Kimble [35, Theorem 5], given a poset $\mathcal{P} = (X, P)$ of arbitrary height, we can construct a height-two poset $\mathcal{P}' = (Y, P')$ from $\mathcal{P} = (X, P)$ in polynomial time so that $\dim(\mathcal{P}) \leq \dim(\mathcal{P}') \leq 1 + \dim(\mathcal{P})$ and $|Y| = 2|X|$. It is also known that $\dim(\mathcal{P}) = \mathrm{ch}(B(\mathcal{P}'))$ [37, Footnote, Page 354]. Therefore, by computing $\mathrm{ch}(B(\mathcal{P}'))$ using the algorithm given by Corollary 2, we can compute a $O(\tau(n))$ approximation of $\dim(\mathcal{P})$. $\square$

Interval dimension of posets. A poset (X, P) is an *interval order*, if each $x \in X$ can be assigned an open interval (l_x, r_x) of the real line such that $(x, y) \in P$ for $x \neq y$ if and only if $r_x \leq l_y$. Fig. 7 shows assignment of intervals to vertices of two interval orders (X, I_1) and (X, I_2).

An *interval order extension* of a partial order (X, P) is an interval order (X, P') such that $\forall x, y \in X, (x, y) \in P \Rightarrow (x, y) \in P'$. The *interval dimension* of a poset $\mathcal{P} = (X, P)$, denoted by $\mathrm{idim}(\mathcal{P})$, is defined as the smallest integer k such that $\mathcal{P}$ can be expressed as the intersection of k interval order extensions of $\mathcal{P}$. For example, consider $X = \{1, 2, 3, 4\}$ and the partial order (X, P), where $P = \{(1, 1), (2, 2), (3, 3), (4, 4), (1, 4), (2, 3)\}$. It is not difficult to see that (X, P) is not an interval order. Fig. 7 shows two interval order extensions (X, I_1) and (X, I_2) of (X, P) such that their intersection is (X, P). Therefore, $\mathrm{idim}((X, P)) = 2$. Since linear orders are interval orders, it follows that $\mathrm{idim}(\mathcal{P}) \leq \dim(\mathcal{P})$. On the other hand, the poset dimension of an interval order can be large.

Since the height-two[1] poset $\mathcal{P}'$ given by Kimble's construction [35,25] from an arbitrary finite poset $\mathcal{P}$ satisfies $\dim(\mathcal{P}) = \mathrm{ch}(B(\mathcal{P}'))$ [37, Footnote, Page 354] and $\mathrm{ch}(B(\mathcal{P}')) = \mathrm{idim}(\mathcal{P}')$ [37, Lemma 4], from the approximation hardness of poset-dimension [10], we can see that interval dimension is hard to approximate within an $O(n^{1-\epsilon})$ factor for any $\epsilon > 0$, unless NP = ZPP. Felsner et al. [20] showed that given a poset (X, P), it is possible to construct another poset (Y, P') in polynomial time, such that $|Y| = 2|X|$ and $\mathrm{idim}((X, P)) = \dim\big((Y, P')\big)$.

Corollary 6. *There is a polynomial time $O(\tau(n))$ factor approximation algorithm for computing the interval dimension of any poset $\mathcal{P} = (X, P)$ defined on a set X of n elements.*

[1] The notion of height-two posets as per our definition is the same as the notion of height-one posets in [37].

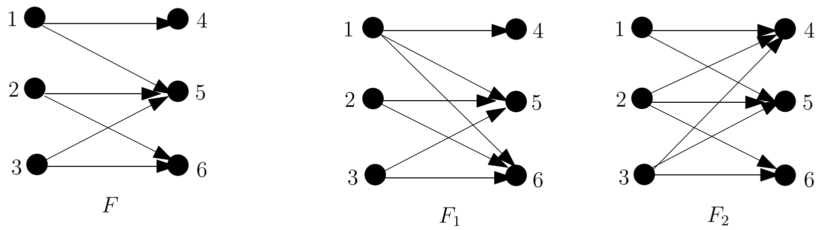

Fig. 8. A digraph F and two Ferrers digraphs F_1 and F_2 whose intersection is F.

Ferrers dimension of digraphs Ferrers relations were introduced by Riguet in 1950s [28]. A digraph $G(V, E)$ is called a *Ferrers digraph* if it does not contain vertices x, y, z, w (not necessarily distinct), satisfying $(x, y) \in E$, $(z, w) \in E$, but $(x, w) \notin E$, $(z, y) \notin E$ [25, Page 33]. The *Ferrers dimension* [15] of a digraph G is the smallest number of Ferrers digraphs whose intersection is G. Fig. 8 shows a digraph F and two Ferrers digraphs F_1 and F_2 whose intersection is F. Since vertices $\{1, 4, 2, 6\}$ in F violate the condition for a Ferrers digraph, the Ferrers dimension of F is 2.

Since a partial order $\mathcal{P}$ has $\dim(\mathcal{P})$ equal to the Ferrers dimension of its underlying digraph [15], Ferrers dimension is also hard to approximate within an $O(n^{1-\epsilon})$ factor for any $\epsilon > 0$, unless NP = ZPP. Cogis [15] showed that given a digraph $G(V, E)$, a poset $\mathcal{P} = (X, P)$ can be constructed in polynomial time, such that $|X| \leq 2|V|$ and the poset dimension of $\mathcal{P}$ is equal to the Ferrers dimension of G.

Corollary 7. *There is a polynomial time $O(\tau(n))$ factor approximation algorithm for computing the Ferrers dimension of a digraph on n vertices.*

6. Parameterized approximations for boxicity with edit distance parameters

The algorithm for finding the boxicity of graphs with large cliques obtained from Theorem 1 was used in the previous section to derive sublinear approximation algorithms for boxicity and cubicity problems and also for some related dimensional parameters. In this section, we will derive some parameterized approximation algorithms for boxicity using Theorem 1 and some techniques that are similar in nature to those used in the proof of Theorem 1.

