Cubicity, Degeneracy, and Crossing Number

Abhijin Adiga, L. Sunil Chandran, and Rogers Mathew

Department of Computer Science and Automation,
Indian Institute of Science,
Bangalore - 560012, India
`{abhijin,sunil,rogers}@csa.iisc.ernet.in`

Abstract

A k-box $B = (R_1, R_2, \ldots, R_k)$, where each R_i is a closed interval on the real line, is defined to be the Cartesian product $R_1 \times R_2 \times \cdots \times R_k$. If each R_i is a unit length interval, we call B a k-cube. *Boxicity* of a graph G, denoted as $box(G)$, is the minimum integer k such that G is an intersection graph of k-boxes. Similarly, the *cubicity* of G, denoted as $cub(G)$, is the minimum integer k such that G is an intersection graph of k-cubes.

It was shown in [L. Sunil Chandran, Mathew C. Francis, and Naveen Sivadasan. Representing graphs as the intersection of axis-parallel cubes. *MCDES-2008, IISc Centenary Conference*, available at *CoRR*, abs/cs/0607092, 2006.] that, for a graph G with maximum degree Δ, $cub(G) \leq \lceil 4(\Delta + 1) \ln n \rceil$. In this paper we show that, for a k-degenerate graph G, $cub(G) \leq (k+2)\lceil 2e \log n \rceil$. Since k is at most Δ and can be much lower, this clearly is a stronger result. We also give an efficient deterministic algorithm that runs in $O(n^2 k)$ time to output a $8k(\lceil 2.42 \log n \rceil + 1)$ dimensional cube representation for G.

The *crossing number* of a graph G, denoted as $CR(G)$, is the minimum number of crossing pairs of edges, over all drawings of G in the plane. An important consequence of the above result is that if the crossing number of a graph G is t, then $box(G)$ is $O(t^{1/4} \lceil \log t \rceil^{3/4})$. This bound is tight upto a factor of $O((\log t)^{3/4})$.

Let $(\mathcal{P}, \leq)$ be a partially ordered set and let $G_{\mathcal{P}}$ denote its underlying comparability graph. Let $dim(\mathcal{P})$ denote the *poset dimension* of $\mathcal{P}$. Another interesting consequence of our result is to show that $dim(\mathcal{P}) \leq 2(k+2)\lceil 2e \log n \rceil$, where k denotes the degeneracy of $G_{\mathcal{P}}$. Also, we get a deterministic algorithm that runs in $O(n^2 k)$ time to construct a $16k(\lceil 2.42 \log n \rceil + 1)$ sized realizer for $\mathcal{P}$. As far as we know, though very good upper bounds exist for poset dimension in terms of maximum degree of its underlying comparability graph, no upper bounds in terms of the degeneracy of the underlying comparability graph is seen in the literature.

1998 ACM Subject Classification G.2.2 Graph Theory

Keywords and phrases Degeneracy, Cubicity, Boxicity, Crossing Number, Interval Graph, Intersection Graph, Poset Dimension, Comparability Graph

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2011.176

1 Introduction

A graph G is an *intersection graph* of sets from a family of sets $\mathcal{F}$, if there exists $f : V(G) \to \mathcal{F}$ such that $(u, v) \in E(G) \Leftrightarrow f(u) \cap f(v) \neq \emptyset$. Representations of graphs as the intersection graphs of various geometrical objects is a well studied topic in graph theory. Probably the most well studied class of intersection graphs are the *interval graphs*. Interval graphs are the intersection graphs of closed intervals on the real line. A restricted form of interval graphs, that allow only intervals of unit length, are *indifference graphs* or *unit interval graphs*.

An interval on the real line can be generalized to a "k-box" in $\mathbb{R}^k$. A k-box $B = (R_1, R_2, \ldots, R_k)$, where each R_i is a closed interval on the real line, is defined to be the

31st Int'l Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2011).
Editors: Supratik Chakraborty, Amit Kumar; pp. 176–190
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Cartesian product $R_1 \times R_2 \times \cdots \times R_k$. If each R_i is a unit length interval, we call B a k-cube. Thus, 1-boxes are just closed intervals on the real line whereas 2-boxes are axis-parallel rectangles in the plane. The parameter boxicity of a graph G, denoted as $box(G)$, is the minimum integer k such that G is an intersection graph of k-boxes. Similarly, the cubicity of G, denoted as $cub(G)$, is the minimum integer k such that G is an intersection graph of k-cubes. Thus, interval graphs are the graphs with boxicity equal to 1 and unit interval graphs are the graphs with cubicity equal to 1. A *k-box representation* or a *k dimensional box representation* of a graph G is a mapping of the vertices of G to k-boxes such that two vertices in G are adjacent if and only if their corresponding k-boxes have a non-empty intersection. In a similar way, we define *k-cube representation* (or *k dimensional cube representation*) of a graph G. Since k-cubes by definition are also k-boxes, boxicity of a graph is at most its cubicity.

The concepts of boxicity and cubicity were introduced by F.S. Roberts in 1969 [15]. Roberts showed that for any graph G on n vertices $box(G) \leq \lfloor \frac{n}{2} \rfloor$ and $cub(G) \leq \lfloor \frac{2n}{3} \rfloor$. Both these bounds are tight since $box(K_{2,2,\ldots,2}) = \lfloor \frac{n}{2} \rfloor$ and $cub(K_{3,3,\ldots,3}) = \lfloor \frac{2n}{3} \rfloor$ where $K_{2,2,\ldots,2}$ denotes the complete $n/2$-partite graph with 2 vertices in each part and $K_{3,3,\ldots,3}$ denotes the complete $n/3$-partite graph with 3 vertices in each part. It is easy to see that the boxicity of any graph is at least the boxicity of any induced subgraph of it.

Box representation of graphs finds application in niche overlap (competition) in ecology and to problems of fleet maintenance in operations research (see [9]). Given a low dimensional box representation, some well known NP-hard problems become polynomial time solvable. For instance, the max-clique problem is polynomial time solvable for graphs with boxicity k because the number of maximal cliques in such graphs is only $O((2n)^k)$.

1.1 Previous Results on Boxicity and Cubicity

It was shown by Cozzens [8] that computing the boxicity of a graph is **NP**-hard. Kratochvíl [11] showed that deciding whether the boxicity of a graph is at most 2 itself is **NP**-complete. It has been shown by Yannakakis [19] that deciding whether the cubicity of a given graph is at least 3 is **NP**-hard.

Researchers have tried to bound the boxicity and cubicity of graph classes with special structure. Scheinerman [16] showed that the boxicity of outerplanar graphs is at most 2. Thomassen [17] proved that the boxicity of planar graphs is bounded from above by 3. Upper bounds for the boxicity of many other graph classes such as chordal graphs, AT-free graphs, permutation graphs etc. were shown in [7] by relating the boxicity of a graph with its treewidth. The cube representation of special classes of graphs like hypercubes and complete multipartite graphs were investigated in [15, 12, 13].

Various other upper bounds on boxicity and cubicity in terms of graph parameters such as maximum degree, treewidth etc. can be seen in [4, 2, 3, 10, 7]. The ratio of cubicity to boxicity of any graph on n vertices was shown to be at most $\lceil \log_2 n \rceil$ in [5].

1.2 Equivalent Definitions for Boxicity and Cubicity

Let $G, G_1, G_2, \ldots, G_b$ be a collection of graphs with $V(G) = V(G_i)$, for every $i \leq b$. We say $G = \bigcap_{i=1}^{b} G_i$ when $E(G) = \bigcap_{i=1}^{b} E(G_i)$. Below, we state two very useful lemmas due to Roberts [15].

▶ **Lemma 1.** *For any graph G, $box(G) \leq k$ if and only if there exist k interval graphs $I_1, \ldots, I_k$ such that $G = I_1 \cap \cdots \cap I_k$.*

▶ **Lemma 2.** *For any graph G, $cub(G) \leq k$ if and only if there exist k indifference graphs (unit interval graphs) $I_1, \ldots, I_k$ such that $G = I_1 \cap \cdots \cap I_k$.*

1.3 Our Results

A graph G is *k-degenerate* if the vertices of G can be enumerated in such a way that every vertex is succeeded by at most k of its neighbors. The least number k such that G is k-degenerate is called the degeneracy of G and any such enumeration is referred to as a *degeneracy order* of $V(G)$. For example, trees and forests are 1-degenerate and planar graphs are 5-degenerate. Series-parallel graphs, outerplanar graphs, non-regular cubic graphs, circle graphs of girth at least 5 etc. are subclasses of 2-degenerate graphs.

