Graph Realizations: Maximum and Minimum Degree in Vertex Neighborhoods

Amotz Bar-Noy, Keerti Choudhary, David Peleg, Dror Rawitz

Abstract

The classical problem of degree sequence realizability asks whether or not a given sequence of n positive integers is equal to the degree sequence of some n-vertex undirected simple graph. While the realizability problem of degree sequences has been well studied for different classes of graphs, there has been relatively little work concerning the realizability of other types of information profiles, such as the vertex neighborhood profiles.

In this paper, we initiate the study of neighborhood degree profiles, wherein, our focus is on the natural problem of realizing maximum and minimum neighborhood degrees. More specifically, we ask the following question: "Given a sequence D of n non-negative integers $0 \le d_1 \le \cdots \le d_n$, does there exist a simple graph with vertices v_1, \ldots, v_n such that for every $1 \le i \le n$, the maximum (resp. minimum) degree in the neighborhood of v_i is exactly d_i ?"

We provide in this work various results for both maximum as well as minimum neighborhood degree for general n vertex graphs. Our results are first of its kind that studies extremal neighborhood degree profiles. For maximum neighborhood degree profiles, we provide a *complete realizability criteria*. In comparison, we observe that the minimum neighborhood profiles are not so well-behaved, for these our necessary and sufficient conditions for realizability differ by a factor of at most two.

1 Introduction

In many application domains involving networks, it is common to view vertex degrees as a central parameter, providing useful information concerning the relative significance (and in certain cases, centrality) of each vertex with respect to the rest of the network, and consequently useful for understanding the network's basic properties. Given an n-vertex graph G with adjacency matrix Adj(G), its $degree\ sequence$ is a sequence consisting of its vertex degrees,

$$DEG(G) = (d_1, \ldots, d_n).$$

Given a graph G or its adjacency matrix, it is easy to extract the degree sequence. An interesting *dual* problem, sometimes referred to as the *realization* problem, concerns a situation where given a sequence of nonnegative integers D, we are asked whether there exists a graph whose degree sequence conforms to D. A sequence for which there exists a realization is called a *graphic* sequence. Erdös and Gallai [9] gave a necessary and sufficient condition for deciding whether a given sequence of integers is graphic (also implying an O(n) decision algorithm). Havel and Hakimi [10, 11] gave a recursive algorithm that given a sequences of integers computes in O(m) time a realizing m-edge graph, or proves that the sequence is not graphic.

^{*}City University of New York (CUNY), USA. E-mail: amotz@sci.brooklyn.cuny.edu

[†]Tel Aviv University, Israel. E-mail: keerti.choudhary@cs.tau.ac.il

[‡]Weizmann Institute of Science, Israel. E-mail: david.peleg@weizmann.ac.il

[§]Bar Ilan University, Israel. E-mail: dror.rawitz@biu.ac.il

Over the years, various extensions of the degree realization problem were studied as well, cf. [1, 3, 19], concerning different characterizations of degree-profiles. The motivation underlying the current paper is rooted in the observation that realization questions of a similar nature pose themselves naturally in a large variety of *other* application contexts, where given *some* type of information profile specifying the desired vertex properties (be it concerning degrees, distances, centrality, or any other property of significance), it can be asked whether there exists a graph conforming to the specified profile. Broadly speaking, this type of investigation may arise, and find potential applications, both in scientific contexts, where the information profile reflects measurement results obtained from some natural network of unknown structure, and the goal is to obtain a model that may explain these measurements, and in engineering contexts, where the information profile represents a specification with some desired properties, and the goal is to find an implementation in the form of a network conforming to that specification.

This basic observation motivates a vast research direction, which was little studied over the last five decades. In this paper we make a step towards a systematic study of one specific type of information profiles, concerning *neighborhood degree* profiles. Such profiles are of theoretical interest in context of social networks (where degrees often reflect influence and centrality, and consequently neighboring degrees reflect "closeness to power"). Neighborhood degrees were considered before in [5], where the profile associated with each vertex i is the *list* of degrees of all vertices in i's neighborhood. In contrast, we focus here on "single parameter" profiles, where the information associated with each vertex relates to a single degree in its neighborhood. Two first natural problems in this direction concern the *maximum* and *minimum* degrees in the vertex neighborhoods. For each vertex i, let d'_i (respectively, d''_i) denote the maximum (resp., minimum) vertex degree in i's neighborhood. Then MAXNDEG $(G) = (d'_1, \ldots, d'_n)$ (resp., MINNDEG $(G) = (d''_1, \ldots, d''_n)$) is the maximum (resp., minimum) neighborhood degree profile of G. The same realizability questions asked above for degree sequences can be posed for neighborhood degree profiles as well. This brings us to the following central question of our work:

Question. Can we efficiently compute for a given sequence $D = (d_1, \ldots, d_n)$ of nonnegative integers an n-vertex graph G (if exists) such that the maximum (resp. minimum) degree in the neighborhood of i-th vertex in G is exactly equal to d_i ? Moreover, is there a closed-form characterization for all n-length realizable sequences?

Our Contributions For simplicity, we represent the input vector D alternatively in a more compact format as $\sigma = (d_\ell^{n_\ell}, \cdots, d_1^{n_1})$, where n_i 's are positive integers with $\sum_{i=1}^\ell n_i = n$; here the specification requires that G contains exactly n_i vertices whose minimum (resp. maximum) degree in neighborhood is d_i . We may assume that $d_\ell > d_{\ell-1} > \cdots > d_1 \ge 1$ (noting that vertices with max/min degree zero are necessarily singletons and can be handled separately).