A study of parameterized algorithms for boxicity was initiated by Adiga et al. [4]. Later, Bruhn et al. [7] also considered various structural parameterizations of boxicity (which appeared after the initial version of this article [1]).

A framework for parameterizing problems with edit distance as the parameter was introduced by Cai [8]. For a family $\mathcal{F}$ of graphs, and $k \geq 0$ an integer, Cai used $\mathcal{F} + ke$ (respectively, $\mathcal{F} - ke$) to denote the family of graphs that can be converted to a graph in $\mathcal{F}$ by deleting (respectively, adding) at most k edges, and $\mathcal{F} + kv$ to denote the family of graphs that can be converted to a graph in $\mathcal{F}$ by deleting at most k vertices. This framework was used by Cai [8], for studying the parameterized complexity of the vertex coloring problem on $\mathcal{F} - ke$, $\mathcal{F} + ke$ and $\mathcal{F} + kv$ graphs, with k as the parameter, for various families of graphs $\mathcal{F}$.

In a similar way, we consider the parameterized complexity of computing boxicity of $\mathcal{F} + k_1e - k_2e$ and $\mathcal{F} + kv$ graphs for families $\mathcal{F}$ of bounded boxicity graphs, using $k_1 + k_2$ (the edge edit distance) and k (the vertex edit distance) as parameters. Note that, parameters like interval completion number, minimum vertex cover size and minimum feedback vertex set size are examples of vertex edit distance parameters from some bounded boxicity graph families (respectively, consider $\mathcal{F}$ to be the family of interval graphs, the family of graphs with no edges, and the family of trees).

A subset $S \subseteq V$ such that $|S| \leq k$ is called a **modulator** for an $\mathcal{F} + kv$ graph $G(V, E)$ if $G \setminus S \in \mathcal{F}$. Similarly, a set E_k of pairs of vertices such that $|E_k| \leq k$ is called a modulator for an $\mathcal{F} - ke$ graph $G(V, E)$ if $G'(V, E \cup E_k) \in \mathcal{F}$. Modulators for graphs in $\mathcal{F} + ke$ and $\mathcal{F} + k_1e - k_2e$ are defined in a similar manner.

The following theorem gives us a parameterized algorithm for computing the boxicity of $\mathcal{F} + kv$ graphs. The proof of this theorem uses Theorem 1.

Theorem 3. *Let $\mathcal{F}$ be a family of graphs such that $\forall G' \in \mathcal{F}$, $\mathrm{box}(G') \leq b \leq n$. Let $T(n)$ denote the time required to compute a b-dimensional box representation of a graph belonging to $\mathcal{F}$ on n vertices. Let G be an $\mathcal{F} + kv$ graph on n vertices. Given a modulator of G, a box representation $\mathcal{B}$ of G, such that $|\mathcal{B}| \leq 2\,\mathrm{box}(G) + b$ can be computed in time $T(n-k) + n^2 2^{O(k^2 \log k)}$.*

Proof. Let $\mathcal{F}$ be the family of graphs of boxicity at most b. Let $G(V, E)$ be an $\mathcal{F} + kv$ graph on n vertices, with a modulator S_k on k vertices such that $G' = G \setminus S_k \in \mathcal{F}$. We define two supergraphs of G, namely $H_1(V, E_1)$ and $H_2(V, E_2)$ such that $E = E_1 \cap E_2$ with $\mathrm{box}(H_1) \leq 2\,\mathrm{box}(G)$, $\mathrm{box}(H_2) \leq b$ and their required valid box representations are computable within the time specified in the theorem. It is easy to see that the union of valid box representations of H_1 and H_2 will be a valid box representation $\mathcal{B}$ of G and hence $|\mathcal{B}| \leq \mathrm{box}(H_1) + \mathrm{box}(H_2) \leq 2\,\mathrm{box}(G) + b$. This will complete our proof of Theorem 3.

We define H_1 to be the graph obtained from G by making $V \setminus S_k$ a clique on $n-k$ vertices, without altering other adjacencies in G. Formally, $E_1 = E \cup \{(x, y) \mid x, y \in V \setminus S_k, x \neq y\}$. Using Theorem 1, we can get an optimal box representation $\mathcal{B}_1$ of H_1 in $n^2 2^{O(k^2 \log k)}$ time. By Lemma 3, $|\mathcal{B}_1| \leq 2\,\mathrm{box}(G)$.

We define H_2 to be the graph obtained from G by making each vertex in S_k adjacent to every other vertex in the graph and leaving other adjacencies in G unaltered. Formally, $E_2 = E \cup \{(x, y) \mid x \in S_k, y \in V, x \neq y\}$. Let $\mathcal{B}'$ be a box representation of G' of dimension at most b (computed in time $T(n-k)$). Then, $\mathcal{B}'$ is a box representation of $H_2[V \setminus S_k]$ as well, because $H_2[V \setminus S_k] = G'$. By Lemma 2, $\mathrm{box}(H_2) = \mathrm{box}(H_2[V \setminus S_k])$ and a box representation $\mathcal{B}_2$ of H_2 of dimension at most $|\mathcal{B}'| \leq b$ can be produced in $O(n^2)$ time.

Since $G = H_1 \cap H_2$, $\mathcal{B} = \mathcal{B}_1 \cup \mathcal{B}_2$ is a valid box representation of G, of dimension at most $2\,\mathrm{box}(G) + b$. All computations were done in $T(n-k) + n^2 2^{O(k^2 \log k)}$ time. □

Using a similar method, we also get a parameterized approximation algorithm for computing the boxicity of $\mathcal{F} + k_1 e - k_2 e$ graphs.