Main Result: It was shown in [2] that, for a graph G with maximum degree Δ, $cub(G) \leq \lceil 4(\Delta+1)\ln n \rceil$. In this paper, we show that, for a k-degenerate graph G, $cub(G) \leq (k+2)\lceil 2e \log n \rceil$. Since k is at most Δ and can be much lower, this clearly is a stronger result. Moreover, we give an *efficient deterministic algorithm* that outputs a $8k(\lceil 2.42 \log n \rceil + 1)$ dimensional cube representation for G in $O(n^2 k)$ time.

Consequence 1: The *crossing number* of a graph G, denoted as $CR(G)$, is the minimum number of crossing pairs of edges, over all drawings of G in the plane. We prove that, if $CR(G) = t$, then $box(G) \leq 66 t^{\frac{1}{4}} \lceil \log 4t \rceil^{\frac{3}{4}} + 6$. This bound is tight upto a factor of $O((\log t)^{\frac{3}{4}})$. See Section 5 for details.

Consequence 2: Let $(\mathcal{P}, \leq)$ be a poset (partially ordered set) and let $G_{\mathcal{P}}$ be the underlying comparability graph of $\mathcal{P}$. A linear extension L of $\mathcal{P}$ is a total order which satisfies $(x \leq y \in \mathcal{P}) \implies (x \leq y \in L)$. A realizer of $\mathcal{P}$ is a set of linear extensions of $\mathcal{P}$, say $\mathcal{R}$, which satisfy the following condition: for any two distinct elements x and y, $x \leq y$ in $\mathcal{P}$ if and only if $x \leq y$ in L, $\forall L \in \mathcal{R}$. The *poset dimension* of $\mathcal{P}$, denoted by $dim(\mathcal{P})$, is the minimum integer k such that there exists a realizer of $\mathcal{P}$ of cardinality k. Yannakakis [19] showed that it is NP-complete to decide whether the dimension of a poset is at most 3. The poset dimension is an extensively studied parameter in the theory of partial order (See [18] for a comprehensive treatment).

There are several research papers in the partial order literature which study the dimension of posets whose underlying comparability graph has some special structure – interval order, semi order and crown posets are some examples. While very good upper bounds (for example $c\Delta(\log \Delta)^2$ in [20], where c is a constant) are known for poset dimension in terms of maximum degree Δ of its underlying comparability graph, as far as we know there are no upper bounds in terms of the degeneracy of the underlying comparability graph. Connecting our main result with a result in [1], we can get an upper bound for poset dimension in terms of the degeneracy of the underlying comparability graph as follows. It was shown in [1] that $dim(\mathcal{P}) < 2box(G_{\mathcal{P}})$. Therefore, if the degeneracy of the underlying comparability graph $G_{\mathcal{P}}$ is k, then our result says that $dim(\mathcal{P}) \leq 2(k+2)\lceil 2e \log n \rceil$. Also, we get a deterministic algorithm that runs in $O(n^2 k)$ time to construct a $16k(\lceil 2.42 \log n \rceil + 1)$ sized realizer for $\mathcal{P}$.

2 Preliminaries

For any finite positive integer n, let $[n]$ denote the set $\{1, 2, \ldots n\}$. Unless mentioned explicitly, all logarithms are to the base e in this paper. All the graphs that we consider are simple, finite and undirected. For a graph G, we denote the vertex set of G by $V(G)$ and the edge set of G by $E(G)$. For any vertex $u \in V(G)$, $N_G(u) = \{v \in V(G) \mid (u, v) \in E(G)\}$. We define $deg_G(u) := |N_G(u)|$. The average degree of G is denoted by $d_{av}(G)$.

Since an interval graph is the intersection graph of closed intervals on the real line, for every interval graph I_a, there exists a function $f_a : V(I_a) \rightarrow \{X \subseteq \mathbb{R} \mid X \text{ is a closed interval}\}$, such that for $u, v \in V(I_a)$, $(u, v) \in E(I_a) \Leftrightarrow f_a(u) \cap f_a(v) \neq \emptyset$. The function f_a is called an *interval representation* of the interval graph I_a. Note that the interval representation of an interval graph need not be unique. Given a closed interval $X = [y, z]$, we define $L(X) := y$ and $R(X) := z$. In a similar way, we call a function f_b a *unit interval representation* of unit interval graph I_b if $f_b : V(I_b) \rightarrow \{X' \subseteq \mathbb{R} \mid X' \text{ is a unit length closed interval}\}$, such that $\forall u, v \in V(I_b)$, $(u, v) \in E(I_b) \Leftrightarrow f_b(u) \cap f_b(v) \neq \emptyset$.

Given a graph G, let $\mathcal{C}$ be a coloring of $V(G)$ using colors $\chi_1, \chi_2, \ldots, \chi_a$. Then, for each $u \in V(G)$, $\mathcal{C}(u)$ denotes the color of u in $\mathcal{C}$.

2.1 Definitions, Notations and Assumptions used in Sections 3 and 4:

Recall that the degeneracy of a graph is the least number k such that it has a vertex enumeration in which each vertex is succeeded by at most k of its neighbors. Such an enumeration is called the degeneracy order. The graph G that we consider in these sections is a k-degenerate graph having $V(G) = \{v_1, v_2, \ldots, v_n\}$, $|E(G)| = m$ and $\overline{m}$ $(= \binom{n}{2} - m)$ denotes the number of non-edges in G. The enumeration $v_1, v_2, \ldots, v_n$ is a degeneracy order of $V(G)$ and is denoted by $\mathcal{D}$. For every $v_i, v_j \in V(G)$, we say $v_i <_{\mathcal{D}} v_j$ if v_i comes before v_j in $\mathcal{D}$ i.e., $v_i <_{\mathcal{D}} v_j$ if and only if $i < j$. Suppose $v_i <_{\mathcal{D}} v_j$. If $(v_i, v_j) \in E(G)$, then we call v_j a *forward neighbor* of v_i and v_i is referred to as a *backward neighbor* of v_j. Observe that since G is k-degenerate, a vertex can have at most k forward neighbors. If $(v_i, v_j) \notin E(G)$, then v_j a *forward non-neighbor* of v_i and v_i is a *backward non-neighbor* of v_j. For any $u \in V(G)$, $N_G^f(u) = \{w \in V(G) \mid w \text{ is a forward neighbor of } u\}$ and $N_G^b(u) = \{w \in V(G) \mid w \text{ is a backward neighbor of } u\}$.
Support sets of a non-edge: For each $(v_x, v_y) \notin E(G)$, where $v_x <_{\mathcal{D}} v_y$, let $S_{xy} = \{v_z \in N_G^f(v_x) \mid v_y <_{\mathcal{D}} v_z\} \cup \{v_y\}$. We call S_{xy} the *weak support set* of the non-edge (v_x, v_y). Define $T_{xy} = S_{xy} \cup \{v_x\}$. We call T_{xy} the *strong support set* of the non-edge (v_x, v_y). Let $\mathcal{C}$ be a coloring (need not be proper) of $V(G)$. We say S_{xy} is *favorably colored* in $\mathcal{C}$, if $\mathcal{C}(v_y) \neq \mathcal{C}(v_w)$, $\forall v_w \in S_{xy} \setminus \{v_y\}$. We say T_{xy} is *favorably colored* in $\mathcal{C}$, if $\mathcal{C}(v_y) \neq \mathcal{C}(v_w)$, $\forall v_w \in T_{xy} \setminus \{v_y\}$

3 Cube Representation and Coloring

▶ **Lemma 3.** *Let G be a k-degenerate graph. Let $\chi = \{\chi_1, \chi_2, \ldots \chi_a\}$ be a set of colors and let $\mathbb{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots \mathcal{C}_b\}$ be a family of colorings (need not be proper) of $V(G)$, where each $\mathcal{C}_i$ uses colors from the set χ. If the strong support set T_{xy} of every non-edge $(v_x, v_y) \notin E(G)$, $v_x <_{\mathcal{D}} v_y$, is favorably colored in some $\mathcal{C}_i$, where $i \in [b]$, then $cub(G) \leq ab$.*

Proof. We prove this by constructing ab unit interval graphs $I_{i,j}$ on the vertex set $V(G)$, where $i \in [a]$ and $j \in [b]$, such that $G = \bigcap_{i=1}^{a} \bigcap_{j=1}^{b} I_{i,j}$. Then the statement will follow from Lemma 2. Let $f_{i,j}$ denote an interval representation of $I_{i,j}$. Let us partition the vertices of $I_{i,j}$ into two parts, namely A^{ij} and B^{ij}, where $A^{ij} = \{v \in V(G) \mid \mathcal{C}_i(v) = \chi_j\}$ and $B^{ij} = V(G) \setminus A^{ij}$. For every $i \in [a]$ and $j \in [b]$, an interval representation $f_{i,j}$ of $I_{i,j}$ is constructed from the coloring $\mathcal{C}_i$ in the following way. For every $v_y \in V(G)$, If $v_y \in A^{ij}$, then

$$f_{i,j}(v_y) = [y+n, y+2n]$$

else

$$f_{i,j}(v_y) = [g^{ij}_{max}(v_y), g^{ij}_{max}(v_y)+n], \text{ where}$$
$$g^{ij}_{max}(v_y) = \max(\{g \mid (v_y, v_g) \in E(G),$$
$$v_g \in A^{ij}\} \cup \{0\}).$$

Since the length of $f_{i,j}(v_y)$ is n, for every $v_y \in V(G)$, $I_{i,j}$ is a unit interval graph. It is easy to see that, $\forall v_x, v_y \in A^{ij}$, $2n \in f_{i,j}(v_x) \cap f_{i,j}(v_y)$ and therefore A^{ij} forms a clique in $I_{i,j}$. Since $n \in f_{i,j}(v_x) \cap f_{i,j}(v_y)$, $\forall v_x, v_y \in B^{i,j}$, $B^{i,j}$ too forms a clique in $I_{i,j}$. For every $(v_x, v_y) \in E(G)$, with $v_x \in A^{ij}$ and $v_y \in B^{ij}$, we have $L(f_{i,j}(v_y)) = g^{ij}_{max}(v_y) \leq n \leq L(f_{i,j}(v_x)) = n + x \leq n + g^{ij}_{max}(v_y)$, where the last inequality is inferred from the fact that $(v_x, v_y) \in E(G)$ and $v_x \in A^{ij}$. But $n + g^{ij}_{max}(v_y) = R(f_{i,j}(v_y))$. Therefore, we get $L(f_{i,j}(v_y)) \leq L(f_{i,j}(v_x)) \leq R(f_{i,j}(v_y))$ and hence $(v_x, v_y) \in E(I_{i,j})$. Hence $I_{i,j}$ is a supergraph of G.

Let $v_x <_{\mathcal{D}} v_y$ and $(v_x, v_y) \notin E(G)$. We now have to show that there exists some unit interval graph $I_{i,j}$ such that $(v_x, v_y) \notin E(I_{i,j})$. We know that, by assumption, there exists a coloring, say $\mathcal{C}_i$ (where $i \in [a]$), such that the strong support set T_{xy} is favorably colored in $\mathcal{C}_i$. Let $\chi_j = \mathcal{C}_i(v_y)$. Let $g = g^{ij}_{max}(v_x)$. We claim that $g < y$. Assume, for contradiction, that $g > y$. Then $g \neq 0$ and $v_g \in A^{ij}$. Since $y > x$, we get $g > x$. Therefore, $v_g \in N^f_G(v_x)$ and $g > y$. This implies that $v_g \in T_{xy}$. Since T_{xy} is favorably colored in $\mathcal{C}_i$, $\mathcal{C}_i(v_g) \neq \chi_j$. This contradicts the fact that $v_g \in A^{ij}$. Thus we prove the claim. Therefore, $R(f_{i,j}(v_x)) = n + g < n + y = L(f_{i,j}(v_y))$ and hence $(v_x, v_y) \notin E(I_{i,j})$. We infer that $G = \bigcap_{i=1}^{a} \bigcap_{j=1}^{b} I_{i,j}$. ◀

4 Cubicity and Degeneracy

4.1 An Upper Bound – Probabilistic Approach

▶ **Theorem 4.** *For every k-degenerate graph G, $cub(G) \leq (k+2) \cdot \lceil 2e \log n \rceil$*

Proof. Let $\chi = \{\chi_1, \chi_2, \dots \chi_{k+2}\}$ be a set of $k+2$ colors. Generate a random coloring $\mathcal{C}_1$ (need not be a proper coloring) of vertices of G in the following way: For each vertex $v_x \in V(G)$, pick a color χ_j, where $j \in [k+2]$, uniformly at random from χ and set $\mathcal{C}_1(v_x) = \chi_j$. In a similar way, independently generate random colorings $\mathcal{C}_2, \mathcal{C}_3, \dots \mathcal{C}_b$, where $b = \lceil 2e \log n \rceil$.

For every $(v_x, v_y) \notin E(G)$ and $v_x <_{\mathcal{D}} v_y$, since G is k-degenerate we have $|T_{xy}| = t \leq k+2$. $Pr[T_{xy}$ is favorably colored in $\mathcal{C}_i] = \frac{(k+2)(k+1)^{t-1}}{(k+2)^{t-1}} = \left(\frac{k+1}{k+2}\right)^{t-1} \geq \left(\frac{k+1}{k+2}\right)^{k+1}$. Therefore, $Pr[T_{xy}$ is not favorably colored in $\mathcal{C}_i] \leq 1 - \left(\frac{k+1}{k+2}\right)^{k+1} \leq e^{-\left(\frac{k+1}{k+2}\right)^{k+1}}$. Now taking $b = \lceil 2e \log n \rceil$,

$$Pr[\bigcup_{x,y:(v_x <_{\mathcal{D}} v_y),((v_x,v_y)\notin E(G))} \bigcap_{i=1}^{b} (T_{xy} \text{ is not favorably colored in } \mathcal{C}_i)]$$
$$\leq n^2 e^{-b\left(\frac{k+1}{k+2}\right)^{k+1}} < 1.$$

Hence, $Pr[\mathcal{C}_1, \mathcal{C}_2, \dots \mathcal{C}_b$ satisfy the condition of Lemma 3$] > 0$. Therefore, there exists a coloring $\mathcal{C}_1, \dots \mathcal{C}_b$, with $b = \lceil 2e \log n \rceil$, of $V(G)$ using colors from the set $\{\chi_1, \chi_2, \dots \chi_{k+2}\}$ such that the condition of Lemma 3 is satisfied. Hence by Lemma 3, $cub(G) \leq (k+2) \cdot \lceil 2e \log n \rceil$. ◀

4.2 Deterministic Algorithm

DET_ALGO(G) is a deterministic algorithm which takes a simple, finite k-degenerate graph G as input and outputs a cube representation in $8k\alpha$ dimensional space i.e., $8k\alpha$ unit interval graphs $I_{1,1}, \dots, I_{1,8k}, \dots, I_{\alpha,1}, \dots, I_{\alpha,8k}$ such that $G = \bigcap_{i=1}^{\alpha} \bigcap_{j=1}^{8k} I_{i,j}$. In order to achieve this, DET_ALGO(G) invokes the procedure CONSTRUCT_COLORING (for a detailed version of this procedure , see Appendix A.5) α times and thereby generates α colorings $\mathcal{C}_1, \dots, \mathcal{C}_\alpha$, where each coloring uses colors from the set $\{\chi_1, \dots, \chi_{8k}\}$. Then from each coloring $\mathcal{C}_i$, it constructs $8k$ unit interval graphs $I_{i,1}, \dots, I_{i,8k}$ using the construction described in Lemma 3, which is implemented in procedure CONSTRUCT_UNIT_INTERVAL_GRAPHS (See Appendix A.1).