Theorem 4. *Let $\mathcal{F}$ be a family of graphs such that $\forall G' \in \mathcal{F}$, $\mathrm{box}(G') \leq b \leq n$. Let $T(n)$ denote the time required to compute a b-dimensional box representation of a graph belonging to $\mathcal{F}$ on n vertices. Let G be an $\mathcal{F} + k_1 e - k_2 e$ graph on n vertices and let $k = k_1 + k_2$. Given a modulator of G, a box representation $\mathcal{B}$ of G, such that $|\mathcal{B}| \leq \mathrm{box}(G) + 2b$, can be computed in time $T(n) + O(n^2) + 2^{O(k^2 \log k)}$.*

Proof. Let $\mathcal{F}$ be the family of graphs of boxicity at most b. Let $G(V, E)$ be an $\mathcal{F} + k_1 e - k_2 e$ graph on n vertices, where $k_1 + k_2 = k$. Let $E_{k_1} \cup E_{k_2}$ be a modulator of G such that $|E_{k_1}| = k_1$, $|E_{k_2}| = k_2$ and $G'(V, \left(E \cup E_{k_2}\right) \setminus E_{k_1}) \in \mathcal{F}$. Let $T \subseteq V(G)$ be the set of vertices incident with edges in $E_{k_1} \cup E_{k_2}$.

As in the proof of Theorem 3, we define two supergraphs of G, namely $H_1(V, E_1)$ and $H_2(V, E_2)$ such that $E = E_1 \cap E_2$ with $\mathrm{box}(H_1) \leq 2b$, $\mathrm{box}(H_2) \leq \mathrm{box}(G)$ and their required valid box representations are computable within the time specified in the theorem. As earlier, the union of valid box representations of H_1 and H_2 will be a valid box representation of $\mathcal{B}$ of G and hence $|\mathcal{B}| \leq \mathrm{box}(H_1) + \mathrm{box}(H_2) \leq 2b + \mathrm{box}(G)$. This will complete our proof of Theorem 4.

Let $H_1(V, E_1)$ be the graph obtained from G' by making T a clique, without altering other adjacencies in G'. Formally, $E_1 = E' \cup \{(x, y) | x, y \in T, x \neq y\}$. Let $\mathcal{B}'$ be a box representation of G' of dimension at most b computed in time $T(n)$. From the box representation $\mathcal{B}'$ of G', in $O(b \cdot n)$=$O(n^2)$ time we can construct (by Lemma 3) a box representation $\mathcal{B}_1$ of H_1 with dimension $2b$.

Let $H_2(V, E_2)$ be the graph obtained from G by making each vertex in $V \setminus T$ adjacent to every other vertex in the graph and leaving other adjacencies in G unaltered. Formally, $E_2 = E \cup \{(x, y) | x \in V \setminus T, y \in V, x \neq y\}$. Clearly, $|T| \leq 2k$ and therefore, using the construction in Proposition 1, an optimal box representation $\mathcal{B}_T$ of $H_2[T]$ can be computed in $2^{O(k^2 \log k)}$ time. By Lemma 2, $\mathrm{box}(H_2) = \mathrm{box}(H_2[T])$ and a box representation $\mathcal{B}_2$ of H_2 of dimension $\mathrm{box}(H_2[T])$ can be computed from the box representation $\mathcal{B}_T$ of $H_2[T]$ in $O(n^2)$ time. Observe that $H_2[T] = G[T]$. Therefore, $|\mathcal{B}_2| = \mathrm{box}(G[T]) \leq \mathrm{box}(G)$, because $G[T]$ is an induced subgraph of G.

Since $G = H_1 \cap H_2$, $\mathcal{B} = \mathcal{B}_1 \cup \mathcal{B}_2$ is a valid box representation of G, of dimension at most $\mathrm{box}(G) + 2b$. All computations were done in $T(n) + O(n^2) + 2^{O(k^2 \log k)}$ time. □

Remark 2. Though in Theorems 3 and 4 it is assumed that a modulator of G for $\mathcal{F}$ is given, in several important special cases (as in the case of corollaries discussed below), the modulator for $\mathcal{F}$ can be computed from G in FPT time. Moreover, in those cases, $T(n)$ is a polynomial in n. Thus, the algorithms given by Theorems 3 and 4 turns out to be FPT approximation algorithms for boxicity.

6.1. Corollaries of Theorems 3 and 4

A general approach for obtaining parameterized approximation algorithms for the boxicity of $\mathcal{F} + k_1 e - k_2 e$ and $\mathcal{F} + kv$ graphs for families $\mathcal{F}$ of bounded boxicity graphs are given by Theorems 3 and 4. Parameterized approximation algorithms for boxicity with various vertex and edge edit distance parameters can be derived using the above approach.

Given an input graph G and a parameter of interest, the general procedure is as follows:

(i) Use known FPT algorithms to compute the parameter of interest for the input graph G and obtain from G a modulator S_k for the corresponding family $\mathcal{F}$.
(ii) Compute a low dimensional box representation for the graph $G' = (G \setminus S_k) \in \mathcal{F}$, in polynomial time.
(iii) Use our algorithm of Theorem 3 (Theorem 4) to derive an FPT approximation algorithm for computing boxicity of G using the parameter of interest.

Some corollaries of Theorem 3 are discussed below.

Corollary 8. FVS *as the parameter: The minimum number of vertices to be deleted from a graph G so that the resultant graph is acyclic is called the feedback vertex set size* (FVS) *of G. If* $\mathrm{FVS}(G) \leq k$, *we get a* $\left(2 + \frac{2}{\mathrm{box}(G)}\right)$ *factor approximation for boxicity with* FVS *as the parameter k, which runs in time* $2^{O(k^2 \log k)} n^{O(1)}$.