Note that in order for G to be equal to $\bigcap_{i=1}^{\alpha} \bigcap_{j=1}^{8k} I_{i,j}$, Lemma 3 requires that the colorings $\mathcal{C}_1, \dots, \mathcal{C}_\alpha$ satisfy the following property: for every $(v_x, v_y) \notin E(G)$, where $v_x <_{\mathcal{D}} v_y$, there exists an $i \in [\alpha]$ such that the strong support set T_{xy} of this non-edge is favorably colored in $\mathcal{C}_i$. The colorings $\mathcal{C}_1, \dots, \mathcal{C}_\alpha$ are generated one by one keeping this objective in mind. At the stage when we have just generated the $(i-1)$-th coloring $\mathcal{C}_{i-1}$, if a non-edge (v_x, v_y) is such that its strong support set T_{xy} is already favorably colored in some $\mathcal{C}_j$, where $j < i$, then we say that the non-edge (v_x, v_y) is already DONE. Naturally at each stage we have to keep track of the non-edges that are not yet DONE. In order to do this, we introduce two data structures BNN_i and FNN_i, for all $i \in [\alpha]$ [1]. For each $v_y \in V(G)$,

$$BNN_i[v_y] = \{v_x \in V(G) \mid v_x \text{ is a backward non-neighbor of } v_y, \text{ and } (v_x, v_y) \text{ is not yet DONE with respect to } \mathcal{C}_1, \dots, \mathcal{C}_{i-1}.\}$$
$$FNN_i[v_y] = \{v_z \in V(G) \mid v_z \text{ is a forward non-neighbor of } v_y, \text{ and } (v_y, v_z) \text{ is not yet DONE with respect to } \mathcal{C}_1, \dots, \mathcal{C}_{i-1}.\}$$

It is easy to see that, $\bigcup_{v_y \in V(G)} BNN_i[v_y] = \bigcup_{v_y \in V(G)} FNN_i[v_y]$ and therefore, $\left(\bigcup_{v_y \in V(G)} BNN_i[v_y] = \emptyset\right) \iff \left(\bigcup_{v_y \in V(G)} FNN_i[v_y] = \emptyset\right)$. In Theorem 7, we show that if we select α to be at least $(\lceil 2.42 \log n \rceil + 1)$, then $FNN_{\alpha+1}[v_y] = \emptyset$, $\forall v_y \in V(G)$. This clearly would mean that all non-edges are DONE with respect to $\mathcal{C}_1, \dots, \mathcal{C}_\alpha$. In other words, the condition of Lemma 3 will be satisfied for $\mathcal{C}_1, \dots, \mathcal{C}_\alpha$.

The only thing that remains to be discussed now is how our coloring strategy (i.e. the procedure CONSTRUCT_COLORING) achieves the above objective, namely $BNN_{\alpha+1}[v_y] = \emptyset$ and $FNN_{\alpha+1}[v_y] = \emptyset$, $\forall v_y \in V(G)$, if $\alpha \geq (\lceil 2.42 \log n \rceil + 1)$. To start with $BNN_1[v_y]$ (respectively $FNN_1[v_y]$) contains all the backward (respectively forward) non-neighbors of v_y. The procedure CONSTRUCT_COLORING(i) generates the i-th coloring $\mathcal{C}_i$ as follows. It colors vertices in the reverse degeneracy order starting from vertex v_n. The partial coloring at the stage when we have colored the vertices v_n to v_z is denoted by $\mathcal{C}_i^{v_z}$. Note that $\mathcal{C}_i^{v_1} = \mathcal{C}_i$. Consider the stage at which the algorithm has already colored the vertices from v_n upto v_{y+1} and is about to color v_y. That is, we have the partial coloring $\mathcal{C}_i^{v_{y+1}}$ and are

[1] BNN – Backward Non-Neighbor, FNN – Forward Non-Neighbor

about to extend it to the partial coloring $\mathcal{C}_i^{v_y}$ by assigning one of the $8k$ possible colors to vertex v_y. Let $\mathcal{C}_i^{v_y=\chi_c}$ denote the partial coloring that results if we extend $\mathcal{C}_i^{v_{y+1}}$ by assigning color χ_c to v_y. The coloring $\mathcal{C}_i$ and the partial colorings $\mathcal{C}_i^{v_z}$, $\forall v_z \in V(G)$ and $\mathcal{C}_i^{v_z=\chi_c}$, $\forall v_z \in V(G), \chi_c \in \{\chi_1, \ldots, \chi_{8k}\}$, will be generically called *the colorings associated with the i-th stage* (i.e. the i-th invocation of CONSTRUCT_COLORING).

With respect to colorings $\mathcal{C}_1, \ldots, \mathcal{C}_{i-1}$ and some coloring $\mathcal{C}_i'$ associated with the i-th stage, we define the following sets:

$$\begin{aligned} W(v_w, \mathcal{C}_i') &= \{v_x \in BNN_i[v_w] \mid \text{the strong support set } T_{xw} \text{ of non-edge} \\ &\quad (v_x, v_w) \text{ is favorably colored in } \mathcal{C}_i'\} && (1)\\ X(v_w, \mathcal{C}_i') &= \{v_x \in BNN_i[v_w] \mid \text{the weak support set } S_{xw} \text{ of non-edge} \\ &\quad (v_x, v_w) \text{ is favorably colored in } \mathcal{C}_i'\} && (2)\\ Y(v_w, \mathcal{C}_i') &= \{v_z \in FNN_i[v_w] \mid \text{the strong support set } T_{wz} \text{ of non-edge} \\ &\quad (v_w, v_z) \text{ is favorably colored in } \mathcal{C}_i'\} && (3)\\ Z(v_w, \mathcal{C}_i') &= \{v_z \in FNN_i[v_w] \mid \text{the weak support set } S_{wz} \text{ of non-edge} \\ &\quad (v_w, v_z) \text{ is favorably colored in } \mathcal{C}_i'\} && (4) \end{aligned}$$

Naturally, we want to give a color χ_c to v_y such that a large number of (not yet DONE) non-edges incident on v_y get DONE. With respect to the colorings $\mathcal{C}_1, \ldots, \mathcal{C}_{i-1}$ and the partial coloring $\mathcal{C}_i^{v_y=\chi_c}$, we define the status of a non-edge incident on v_y as follows: A non-edge $(v_y, v_z) \in FNN_i[v_y]$ is DONE[2] if T_{yz} is favorably colored in $\mathcal{C}_i^{v_y=\chi_c}$ and is NOT-DONE if T_{yz} is not favorably colored in $\mathcal{C}_i^{v_y=\chi_c}$. A non-edge $(v_x, v_y) \in BNN_i[v_y]$ is HOPELESS[3] if S_{xy} (which happens to be a proper subset of T_{xy}) is not favorably colored in $\mathcal{C}_i^{v_y=\chi_c}$ and is HOPEFUL if S_{xy} is favorably colored in $\mathcal{C}_i^{v_y=\chi_c}$. So when we decide a color for v_y, our intention is to make a large fraction of the HOPEFUL non-edges of $FNN_i[v_y]$ (i.e. the set $Z(v_y, \mathcal{C}_i^{v_y=\chi_c})$), DONE and to make a large fraction of $BNN_i[v_y]$, HOPEFUL. More formally, we want the algorithm to assign a color χ_c to v_y such that the following two conditions are satisfied.
(i) $|X(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{3}{4}|BNN_i[v_y]|$, and
(ii)$|Y(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{3}{4}|Z(v_y, \mathcal{C}_i^{v_y=\chi_c})|$.
The obvious question then is, whether such a color χ_c always exists, for each $v_y \in V(G)$. Lemma 5 answers this question in the affirmative. It follows that, the number of non-edges that are not yet DONE with respect to colorings $\mathcal{C}_1, \ldots \mathcal{C}_i$ is at most a constant fraction of the number of non-edges that were not DONE with respect to colorings $\mathcal{C}_1, \ldots \mathcal{C}_{i-1}$. This is formally proved in Lemma 6. That $BNN_{\alpha+1}[v_y] = \emptyset$ and $FNN_{\alpha+1}[v_y] = \emptyset$, $\forall v_y \in V(G)$, is a consequence of this and is formally proved in Theorem 7.

▶ **Lemma 5.** *For every $i \in [\alpha], v_y \in V(G)$, (i) $|X(v_y, \mathcal{C}_i)| \geq \frac{3}{4}|BNN_i[v_y]|$, and (ii)$|Y(v_y, \mathcal{C}_i)| \geq \frac{3}{4}|Z(v_y, \mathcal{C}_i)|$.*

Proof. See Appendix A.2. ◀

▶ **Lemma 6.** *Let $\overline{m}_i = \Sigma_{y \in [n]}|FNN_i[v_y]|$. Then $\overline{m}_{i+1} \leq \frac{7}{16}\overline{m}_i$.*

---

[2] Recall that we had defined earlier that a non-edge (v_x, v_y) is DONE with respect to a list of colorings $\mathcal{C}_1, \ldots, \mathcal{C}_{i-1}$ if T_{xy} was favorably colored in some C_j, where $j < i$. Here we extend this notion, by allowing the partial coloring $\mathcal{C}_i^{v_y=\chi_c}$ also in the list.

[3] A HOPELESS non-edge (v_x, v_y) will not be DONE with respect to $\mathcal{C}_1, \ldots, \mathcal{C}_i$ if we set $\mathcal{C}_i(v_y) = \chi_c$, irrespective of the color given to $v_{y-1}, \ldots, v_1$.