Proof. If $\mathrm{FVS}(G) \leq k$, using existing FPT algorithms [9], in $O(3.83^k kn^2)$ time we can compute a minimum feedback vertex set S of $G(V, E)$ such that $G' = G(V \setminus S)$ is a forest. Thus, with modulator S, $G \in \mathcal{F} + kv$, where $\mathcal{F}$ is the family of graphs which are forests. Since a box representation of dimension two can be computed in polynomial time for any forest [32], using our algorithm of Theorem 3, we get a $2 + \frac{2}{\mathrm{box}(G)}$ factor approximation for boxicity with FVS as the parameter k, which runs in time $2^{O(k^2 \log k)} n^{O(1)}$. □

Remark 3. For the boxicity problem parameterized by FVS, Adiga et al. [4] gave an algorithm with the same approximation factor as the algorithm discussed above, but with running time $2^{O(2^k k^2)} n^{O(1)}$. The running time of our algorithm is better.

Corollary 9. *Proper Interval Vertex Deletion number (*PIVD*) as the parameter: The minimum number of vertices to be deleted from the graph G, so that the resultant graph is a proper interval graph, is called* $\mathrm{PIVD}(G)$*. If* $\mathrm{PIVD}(G) \leq k$*, we get a* $2 + \frac{1}{\mathrm{box}(G)}$ *factor approximation for boxicity with* PIVD *as the parameter k, which runs in time* $2^{O(k^2 \log k)} n^{O(1)}$*.*

Proof. If $\mathrm{PIVD}(G)$ is at most k, we can use the FPT algorithm running in $O(6^k kn^6)$ time for proper interval vertex deletion [23] to compute a $S \subseteq V$ with $|S| \leq k$ such that $G \setminus S$ is a proper interval graph. Thus, with modulator S, $G \in \mathcal{F} + kv$, where $\mathcal{F}$ is the family of all proper interval graphs. Since a box representation of dimension one can be computed in polynomial time for any proper interval graph [5], using our algorithm of Theorem 3, we get a $2 + \frac{1}{\mathrm{box}(G)}$ factor approximation for boxicity with PIVD as the parameter k, which runs in time $2^{O(k^2 \log k)} n^{O(1)}$. □

Remark 4. It is easy to see that $\mathrm{PIVD}(G) \leq \mathrm{MVC}(G)$, where $\mathrm{MVC}(G)$ is the size of the minimum vertex cover number of G. Hence, $\mathrm{PIVD}(G)$ is a better parameter than the parameter $\mathrm{MVC}(G)$. This algorithm has the same running time as the additive one approximation algorithm for boxicity with $\mathrm{MVC}(G)$ as the parameter, discussed in [3].

Corollary 10. *Planar Vertex Deletion number (*PVD*) as the parameter: The minimum number of vertices to be deleted from G to make it a planar graph, is called the planar vertex deletion number of G. If* $\mathrm{PVD}(G) \leq k$*, we get an FPT algorithm for boxicity, giving a* $\left(2 + \frac{3}{\mathrm{box}(G)}\right)$ *factor approximation for boxicity using planar vertex deletion number as the parameter.*

Proof. If $G \in \text{Planar}+kv$, we can use the FPT algorithm running in $O(f(k)n^2)$ time for planar deletion [26] to compute a $S \subseteq V$ with $|S| \leq k$ such that $G \setminus S$ is planar. Thus, with modulator S, $G \in \mathcal{F} + kv$, where $\mathcal{F}$ is the family of planar graphs. Since planar graphs have 3 dimensional box representations computable in polynomial time [34], using our algorithm of Theorem 3, we get an FPT algorithm for boxicity, giving a $2 + \frac{3}{\mathrm{box}(G)}$ factor approximation for boxicity of graphs that can be made planar by deleting at most k vertices, using planar vertex deletion number as the parameter. □

Theorem 4 also gives us FPT approximation algorithms for computing boxicity with various parameters of interest.

Corollary 11. *Interval Completion number as the parameter: The minimum number of edges to be added to a graph G, so that the resultant graph is an interval graph, is called the interval completion number of G. If the interval completion number G is at most k, we get an FPT algorithm that achieves an additive 2 approximation for* $\mathrm{box}(G)$ *which runs in time* $2^{O(k^2 \log k)} n^{O(1)}$*.*

Proof. If the interval completion number of a graph $G(V, E)$ is at most k, we can use the FPT algorithm for interval completion [36] with running time $O(k^{2k} n^{O(1)}) = 2^{O(k \log k)} n^{O(1)}$ to compute E_k such that $|E_k| \leq k$ and $G'(V, E \cup E_k)$ is an interval graph. Thus, with modulator E_k, $G \in \mathcal{F} - ke$, where $\mathcal{F}$ is the class of interval graphs. Since a box representation of dimension one can be computed in polynomial time for any interval graph [5], combining with our algorithm of Theorem 4, we get an FPT algorithm that achieves an additive 2 factor approximation for $\mathrm{box}(G)$, with interval completion number k as the parameter which runs in time $2^{O(k^2 \log k)} n^{O(1)}$. □

Corollary 12. *Proper Interval Edge Deletion number (*PIED*) as the parameter: The minimum number of edges to be deleted from the graph G, so that the resultant graph is a proper interval graph, is called* $\mathrm{PIED}(G)$*. If* $\mathrm{PIED}(G)$ *is at most k, we get an FPT algorithm that achieves an additive 2 approximation for* $\mathrm{box}(G)$*, with* $\mathrm{PIED}(G)$ *as the parameter k, which runs in time* $2^{O(k^2 \log k)} n^{O(1)}$*.*

Proof. If $\mathrm{PIED}(G)$ is at most k, we can use the FPT algorithm running in $O(9^k n^{O(1)})$ time for proper interval edge deletion [23] to compute a $E_k \subseteq E$ with $|E_k| \leq k$ such that $G'(V, E \setminus E_k)$ is a proper interval graph. Thus, with modulator S, $G \in \mathcal{F} + ke$, where $\mathcal{F}$ is the family of all proper interval graphs. Since a box representation of dimension one can be computed in polynomial time for any interval graph, combining with our algorithm of Theorem 4, we get an FPT algorithm that achieves an additive 2 factor approximation for $\mathrm{box}(G)$, with PIED as the parameter k, which runs in time $2^{O(k^2 \log k)} n^{O(1)}$. □

Corollary 13. *Planar Edge Deletion number (*PED*) as the parameter: The minimum number of edges to be deleted from G so that the resultant graph is planar is called* $\mathrm{PED}(G)$*. If* $\mathrm{PED}(G) \leq k$*, we get an FPT algorithm that gives an additive 6 approximation for* $\mathrm{box}(G)$ *with* $\mathrm{PED}(G)$ *as the parameter.*

Proof. If $\mathrm{PED}(G) \leq k$, we can use the FPT algorithm for planar edge deletion [22] to compute $E_k \subseteq E$ such that $|E_k| \leq k$ and $G'(V, E \setminus E_k)$ is a planar graph. Thus, with modulator E_k, $G \in \mathcal{F} + ke$, where $\mathcal{F}$ is the class of planar graphs. Since planar graphs have 3 dimensional box representations computable in polynomial time [34], using our algorithm of Theorem 4, we get an FPT algorithm that gives an additive 6-factor approximation for $\mathrm{box}(G)$ with $\mathrm{PED}(G)$ as the parameter. □

Corollary 14. *Max Leaf number* (ML) *as the parameter: The number of the maximum possible leaves in any spanning tree of a graph G is called* $\mathrm{ML}(G)$*. If* $\mathrm{ML}(G) \leq k$*, we get an FPT algorithm that achieves an additive 2 approximation for* $\mathrm{box}(G)$ *which runs in time* $2^{O(k^3 \log k)} n^{O(1)}$.

Proof. The underlying algorithm here same as that in the proof of Theorem 4.

We assume that G is connected and the max leaf number of G is at most k. If the graph G is just a cycle on n vertices ($n \geq 3$), we know that $\mathrm{box}(G) = 1$ if $n = 3$ and $\mathrm{box}(G) = 2$ if $n > 3$. Thus, we can also assume that G is not a cycle. Moreover, the maximum degree of any vertex in G is at most k; otherwise we can start with a vertex of degree at least $k + 1$ and grow it to a spanning tree with more than k leaves, which is a contradiction.

In the proof of Theorem 4, supergraphs H_1 and H_2 were obtained by modifying a certain graph G' whose edge edit distance to G is small. Here, we will define G' in a slightly different way and then define H_1 and H_2 in a similar way as we did in the proof of Theorem 4.

We start by defining a graph G_1, such that G is a subdivision of G_1. For this, we use the following result.

Property 1 (*Fellows et al. [19]*)**.** *If the max leaf number of a graph G is at most k, then G is a subdivision of a graph* $G_1(V', F)$ *with* $|V'| \leq 4k - 2$ *and* $V' \subseteq V$*. (*G_1 *may contain multi edges and self loops.)*

Let $G_1(V', F)$ the graph given by Property 1. Since G is not a cycle, we can eliminate all degree two vertices from G_1 one by one, by edge contractions. Therefore, without loss of generality, we can assume that there are no degree two vertices in G_1 and V' is precisely the set of vertices of G whose degree is not equal to 2.

Claim 1. *There are at most* $4k - 2$ *vertices in G, whose degree is not equal to* 2.

Proof. As explained above, we assume that there are no degree two vertices in G_1. Since G is a subdivision of G_1 and a subdivision only introduces degree 2 vertices, we can conclude that there are at most $4k - 2$ vertices in G, whose degree is not equal to 2. □

Let $E_k \subseteq E$ be the set of edges of G which have at least one of its incident vertices belonging to V'. Now, we will define G' as the graph with vertex set V and edge set $E \setminus E_k$.

Claim 2. *The graph* $G'(V, E \setminus E_k)$ *is an interval graph and it can be computed in polynomial time from G.*

Proof. Since G is a subdivision of $G_1(V', F)$, it is easy to see that, the graph $G'(V, E \setminus E_k)$ is just a collection of vertex disjoint paths and isolated vertices. It is straightforward to verify that G' is an interval graph. To compute G', we just need to compute E_k. Since E_k is defined from V' and G, we only need to compute the set V', which is precisely the set of vertices of G whose degree is not equal to 2. This can be done in polynomial time. □

Since G' is an interval graph, we have $\mathrm{box}(G') \leq 1$ and an interval representation of G' can be constructed in linear time [5]. Let T be the set of vertices of G, which are incident to at least one edge in E_k. In other words, $T = V' \bigcup_{v \in V'} N_G(v)$. Since maximum degree of G is at most k (as explained at the beginning of this proof), we get $|T| \leq |V'| + k \cdot |V'| \leq (4k-2) + k \cdot (4k-2) = O(k^2)$. From the proof of Theorem 4, we can notice that the proof goes through with this definition of T and the complexity of the algorithm depends only on $|T|$ and not on the number of edges being modified in G.

For clarity, we just repeat some important points of the algorithm of Theorem 4 here, with modifications occurring mainly in the running time analysis. Let $H_1(V, E_1)$ be the graph obtained from G' by making T a clique, without altering other adjacencies in G'. From the box representation of G' of dimension one, in $O(n)$ time we can construct (by Lemma 3) a box representation $\mathcal{B}_1$ of H_1 with dimension 2.