Algorithm 4.1 DET_ALGO(G)

for $y = n$ to 1 **do**
  1. Initialize $BNN_1[v_y] \leftarrow \{v_x \in V(G) \mid v_x <_{\mathcal{D}} v_y, (v_x, v_y) \notin E(G)\}$.
  2. Initialize $FNN_1[v_y] \leftarrow \{v_z \in V(G) \mid v_y <_{\mathcal{D}} v_z, (v_y, v_z) \notin E(G)\}$.

end for
3. SET FLAG ← TRUE.
4. SET i ← 0.

while FLAG = TRUE **do**
  5. i++.
  6. $\mathcal{C}_i$ = CONSTRUCT_COLORING(i).
  for $y = 1$ to n **do**
    7. SET $BNN_{i+1}[v_y] \leftarrow BNN_i[v_y] \setminus W(v_y, \mathcal{C}_i)$
    8. SET $FNN_{i+1}[v_y] \leftarrow FNN_i[v_y] \setminus Y(v_y, \mathcal{C}_i)$
  end for
  9. If $FNN_{i+1}[v_y] = \emptyset$, $\forall v_y \in V(G)$, then FLAG = FALSE.

end while
10. SET $\alpha \leftarrow i$
11. CONSTRUCT_UNIT_INTERVAL_GRAPHS()

Proof. See Appendix A.3. ◀

▶ **Theorem 7.** *Let G be a k-degenerate graph. Algorithm DET_ALGO(G) constructs a valid* $8k(\lceil 2.42 \log n \rceil + 1)$ *dimensional cube representation for G.*

Proof. The algorithm constructs α colorings $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_\alpha$ of $V(G)$, where each coloring uses colors from the set $\{\chi_1, \chi_2, \ldots \chi_{8k}\}$. From Lemma 6, we have $\overline{m}_{i+1} \leq \frac{7}{16}\overline{m}_i$. Also, $\overline{m}_1 = |\Sigma_{y \in [n]} FNN_1[v_y]| \leq n^2$. Putting $\alpha = (\lceil 2.42 \log n \rceil + 1)$, we get $\overline{m}_\alpha \leq 1$. That is, for every $y \in [n]$, $FNN_{\alpha+1}[v_y] = EMPTY$. This means that, for every $(v_x, v_y) \notin E(G)$, where $v_x <_{\mathcal{D}} v_y$, there exists an $i \in [\alpha]$ such that T_{xy} is favorably colored in $\mathcal{C}_i$. Then by Lemma 3 , $cub(G) \leq 8k(\lceil 2.42 \log n \rceil + 1)$. The procedure CONSTRUCT_UNIT_INTERVAL_GRAPHS constructs $8k(\lceil 2.42 \log n \rceil + 1)$ unit interval graphs whose intersection gives G, as described in Lemma 3. Thus we prove the theorem. ◀

4.2.1 Running Time Analysis

▶ **Lemma 8.** *The procedure CONSTRUCT_COLORING(i) can be implemented to run in* $O(k\overline{m}_i + kn)$ *time, where* $\overline{m}_i = \Sigma_{y \in [n]} |FNN_i[v_y]|$.

Proof. See Appendix A.4. ◀

▶ **Theorem 9.** *DET_ALGO(G) runs in* $O(n^2 k)$ *time.*

Proof. The algorithm invokes the function CONSTRUCT_COLORING(i) α times to construct colorings $\mathcal{C}_1, \mathcal{C}_2, \ldots \mathcal{C}_\alpha$ of $V(G)$. By Lemma 8, to construct these α colorings it requires $O(\Sigma_{i=1}^{\alpha}(\overline{m}_i k) + \alpha k n)$ time. From Lemma 6, we get that $\Sigma_{i=1}^{\alpha}(\overline{m}_i)$ is $O(\overline{m})$. Since $\alpha = (\lceil 2.42 \log n \rceil + 1)$, the running time of the while loop in DET_ALGO(G) is $O(\overline{m}k + nk \log n)$. It is easy to see that the procedure CONSTRUCT_UNIT_INTERVAL_GRAPHS() runs in $O(nk \log n)$ time. Since $\overline{m} \leq n^2$, DET_ALGO(G) runs in $O(n^2 k)$ time. ◀

Algorithm 4.2 CONSTRUCT_COLORING(i)

/*For a detailed version of this procedure, see Appendix A.5.
All data structures are assumed to be global.
Notational Note:
Let $\mathcal{C}_i^{v_z}$ denote the partial coloring at the stage when we have colored the vertices v_n to v_z. Let $\mathcal{C}_i^{v_z=\chi_c}$ denote the partial coloring that results if we extend $\mathcal{C}_i^{v_z+1}$ by assigning color χ_c to v_z.*/

for $y = n$ to 1 **do**
  for each $\chi_c \in \{\chi_1, \ldots, \chi_{8k}\}$ **do**
    1. Compute $|X(v_y, \mathcal{C}_i^{v_y=\chi_c})|$, $|Y(v_y, \mathcal{C}_i^{v_y=\chi_c})|$, and $|Z(v_y, \mathcal{C}_i^{v_y=\chi_c})|$ as per equations (2),(3), and (4) respectively.
    if $|X(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{3}{4}|BNN_i[v_y]|$ and $|Y(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{3}{4}|Z(v_y, \mathcal{C}_i^{v_y=\chi_c})|$ **then**
      2. SET $\mathcal{C}_i^{v_y} \leftarrow \mathcal{C}_i^{v_y=\chi_c}$ (i.e. SET $\mathcal{C}_i(v_y) \leftarrow \chi_c$).
      3. SET $Y(v_y, \mathcal{C}_i^{v_y}) \leftarrow Y(v_y, \mathcal{C}_i^{v_y=\chi_c})$
      4. BREAK.
    end if
  end for
end for
for $y = 1$ to n **do**
  5. Compute $W(v_y, \mathcal{C}_i)$ as per equation (1)
  6. SET $Y(v_y, \mathcal{C}_i) \leftarrow Y(v_y, \mathcal{C}_i^{v_1})$
end for
7. Return $\mathcal{C}_i$.

5 Boxicity and Crossing Number

5.1 A Useful Lemma

For a graph H, let $V_A, V_B \subseteq V(H)$ such that $V(H) = V_A \uplus V_B$. Let $S_B(H)$ be the graph with $V(S_B(H)) = V(H)$ and $E(S_B(H)) = E(H) \setminus \{(u,v) \mid u, v \in V_B\}$. In other words, $S_B(H)$ is obtained from H by making V_B a stable set. Let H_B be the subgraph of H induced on V_B.

▶ **Lemma 10.** $box(H) \leq 2box(S_B(H)) + box(H_B)$.

Proof. See Appendix A.6. ◀

5.2 Crossing Number

Crossing number of a graph G, denoted as $CR(G)$, is the minimum number of crossing pairs of edges, over all drawings of G in the plane. A graph G is planar if and only if $CR(G) = 0$. Determination of the crossing number is an NP-complete problem.

The following theorem is due to Pach and Tóth [14]

▶ **Theorem 11.** *For a graph G with n vertices and $m \geq 7.5n$ edges, $CR(G) \geq \frac{1}{33.75}\frac{m^3}{n^2}$, and this estimate is tight upto a constant factor.*

The following claim directly follows from the above theorem.

▶ Claim 12. For a graph G, if $CR(G) \leq t$, then $d_{av}(G) \leq 2(\frac{33.75t}{n})^{1/3} + 15$.

Proof. If $m < 7.5n$, then $d_{av} < 15$. Otherwise, we have $m \leq (33.75n^2t)^{1/3}$ implying that $d_{av} \leq 2(\frac{33.75t}{n})^{1/3}$. ◀

We now prove the main theorem of this section.

▶ **Theorem 13.** *For a graph G with $CR(G) = t$, $box(G) \leq 66 \cdot t^{\frac{1}{4}} \lceil \log 4t \rceil^{\frac{3}{4}} + 6$.*

Proof. Consider a drawing P of G with t crossings. We say a vertex v *participates* in a given crossing in P, if at least one of the edges of the given crossing is incident on v.