Let $H_2(V, E_2)$ be the graph obtained from G by making vertices in $V \setminus T$ adjacent to every other vertex in the graph and maintaining other adjacencies in G unaltered. As in the proof of Theorem 4, we have $H_2[T] = G[T]$. Hence, $\mathrm{box}(H_2[T]) = \mathrm{box}(G[T]) \leq \mathrm{treewidth}(G[T]) + 2 \leq \mathrm{treewidth}(G) + 2 \leq 2 \cdot \mathrm{ML}(G) + 2 \leq 2k + 2$ [12,19]. We know that $|T| = O(k^2)$ and therefore, using the construction in Proposition 1, an optimal box representation $\mathcal{B}_T$ of $H_2[T]$ can be computed in $2^{O(k^3 \log k)}$ time and from $\mathcal{B}_T$, an optimal box representation of H_2 of dimension at most $\mathrm{box}(G)$ is computed in polynomial time.

Union of box representations of H_1 and H_2 gives a $2 + \mathrm{box}(G)$ dimensional box representation for G, obtained in $2^{O(k^3 \log k)} n^{O(1)}$ time. □

Remark 5. If $\mathrm{ML}(G) \leq k$, we get an FPT algorithm that achieves an additive 2 approximation for $\mathrm{box}(G)$ which runs in time $2^{O(k^3 \log k)} n^{O(1)}$, the running time and approximation ratio being the same as in Adiga et al. [4]. Bruhn et al. [7] gave an additive 1 approximation algorithm for the same problem. However, the running time of their algorithm is very high, compared to that of the algorithm presented here.

7. An FPT approximation scheme for computing cubicity with parameter MVC

Fellows et al. [18, Corollary 5] proved an existential result that for every fixed pair of integers k and b, there is an $f(k)\cdot n$ time algorithm which determines whether a given graph G on n vertices and $\mathrm{MVC}(G) \leq k$ has cubicity at most b. In the theorem below, we derive a FPT approximation algorithm, for computing the cubicity of graphs, using their vertex cover number as the parameter. Our algorithm is constructive.

Theorem 5. *Let G be a graph on n vertices. A cube representation of G which is of dimension at most $2\,\mathrm{cub}(G)$ can be computed in time $2^{O(2^k k^2)} n^{O(1)}$, where $k = \mathrm{MVC}(G)$. By allowing a larger running time of $2^{O(g(k,\epsilon))} n^{O(1)}$, we can achieve a $(1+\epsilon)$ approximation factor, for any $\epsilon > 0$, where $g(k,\epsilon) = \frac{1}{\epsilon} k^3 2^{\frac{4k}{\epsilon}}$.*

Proof. Let $G(V, E)$ be a graph on n vertices. Without loss of generality, we can assume that G is connected. We can compute a minimum vertex cover of G in time $2^{O(k)} n^{O(1)}$ [27]. Let $S \subseteq V$ be a vertex cover of G of cardinality k. We define two supergraphs of G, namely $H_1(V, E_1)$ and $H_2(V, E_2)$ such that $E = E_1 \cap E_2$ with $\mathrm{cub}(H_1) \leq \mathrm{cub}(G)$ and $\mathrm{cub}(H_2) \leq \mathrm{cub}(G)$.

Let $S \subseteq V$ be a vertex cover of G of cardinality k. First we define an equivalence relation on the vertices of the independent set $V \setminus S$ such that vertices u and v are in the same equivalence class if and only if $N_G(u) = N_G(v)$. Let $A_1, A_2, \ldots, A_t$ be the equivalence classes. We define H_1 to be the graph obtained from G by making each A_i into a clique and maintaining other adjacencies as it is in G. Formally, $E_1 = E \cup \{(u, v) \mid u \neq v$ and u, v belong to the same A_i, for some $1 \leq i \leq t\}$.

For each A_i, let us consider the mapping $n_{A_i} : A_i \mapsto \{1, 2, \ldots, |A_i|\}$, where $n_{A_i}(v)$ is the unique number representing $v \in A_i$. (Note that if $u \in A_i$ and $v \in A_j$, where $i \neq j$, then, $n_{A_i}(u)$ and $n_{A_j}(v)$ could potentially be the same.) Let $s = \max_{1\leq i\leq t} |A_i|$. We define one more partitioning of the independent set $V \setminus S$ into equivalence classes $B_1, B_2, \ldots, B_s$ such that for $1 \leq i \leq s$, $B_i = \{v \mid n_{A_j}(v) = i$, for some $1 \leq j \leq t\}$. We define H_2 to be the graph obtained from G by making each B_i into a clique, and making each vertex in S adjacent to every other vertex in V. Formally, $E_2 = \{(u, v) \mid u \neq v$ and $u \in S, v \in V\} \cup \{(u, v) \mid u \neq v$ and u, v belong to the same B_i, for some $1 \leq i \leq s\}$.

If u, v are two adjacent vertices of a graph G such that $N_G(u) \cup \{u\} = N_G(v) \cup \{v\}$, we call them as twin vertices. G' is called a reduced graph of G if G' is obtained from G by repeatedly contracting the edges among pairs of twin vertices.

Claim 3. *If G' is a reduced graph of G, then, $\mathrm{cub}(G) = \mathrm{cub}(G')$ and from an optimal cube representation $\mathcal{C}'$ of G', in polynomial time, we can obtain an optimal cube representation $\mathcal{C}$ of G.*

Proof. Let $\mathcal{C}' = \{I'_1, I'_2, \cdots, I'_p\}$ be an optimal cube representation of G'. For each $1 \leq i \leq p$, define the interval graph I_i as follows: If $u \in V(G')$, then the interval corresponding to u in I_i is same as it is in I'_i. If $u \in V(G) \setminus V(G')$, then $\exists v \in V(G')$ such that u, v are twins in G. In this case, define the interval corresponding to u in I_i is same as the interval of its twin v in I'_i. It can be verified that $\mathcal{C} = \{I_1, I_2, \cdots, I_p\}$ is a valid cube representation of G. Thus, $\mathrm{cub}(G) \leq p$. Since G' is an induced subgraph of G, we also have $\mathrm{cub}(G) \geq \mathrm{cub}(G') = p$. □