Partition the vertices of G into two parts, namely V_A and V_B, such that $V_B = \{v \in V(G) \mid v \text{ participates in some crossing in } P\}$ and $V_A = V(G) \setminus V_B$. Let $S_B(G)$ be the graph with $V(S_B(G)) = V(G)$ and $E(S_B(G)) = E(G) \setminus \{(u,v) \mid u, v \in V_B\}$. In other words, $S_B(G)$ is obtained from G by making V_B a stable set. Let G_B be the subgraph of G induced on V_B. Then by Lemma 10,

$$box(G) \leq 2box(S_B(G)) + box(G_B).$$

Observe that $S_B(G)$ is a planar graph and hence its boxicity is at most 3 (see [17]). Therefore, $box(G) \leq 6 + box(G_B)$. For ease of notation, let $H \equiv G_B$. Then,

$$box(G) \leq 6 + box(H). \tag{5}$$

We have $CR(H) = CR(G) = t$. Let $n = |V(H)|$ and $m = |E(H)|$. At most 4 vertices participate in a given crossing. Since each vertex in H participates in some crossing in P, we get

$$n \leq 4t.$$

Let $V(H) = \{v_1, v_2, \ldots, v_n\}$. Let $v_1, v_2, \ldots, v_n$ be an ordering of the vertices of H, such that for each $i \in [n]$, $deg_{H_i}(v_i) \leq deg_{H_i}(v), \forall v \in V(H_i)$, where H_i denotes the subgraph of H induced on vertex set $\{v_i, v_{i+1}, \ldots, v_n\}$. Let $k = \left(\frac{33.75}{3}\right)^{\frac{1}{4}} \left(\frac{t}{\lceil \log 4t \rceil}\right)^{\frac{1}{4}}$. Let $x = \min(\{i \in [n] \mid deg_{H_i}(v_i) > k\})$. Partition $V(H)$ into two parts, namely $V_C = \{v_1, v_2, \ldots, v_{x-1}\}$ and $V_D = \{v_x, v_{x+1}, \ldots, v_n\}$. Let $S_D(H)$ be the graph with $V(S_D(H)) = V(H)$ and $E(S_D(H)) = E(H) \setminus \{(u,v) \mid u, v \in V_D\}$. In other words, $S_D(H)$ is obtained from H by making V_D a stable set. Let H_D be the subgraph of H induced on V_D. Then by Lemma 10,

$$box(H) \leq 2box(S_D(H)) + box(H_D).$$

Note that $S_D(H)$ is k-degenerate. If $k = 1$, then $S_D(H)$ is a forest and hence its boxicity is at most 2. Suppose $k > 1$. Then by Theorem 4, $box(S_D(H)) \leq cub(S_D(H)) \leq (k + 2)\lceil 2e \log n \rceil \leq 12k\lceil \log(4t) \rceil \leq 12 \left(\frac{33.75}{3}\right)^{\frac{1}{4}} t^{\frac{1}{4}} \lceil \log 4t \rceil^{\frac{3}{4}}$. Thus we have,

$$box(H) \leq 24 \left(\frac{33.75}{3}\right)^{\frac{1}{4}} t^{\frac{1}{4}} \lceil \log 4t \rceil^{\frac{3}{4}} + box(H_D). \tag{6}$$

Since $H_D \equiv H_x$, v_x is a minimum degree vertex of H_D. Therefore, $d_{av}(H_D) > deg_{H_D}(v_x) > k$. Then by Claim 12, we have

$$k = \left(\frac{33.75}{3}\right)^{\frac{1}{4}} \left(\frac{t}{\lceil \log 4t \rceil}\right)^{\frac{1}{4}} < d_{av}(H_D) \leq 2\left(\frac{33.75t}{|V(H_D)|}\right)^{1/3} + 15.$$

From this, we get $|V(H_D)| \leq 48^{\frac{3}{4}}(33.75t)^{\frac{1}{4}} \lceil \log 4t \rceil^{\frac{3}{4}}$. Since boxicity of a graph is at most half the number of its vertices[15] , we get $box(H_D) \leq \frac{48^{\frac{3}{4}}(33.75t)^{\frac{1}{4}} \lceil \log 4t \rceil^{\frac{3}{4}}}{2}$. Substituting this in Inequality 6, we get

$$box(H) \leq 66t^{\frac{1}{4}} \lceil \log 4t \rceil^{\frac{3}{4}}$$

Therefore from Inequality 5 ,we get

$$box(G) \leq 66t^{\frac{1}{4}}\lceil \log 4t \rceil^{\frac{3}{4}} + 6.$$

◀

5.2.1 Tightness of Theorem 13:

We know that, for any graph G on n vertices and m edges, $CR(G) \leq m(m-1)/2 \leq m^2 \leq n^4$. Let $G \equiv K_{2,2,\ldots,2}$ denote the complete $\frac{n}{2}$-partite graph with 2 vertices in each part and let $t = CR(G)$. From [15], we know that $box(G) = \lfloor \frac{n}{2} \rfloor \geq \lfloor \frac{t^{1/4}}{2} \rfloor$. Therefore, the bound given by Theorem 13 is tight upto a factor of $O((\log t)^{\frac{3}{4}})$.

References

1 Abhijin Adiga, Diptendu Bhowmick, and L. Sunil Chandran. Boxicity and poset dimension. In *COCOON*, pages 3–12, 2010.
2 L. Sunil Chandran, Mathew C. Francis, and Naveen Sivadasan. Representing graphs as the intersection of axis-parallel cubes. *MCDES-2008, IISc Centenary Conference*, available at *CoRR*, abs/cs/0607092, 2006.
3 L. Sunil Chandran, Mathew C. Francis, and Naveen Sivadasan. Boxicity and maximum degree. *Journal of Combinatorial Theory, Series B*, 98(2):443–445, March 2008.
4 L. Sunil Chandran, Mathew C. Francis, and Naveen Sivadasan. Geometric representation of graphs in low dimension using axis parallel boxes. *Algorithmica*, 56(2):129–140, 2010.
5 L. Sunil Chandran and K. Ashik Mathew. An upper bound for cubicity in terms of boxicity. *Discrete Mathematics*, In Press, Corrected Proof, doi:10.1016/j.disc.2008.04.011, 2008.
6 L. Sunil Chandran, Rogers Mathew, and Naveen Sivadasan. Boxicity of line graphs. *CoRR*, abs/1009.4471, 2010.
7 L. Sunil Chandran and Naveen Sivadasan. Boxicity and treewidth. *Journal of Combinatorial Theory, Series B*, 97(5):733–744, September 2007.
8 M. B. Cozzens. Higher and multidimensional analogues of interval graphs. Ph. D. thesis, Rutgers University, New Brunswick, NJ, 1981.
9 M. B. Cozzens and F. S. Roberts. Computing the boxicity of a graph by covering its complement by cointerval graphs. *Discrete Applied Mathematics*, 6:217–228, 1983.
10 Louis Esperet. Boxicity of graphs with bounded degree. *European Journal of Combinatorics*, doi:10.1016/j.ejc.2008.10.003, 2008.
11 J. Kratochvil. A special planar satisfiability problem and a consequence of its NP–completeness. *Discrete Applied Mathematics*, 52:233–252, 1994.
12 H. Maehara. Sphericity exceeds cubicity for almost all complete bipartite graphs. *Journal of Combinatorial Theory, Series B*, 40(2):231–235, April 1986.
13 T.S. Michael and Thomas Quint. Sphericity, cubicity, and edge clique covers of graphs. *Discrete Applied Mathematics*, 154(8):1309–1313, May 2006.
14 János Pach and Géza Tóth. Graphs drawn with few crossings per edge. *Combinatorica*, 17(3):427–439, 1997.
15 F. S. Roberts. *Recent Progresses in Combinatorics*, chapter On the boxicity and cubicity of a graph, pages 301–310. Academic Press, New York, 1969.
16 E. R. Scheinerman. Intersection classes and multiple intersection parameters. Ph. D. thesis, Princeton University, 1984.
17 C. Thomassen. Interval representations of planar graphs. *Journal of Combinatorial Theory, Series B*, 40:9–20, 1986.

18 W.T. Trotter. *Combinatorics and partially ordered sets: Dimension theory*. Johns Hopkins Univ Pr, 2001.

19 Mihalis Yannakakis. The complexity of the partial order dimension problem. *SIAM Journal on Algebraic Discrete Methods*, 3:351–358, 1982.