Observe that in graph H_1, vertices in each A_i, $1 \leq i \leq t$ are twins of each other. We can construct a reduced graph H'_1 of H_1 by contracting all vertices in A_i to a single vertex, for each $1 \leq i \leq t$. Now, H'_1 has only $t + |S|$ vertices, which is at most $2^k - 1 + k$. It is known that $\mathrm{cub}(H'_1) \leq \mathrm{MVC}(H'_1) + \lceil \log(|V(H'_1)| - \mathrm{MVC}(H'_1)) \rceil - 1$ [11]. Since $\mathrm{MVC}(H'_1) = k$, we get $\mathrm{cub}(H'_1) \leq 2k - 1$. Using the construction in Proposition 1, we can compute an optimal cube representation $\mathcal{C}'_1$ of H'_1 in time $2^{O(2^k k^2)}$. By the claim above, from $\mathcal{C}'_1$ we can get an optimal cube representation $\mathcal{C}_1$ of H_1 in polynomial time, with $|\mathcal{C}_1| = \mathrm{cub}(H'_1)$. Observe that H'_1 is an induced subgraph of G, which implies $|\mathcal{C}_1| \leq \mathrm{cub}(G)$.

Similarly, observe that, in graph H_2, vertices in each B_i, $1 \leq i \leq s$ are twins of each other. We can construct a reduced graph H'_2 of H_2 by contracting all vertices in B_i to a single vertex, for each $1 \leq i \leq s$ and contracting S to a single vertex. Now, H'_2 is a graph on $s + 1$ vertices. We can also observe that H'_2 is in fact a star graph with s leaves. In polynomial time, we can construct an optimal cube representation $\mathcal{C}'_2$ of H'_2 which is of dimension $\lceil \log s \rceil$ [32]. As earlier, from $\mathcal{C}'_2$ we can get an optimal cube representation $\mathcal{C}_2$ of H_2 in polynomial time, with $|\mathcal{C}_2| = \mathrm{cub}(H'_2) = \lceil \log s \rceil$. Observe that H'_2 is an induced subgraph of G, which implies $|\mathcal{C}_2| \leq \mathrm{cub}(G)$.

It can be easily verified that $E = E_1 \cap E_2$ and hence $\mathcal{C}_1 \cup \mathcal{C}_2$ is a valid cube representation of G of dimension $|\mathcal{C}_1| + |\mathcal{C}_2| \leq 2\,\mathrm{cub}(G)$, constructible in $2^{O(2^k k^2)} n^{O(1)}$ time.

We can also achieve a $(1+\epsilon)$ approximation factor, for any $\epsilon > 0$ by allowing a larger running time as explained below. Define $f(k_\epsilon) = k\left(1 + 2^{(2k-1)/\epsilon}\right)$, where $k = \mathrm{MVC}(G)$. If $|V(G)| = n \leq f(k_\epsilon)$, then, by Proposition 1, we can get an optimal cube representation of G in time $2^{O(f^{2k_\epsilon} \log f(k_\epsilon))}$. Otherwise, we have $(2k-1)/\lceil \log \lceil (n-k)/k \rceil \rceil \leq \epsilon$. In this case, we use the construction described above, to get a cube representation of G of dimension $|\mathcal{C}_1| + |\mathcal{C}_2|$. We prove that in this case, $|\mathcal{C}_1| + |\mathcal{C}_2| \leq \mathrm{cub}(G)(1+\epsilon)$.

It is known that $\mathrm{cub}(G) \geq \lceil \log \psi(G) \rceil$, where $\psi(G)$ is the number of leaf nodes in the largest induced star in G [3]. By the pigeon hole principle, $\max_{v\in S} |N_G(v) \cap (V \setminus S)| \geq \lceil (n-k)/k \rceil$. Therefore, $\mathrm{cub}(G) \geq \lceil \log \psi(G) \rceil \geq \lceil \log \lceil (n-k)/k \rceil \rceil$. Recall that $|\mathcal{C}_1| \leq 2k-1$. Therefore, $|\mathcal{C}_1| + |\mathcal{C}_2| \leq 2k - 1 + \mathrm{cub}(G) \leq \mathrm{cub}(G)\left(\frac{2k-1}{\mathrm{cub}(G)} + 1\right) \leq \mathrm{cub}(G)\left(\frac{2k-1}{\lceil \log \lceil (n-k)/k \rceil \rceil} + 1\right) \leq \mathrm{cub}(G)(1+\epsilon)$.

The total running time of this algorithm is $2^{O\left(\frac{1}{\epsilon} k^3 2^{\frac{4k}{\epsilon}}\right)} n^{O(1)}$. □

8. Conclusion

We have presented $o(n)$ factor approximation algorithms for computing the boxicity and cubicity of graphs. Using these algorithms, we also derived $o(n)$ factor approximation algorithms for some related well-known problems, including poset dimension and Ferrers dimension. To the best of our knowledge, for none of these problems polynomial time sublinear factor approximation algorithms were known previously. Since polynomial time approximations within an $O(n^{1-\epsilon})$ factor for any $\epsilon > 0$ are considered unlikely for any of these problems, no significant improvement in the approximation factor can be expected. We have also presented a general method of obtaining parameterized approximation algorithms for boxicity using vertex and edge edit distance parameters.

References

[1] A. Adiga, J. Babu, L.S. Chandran, Polynomial time and parameterized approximation algorithms for boxicity, in: IPEC, 2012, pp. 135–146.
[2] A. Adiga, D. Bhowmick, L.S. Chandran, Boxicity and poset dimension, SIAM J. Discrete Math. 25 (4) (2011) 1687–1698.
[3] A. Adiga, L.S. Chandran, Cubicity of interval graphs and the claw number, J. Graph Theory 65 (2010) 323–333.
[4] A. Adiga, R. Chitnis, S. Saurabh, Parameterized algorithms for boxicity, in: ISAAC, 2010, pp. 366–377.
[5] K.S. Booth, G.S. Lueker, Testing for the consecutive ones property, interval graphs, and graph planarity using pq-tree algorithms, J. Comput. System Sci. 13 (3) (1976) 335–379.
[6] H. Breu, D.G. Kirkpatrick, Unit disk graph recognition is NP-hard, Comput. Geom. 9 (1998) 2–34.
[7] H. Bruhn, M. Chopin, F. Joos, O. Schaudt, Structural parameterizations for boxicity, Algorithmica 74 (4) (2016) 1453–1472.
[8] L. Cai, Parameterized complexity of vertex colouring, Discrete Appl. Math. 127 (3) (2003) 415–429.
[9] Y. Cao, J. Chen, Y. Liu, On feedback vertex set: New measure and new structures, Algorithmica 73 (1) (2015) 63–86.
[10] P. Chalermsook, B. Laekhanukit, D. Nanongkai, Graph products revisited: Tight approximation hardness of induced matching, poset dimension and more, SODA 2013, 2013, pp. 1557–1576.
[11] L.S. Chandran, A. Das, C.D. Shah, Cubicity, boxicity, and vertex cover, Discrete Math. 309 (8) (2009) 2488–2496.
[12] L.S. Chandran, N. Sivadasan, Boxicity and treewidth, J. Combin. Theory Ser. B 97 (2007) 733–744.
[13] S. Chaplick, P. Hell, Y. Otachi, T. Saitoh, R. Uehara, Ferrers dimension; boxicity; segment-ray graph; unit grid intersection graph, Discrete Appl. Math. 216 (Part 1) (2017) 130–135.
[14] V. Chvátal, P.L. Hammer, Aggregation of inequalities in integer programming, Ann. Discrete Math.: Stud. Integer Program. 1 (1977) 145–162.
[15] O. Cogis, On the Ferrers dimension of a digraph, Discrete Math. 38 (1) (1982) 47–52.
[16] M.B. Cozzens, M.D. Halsey, The relationship between the threshold dimension of split graphs and various dimensional parameters, Discrete Appl. Math. 30 (2–3) (1991) 125–135.
[17] B. Dushnik, E.W. Miller, Partially ordered sets, Amer. J. Math. 63 (3) (1941) 600–610.
[18] M.R. Fellows, D. Hermelin, F.A. Rosamond, Well-quasi-orders in subclasses of bounded treewidth graphs, in: IWPEC, 2009, pp. 149–160.
[19] M. Fellows, D. Lokshtanov, N. Misra, M. Mnich, F. Rosamond, S. Saurabh, The complexity ecology of parameters: An illustration using bounded max leaf number, Theoret. Comput. Sci. 45 (4) (2009) 822–848.
[20] S. Felsner, M. Habib, R.H. Möhring, On the interplay between interval dimension and dimension, SIAM J. Discrete Math. 7 (1) (1994) 32–40.
[21] S. Felsner, I. Mustata, M. Pergel, The complexity of the partial order dimension problem - closing the gap, SIAM J. Discrete Math. 31 (2017) 172–189.
[22] M. Grohe, Computing crossing numbers in quadratic time, J. Comput. System Sci. 68 (2004) 285–302.
[23] P. Hof, Y. Villanger, Proper interval vertex deletion, Algorithmica 65 (4) (2013) 845–867.
[24] J. Kratochvíl, A special planar satisfiability problem and a consequence of its NP-completeness, Discrete Appl. Math. 52 (3) (1994) 233–252.
[25] N.V.R. Mahadev, U.N. Peled, Threshold Graphs and Related Topics, in: Annals of Discrete Mathematics, vol. 56, Elsevier, 1995.
[26] D. Marx, I. Schlotter, Obtaining a planar graph by vertex deletion, Algorithmica 62 (3–4) (2012) 807–822.
[27] R. Niedermeier, Invitation to Fixed-Parameter Algorithms, in: Oxford Lecture Series in Mathematics and Its Applications, vol. 31, Oxford University Press, 2006.
[28] J. Riguet, Les relations de Ferrers, C. R. Séances Hebd. Acad. Sci. (Paris) 232 (1951) 1729–1730.
[29] F.S. Roberts, On the boxicity and cubicity of a graph, in: Recent Progresses in Combinatorics, Academic Press, New York, 1969, pp. 301–310.
[30] B. Rosgen, L. Stewart, Complexity results on graphs with few cliques, Discrete Math. Theoret. Comput. Sci. 9 (2007) 127–136.
[31] E.R. Scheinerman, Intersection Classes and Multiple Intersection Parameters of Graphs, Princeton University, 1984 (Ph.D. Thesis).
[32] C.D. Shah, Boxicity, Cubicity, and Vertex Cover, (M. Sc Thesis), IISc, Bangalore, 2008 http://clweb.csa.iisc.ernet.in/chintan/chintan_thesis.pdf.
[33] J.P. Spinrad, Efficient Graph Representations, in: Fields Institute Monographs, vol. 19, American Mathematical Society, 2003.
[34] C. Thomassen, Interval representations of planar graphs, J. Combin. Theory Ser. B 40 (1986) 9–20.
[35] W.T. Trotter Jr., Combinatorial problems in dimension theory for partially ordered sets, in: Problemes Combinatoires et Theorie des Graphes, Colloque Internationeaux C.N.R.S. (260), 1978, pp. 403–406.
[36] Y. Villanger, P. Heggernes, C. Paul, J.A. Telle, Interval completion is fixed parameter tractable, SIAM J. Comput. 38 (5) (2008) 2007–2020.
[37] M. Yannakakis, The complexity of the partial order dimension problem, SIAM J. Algebr. Discrete Methods 3 (3) (1982) 351–358.