20 Z. Füredi and J. Kahn. On the dimensions of ordered sets of bounded degree. *Order*, 3(1):15–20, 1986.

A Appendix

A.1 Procedure CONSTRUCT_UNIT_INTERVAL_GRAPHS()

Algorithm A.1 CONSTRUCT_UNIT_INTERVAL_GRAPHS()

/*All data structures are assumed to be global. */

1. INITIALIZE $L(f_{i,j}(v_y)) \leftarrow 0, R(f_{i,j}(v_y)) \leftarrow n, \forall y \in [n], i \in \alpha, j \in [8k]$

for $i = 1$ to α **do**
  for $y = n$ to 1 **do**
    2. SET $j \leftarrow c$, such that $\mathcal{C}_i(v_y) = \chi_c$
    3. SET $L(f_{i,j}(v_y)) \leftarrow y + n$
    4. SET $R(f_{i,j}(v_y)) \leftarrow y + 2n$
    for each $v \in N_G^b(v_y)$ **do**
      if $(\mathcal{C}_i(v) \neq j) \cap (L(f_{i,j}(v)) = 0)$ **then**
        5. SET $L(f_{i,j}(v)) \leftarrow y$
        6. SET $R(f_{i,j}(v)) \leftarrow y + n$
      end if
    end for
  end for
end for

7. Output $f_{i,j}(v_y), \forall y \in [n], i \in \alpha, j \in [8k]$

A.2 The proof of Lemma 5

Proof. The statement of the lemma is obvious if the BREAK statement in Step 4 of CONSTRUCT_COLORING(i) is executed, for every $i \in [\alpha]$ and $v_y \in V(G)$. In order to prove that the BREAK statement will be executed, it is sufficient to show that there exists a color $\chi_c \in \{\chi_1, \ldots, \chi_{8k}\}$ such that $|X(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{3}{4}|BNN_i[v_y]|$ and $|Y(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{3}{4}|Z(v_y, \mathcal{C}_i^{v_y=\chi_c})|$. Since the vertices in $Z(v_y, \mathcal{C}_i^{v_y=\chi_c})$ or $Z(v_y, \mathcal{C}_i)$ do not depend on the colors given to $v_1, \ldots v_y$, we have $Z(v_y, \mathcal{C}_i^{v_y=\chi_c}) = Z(v_y, \mathcal{C}_i)$. Hence, $Z(v_y, \mathcal{C}_i^{v_y=\chi_c})$ and $Z(v_y, \mathcal{C}_i)$ can be used interchangeably.

Let $A = BNN_i[v_y] \times Z(v_y, \mathcal{C}_i)$. Let $< v_x, v_z >$ be an element of A. We say a color χ_c is *good* for $< v_x, v_z >$, if $v_x \in X(v_y, \mathcal{C}_i^{v_y=\chi_c})$ and $v_z \in Y(v_y, \mathcal{C}_i^{v_y=\chi_c})$. In other words, χ_c is *good* for $< v_x, v_z >$, if both S_{xy} and T_{yz} are favorably colored in $\mathcal{C}_i^{v_y=\chi_c}$. S_{xy} is favorably colored in $\mathcal{C}_i^{v_y=\chi_c}$, if $\chi_c \notin P$, where $P = \{\mathcal{C}_i^{v_y=\chi_c}(v_w) \mid v_w \in N_G^f(v_x), v_y <_{\mathcal{D}} v_w\}$. Since $|N_G^f(v_x)| \leq k$, $|P| \leq k$. Therefore, there are at least $8k - k = 7k$ possible values that χ_c can take such that S_{xy} is favorably colored in $\mathcal{C}_i^{v_y=\chi_c}$. For T_{yz} also to be favorably colored in $\mathcal{C}_i^{v_y=\chi_c}$, the only thing required is that $\chi_c \neq \mathcal{C}_i^{v_y=\chi_c}(v_z)$, since $v_z \in Z(v_y, \mathcal{C}_i)$ and therefore S_{yz} is already favorably colored. This implies that there are at least $7k-1$ possible values that χ_c can take such that both S_{xy} and T_{yz} are favorably colored in $\mathcal{C}_i^{v_y=\chi_c}$. In other words, there are at least $7k-1$ *good* colors for $< v_x, v_z >$. Thus for each element in A, there are at least $7k-1$ colors *good* for it. For each color $\chi_j \in \{\chi_1, \ldots, \chi_{8k}\}$, let $S^j = \{< v_x, v_z > \in A \mid \chi_j$ is *good* for $< v_x, v_z >\} = X(v_y, \mathcal{C}_i^{v_y=\chi_j}) \times Y(v_y, \mathcal{C}_i^{v_y=\chi_j})$. Since there are at least $(7k-1)$ colors *good* for each element in A, $\Sigma_{j\in[8k]}|S^j| \geq (7k-1)|A|$. Then by pigeonhole principle, there exists a $c \in [8k]$ such that $|S^c| = |X(v_y, \mathcal{C}_i^{v_y=\chi_c})| \cdot |Y(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{(7k-1)}{8k}|A| = \frac{7k-1}{8k}|BNN_i[v_y]| \cdot |Z(v_y, \mathcal{C}_i)| \geq \frac{3}{4}|BNN_i[v_y]| \cdot |Z(v_y, \mathcal{C}_i)|$ elements of A. In other words, $|X(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{3}{4}|BNN_i[v_y]|$ and $|Y(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{3}{4}|Z(v_y, \mathcal{C}_i^{v_y=\chi_c})|$. ◀

A.3 The proof of Lemma 6

Proof. From step 8 of DET_ALGO(G), we have $|FNN_{i+1}[v_y]| = |FNN_i[v_y]| - |Y(v_y, \mathcal{C}_i)| \leq |FNN_i[v_y]| - \frac{3}{4}|Z(v_y, \mathcal{C}_i)|$ (using Lemma 5). Taking summation over all $y \in [n]$, we get $\overline{m}_{i+1} \leq \overline{m}_i - \frac{3}{4}\Sigma_{y\in[n]}|Z(v_y, \mathcal{C}_i)| = \overline{m}_i - \frac{3}{4}\Sigma_{y\in[n]}|X(v_y, \mathcal{C}_i)|$. The last equality comes from the fact that both $\Sigma_{y\in[n]}|X(v_y, \mathcal{C}_i)|$ and $\Sigma_{y\in[n]}|Z(v_y, \mathcal{C}_i)|$ represent the number of HOPEFUL non-edges in G with respect to colorings $\mathcal{C}_1, \ldots, \mathcal{C}_i$. From Lemma 5, we have $|X(v_y, \mathcal{C}_i)| \geq \frac{3}{4}|BNN_i[v_y]|$. Therefore, $\overline{m}_{i+1} \leq \overline{m}_i - (\frac{3}{4})^2\Sigma_{y\in[n]}|BNN_i[v_y]|$. Since $\Sigma_{y\in[n]}|BNN_i[v_y]| = \Sigma_{y\in[n]}|FNN_i[v_y]|$, we get $\overline{m}_{i+1} \leq \overline{m}_i - (\frac{3}{4})^2\Sigma_{y\in[n]}|FNN_i[v_y]| = \overline{m}_i - \frac{9}{16}\overline{m}_i = \frac{7}{16}\overline{m}_i$. ◀

A.4 The proof of Lemma 8

Proof. A detailed description of the procedure is given in Section A.5. To implement the procedure efficiently, we make use of an $(n \times 8k)$ $0-1$ matrix, hereafter called *FNC* (Forward Neighbor Color), and two $(n \times n)$ $0-1$ matrices named *HOPE_MATRIX* and *DONE_MATRIX* respectively. At the beginning of the procedure each of these matrices have all entries set to 0. As the procedure progresses, we change some of the entries to 1 in such a way that,
$\forall w \in [n], j \in [8k], FNC[w][j] = 1 \iff \exists v_z \in N_G^f(v_w)$ such that v_z is already colored by the procedure with color χ_j.
$\forall w, z \in [n], v_w \in BNN_i[v_z], HOPE\_MATRIX[w][z] = 1 \iff S_{wz}$ is already favorably colored by the procedure.
$\forall w, z \in [n], v_w \in BNN_i[v_z], DONE\_MATRIX[w][z] = 1 \iff T_{wz}$ is already favorably colored by the procedure.

In order for the above matrices to satisfy their respective properties, the only thing that needs to be done is to update these matrices at each stage of the procedure. Consider the stage at which the procedure is extending partial coloring $\mathcal{C}_i^{v_y+1}$ to $\mathcal{C}_i^{v_y}$ by assigning color χ_c to v_y. At this stage, the matrices *FNC*, *HOPE_MATRIX* and *DONE_MATRIX* are updated as described in steps 11(a), 12(a) and 13(a) respectively. Note that this can be done in $O(|BNN_i[v_y]| + |FNN_i[v_y]| + |N_G^b(v_y)|)$ time. Steps 4(a)-(b), 5(a)-(b) and 6(a)-(b) compute $X(v_y, \mathcal{C}_i^{v_y=\chi_c}), Y(v_y, \mathcal{C}_i^{v_y=\chi_c})$ and $Z(v_y, \mathcal{C}_i^{v_y=\chi_c})$ respectively in $O(|BNN_i[v_y]| + |FNN_i[v_y]|)$ time. Computing $W(v_y, \mathcal{C}_i)$ is done in step 15 (a)–(b) in $O(|BNN_i[v_y]|)$ time.

Since steps 4 to 14, in the worst case, are run for each $v_y \in V(G), \chi_c \in \{\chi_1, \ldots, \chi_{8k}\}$, the procedure runs in $O(k(\Sigma_{y\in[n]}(|BNN_i[v_y]| + |FNN_i[v_y]|) + \Sigma_{y\in[n]}|N_G^b(v_y)|))$ time. We know that $\Sigma_{y\in[n]}(|BNN_i[v_y]| + |FNN_i[v_y]|) = 2\overline{m}_i$ and $\Sigma_{y\in[n]}|N_G^b(v_y)| = m \leq kn$. Hence the Lemma. ◀

A.5 A Detailed version of procedure CONSTRUCT_COLORING(i)

Algorithm A.2 CONSTRUCT_COLORING(i) /* detailed */

/*All data structures are assumed to be global.
Notational Note:
Let $\mathcal{C}_i^{v_z}$ denote the partial coloring at the stage when we have colored the vertices v_n to v_z. Let $\mathcal{C}_i^{v_z=\chi_c}$ denote the partial coloring that results if we extend $\mathcal{C}_i^{v_{z+1}}$ by assigning color χ_c to v_z. */

1. Initialize $FNC[w][j] \leftarrow 0, \forall w \in [n], j \in [8k]$
2. Initialize $HOPE\_MATRIX[w][z] \leftarrow 0, \forall w, z \in [n]$
3. Initialize $DONE\_MATRIX[w][z] \leftarrow 0, \forall w, z \in [n]$

for $y = n$ to 1 **do**
  for each $\chi_c \in \{\chi_1, \ldots, \chi_{8k}\}$ **do**
    4. Compute $X(v_y, \mathcal{C}_i^{v_y=\chi_c})$ /*as described in steps (a) and (b) below */
      (a) Initialize $X(v_y, \mathcal{C}_i^{v_y=\chi_c}) \leftarrow \emptyset$
      (b) $\forall v_x \in BNN_i[v_y]$, if $FNC[x][c] = 0$, then
      SET $X(v_y, \mathcal{C}_i^{v_y=\chi_c}) \leftarrow X(v_y, \mathcal{C}_i^{v_y=\chi_c}) \cup \{v_x\}$
    5. Compute $Y(v_y, \mathcal{C}_i^{v_y=\chi_c})$ /*as described in steps (a) and (b) below */
      (a) Initialize $Y(v_y, \mathcal{C}_i^{v_y=\chi_c}) \leftarrow \emptyset$
      (b) $\forall v_z \in FNN_i[v_y]$, if ($HOPE\_MATRIX[y][z] = 1$) and
      $(\mathcal{C}_i^{v_y=\chi_c}(v_z) \neq \chi_c)$, then SET $Y(v_y, \mathcal{C}_i^{v_y=\chi_c}) \leftarrow Y(v_y, \mathcal{C}_i^{v_y=\chi_c}) \cup \{v_z\}$
    6. Compute $Z(v_y, \mathcal{C}_i^{v_y=\chi_c})$ /*as described in steps (a) and (b) below */
      (a) Initialize $Z(v_y, \mathcal{C}_i^{v_y=\chi_c}) \leftarrow \emptyset$
      (b) $\forall v_z \in FNN_i[v_y]$, if $HOPE\_MATRIX[y][z] = 1$,
      then SET $Z(v_y, \mathcal{C}_i^{v_y=\chi_c}) \leftarrow Z(v_y, \mathcal{C}_i^{v_y=\chi_c}) \cup \{v_z\}$
    if $|X(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{3}{4}|BNN_i[v_y]|$ and $|Y(v_y, \mathcal{C}_i^{v_y=\chi_c})| \geq \frac{3}{4}|Z(v_y, \mathcal{C}_i^{v_y=\chi_c})|$ **then**
      7. SET $\mathcal{C}_i^{v_y} \leftarrow \mathcal{C}_i^{v_y=\chi_c}$ (i.e. SET $\mathcal{C}_i(v_y) \leftarrow \chi_c$).
      8. SET $X(v_y, \mathcal{C}_i^{v_y}) \leftarrow X(v_y, \mathcal{C}_i^{v_y=\chi_c})$
      9. SET $Y(v_y, \mathcal{C}_i^{v_y}) \leftarrow Y(v_y, \mathcal{C}_i^{v_y=\chi_c})$
      10. SET $Z(v_y, \mathcal{C}_i^{v_y}) \leftarrow Z(v_y, \mathcal{C}_i^{v_y=\chi_c})$
      11. Update FNC matrix. /* as described in step (a) below */
        (a) $\forall v_x \in N_G^b(v_y)$, SET $FNC[x][c] \leftarrow 1$
      12. Update $HOPE\_MATRIX$ /* as described in step (a) below */
        (a) $\forall v_x \in X(v_y, \mathcal{C}_i^{v_y})$, SET $HOPE\_MATRIX[x][y] \leftarrow 1$
      13. Update $DONE\_MATRIX$ /* as described in step (a) below */
        (a) $\forall v_z \in Y(v_y, \mathcal{C}_i^{v_y})$, SET $DONE\_MATRIX[y][z] \leftarrow 1$
      14. BREAK.
    end if
  end for
end for
for $y = 1$ to n **do**
  15. Compute $W(v_y, \mathcal{C}_i)$ /*as described in steps (a) and (b) below */
    (a) Initialize $W(v_y, \mathcal{C}_i) \leftarrow \emptyset$
    (b) $\forall v_x \in BNN_i[v_y]$, if $DONE\_MATRIX[x][y] = 1$, then
    SET $W(v_y, \mathcal{C}_i) \leftarrow W(v_y, \mathcal{C}_i) \cup \{v_x\}$
  16. SET $Y(v_y, \mathcal{C}_i) \leftarrow Y(v_y, \mathcal{C}_i^{v_1})$
end for
17. Return $\mathcal{C}_i$.

A.6 The proof of Lemma 10

Proof. Let $C_B(H)$ be the graph with $V(C_B(H)) = V(H)$ and $E(C_B(H)) = E(H) \cup \{(u,v) \mid u,v \in V_B\}$. In other words, $C_B(H)$ is obtained from H by making V_B a clique. Let H' be the graph with $V(H') = V(H)$ and $E(H') = E(H) \cup \{(u,v) \mid u \in V_A\}$. Observe that

$$H = C_B(H) \cap H'.$$

In conjunction with Lemma 1, this implies that

$$box(H) \leq box(C_B(H)) + box(H'). \tag{7}$$

▶ Claim 14. $box(C_B(H)) \leq 2box(S_B(H))$.

Proof of this claim is very similar to the proof of Lemma 3 in [6] and hence we only give a brief outline of it here. Assume $box(S_B(H)) = r$. Then by Lemma 1, there exist r interval graphs $I_1, \ldots, I_r$ such that $S_B(H) = I_1 \cap I_2 \cap \cdots \cap I_r$. For each $i \in [r]$, let f_i denote an interval representation of I_i. From these r interval graphs we construct $2r$ interval graphs $I'_1, I'_2, \ldots, I'_r, I''_1, I''_2, \ldots, I''_r$ as outlined below. Let f'_i, f''_i denote interval representations of I'_i and I''_i respectively, where $i \in [r]$.

Construction of f'_i :

$$\begin{aligned} \forall u \in V_A,\ f'_i(u) &= f_i(u). \\ \forall u \in V_B,\ f'_i(u) &= [\min_{v \in V_B}(L(f_i(v))), R(f_i(u))]. \end{aligned}$$

Construction of f''_i :

$$\begin{aligned} \forall u \in A,\ f''_i(u) &= f_i(u). \\ \forall u \in B,\ f''_i(u) &= [L(f_i(u)), \max_{v \in V_B}(R(f_i(v)))]. \end{aligned}$$

We leave it to the reader to verify that $C_B(H) = \bigcap_{i=1}^{r}(I'_i \cap I''_i)$.

▶ Claim 15. $box(H') \leq box(H_B)$.

Clearly, H' is obtained from H_B by adding universal vertices one after the other. Since adding a universal vertex to a graph does not increase its boxicity, $box(H') \leq box(H_B)$.

Combining Inequality 7, Claim 14 and Claim 15, we get $box(H) \leq 2box(S_B(H)) + box(H_B)$. ◀