(a) Minimum Neighborhood degree: In Section 3 we show the following necessary and sufficient conditions for $\sigma = (d_{\ell}^{n_{\ell}}, \cdots, d_{1}^{n_{1}})$ to be MINNDEG realizable.

The necessary condition is that for each $i \in [1, \ell]$,

$$d_i \le n_1 + n_2 + \ldots + n_i - 1$$
, and (NC1)

$$d_{\ell} \le \left\lfloor \frac{n_1 d_1}{d_1 + 1} \right\rfloor + \left\lfloor \frac{n_2 d_2}{d_2 + 1} \right\rfloor + \ldots + \left\lfloor \frac{n_{\ell} d_{\ell}}{d_{\ell} + 1} \right\rfloor. \tag{NC2}$$

The sufficient condition is that for each $i \in [1, \ell]$,

$$d_i \le \left\lfloor \frac{n_1 d_1}{d_1 + 1} \right\rfloor + \left\lfloor \frac{n_2 d_2}{d_2 + 1} \right\rfloor + \ldots + \left\lfloor \frac{n_i d_i}{d_i + 1} \right\rfloor. \tag{SC}$$

Remark 1. For any sequence $\sigma = (d_{\ell}^{n_{\ell}}, \cdots, d_{1}^{n_{1}})$ satisfying the first necessary condition (NC1), the sequence $\sigma^{\gamma} = (d_{\ell}^{\lceil \gamma n_{\ell} \rceil}, \dots, d_{1}^{\lceil \gamma n_{1} \rceil})$, where $\gamma = (d_{1} + 1)/d_{1}$ satisfies the sufficient condition (SC), thus our necessary and sufficient conditions differ by a factor of at most 2 in the n_{i} 's.

Remark 2. For ℓ bounded by 3, we show that $\sigma = (d_{\ell}^{n_{\ell}}, \cdots, d_{1}^{n_{1}})$ is MINNDEG-realizable if and only if along with (NC1) and (NC2) following is satisfied:

Either
$$d_2 \le \left\lfloor \frac{n_1 d_1}{d_1 + 1} \right\rfloor + \left\lfloor \frac{n_2 d_2}{d_2 + 1} \right\rfloor$$
, or $d_3 + 1 \le n_1 + n_2 + n_3 - \left(1 + \left\lceil \frac{d_2 - n_2}{d_1} \right\rceil \right)$ (NC3)

We leave it as an open question to resolve the problem in general.

Open Question. Does there exist a closed-form characterization for realizing MINNDEG profiles for general graphs?

- (b) Maximum Neighborhood degree: We perform an extensive study of maximum neighborhood degree profiles.
 - 1. In Section 4, we obtain the necessary and sufficient conditions for $\sigma=(d_\ell^{n_\ell},\cdots,d_1^{n_1})$ to be MAXNDEG realizable.

For general graphs we obtain the following characterization.

$$d_{\ell} \leq n_{\ell} - 1$$
, and $d_1 \geq 2$ or n_1 is even

We also study the version of the problem in which the realization is required to be connected. Our characterization is as follows.

$$d_{\ell} \le n_{\ell} - 1$$
, and $d_1 \ge 2$ or $\sigma = (1^2)$.

2. Further, we consider the open neighborhoods, wherein a vertex is not counted in its own neighborhood. These are more involved and are discussed in Section 5. Our results for open neighborhood are summarised in Table 1.

Graph	Complete characterisation
Connected Graphs	$d_{\ell} \le \min\{n_{\ell}, n-1\}$
	$d_1 \geq 2 \text{ or } \sigma = (d^d, 1^1) \text{ or } \sigma = (1^2)$
	$\sigma \neq (d_{\ell}^{d_{\ell}+1}, 2^1)$
General graphs	σ can be split ¹ into two profiles σ_1 and σ_2 such that
	(i) σ_1 has a <i>connected</i> MAXNDEG-open realization, and
	(ii) $\sigma_2 = (1^{2\alpha})$ or $\sigma_2 = (d^d, 1^{2\alpha+1})$, for integers $d \ge 2, \alpha \ge 0$.

Table 1: Max-neighbouring-degree realizability for open neighborhood.

A profile $\sigma = (d_{\ell}^{n_{\ell}}, \dots, d_{1}^{n_{1}})$ is said to be split into two profiles $\sigma_{1} = (d_{\ell}^{p_{\ell}}, \dots, d_{1}^{p_{1}})$ and $\sigma_{2} = (d_{\ell}^{q_{\ell}}, \dots, d_{1}^{q_{1}})$ if $n_{i} = p_{i} + q_{i}$ for each $i \in [1, \ell]$.

3. Enumerating realizable maximum neighborhood degree profiles:

The simplicity of above characterizations enables us to enumerate and count the number of realizable profiles. This gives a way to sample uniformly a random MAXNDEG realizable profile. In contrast, counting and sampling are open problems for the traditional degree sequence realizability problem. In Appendix, we show that the number of realizable profiles of length n is $\lceil (2^{n-1} + (-1)^n)/3 \rceil$ for general graphs and 2^{n-3} for connected graphs. In comparison, the total number of non-increasing sequences of length n on the numbers $1, \ldots, n-1$ is $\Theta(4^n/\sqrt{n})$.

In Section 6, we discuss the apparent difference in difficulty between MAXNDEG and MINNDEG profiles and propose a possible explanation.

Further Related Work Many works have addressed related questions such as finding all the (non-isomorphic) graphs that realize a given degree sequence, counting all the (non-isomorphic) realizing graphs of a given degree sequence, sampling a random realization for a given degree sequence as uniformly as possible, or determining the conditions under which a given degree sequence defines a unique realizing graph (a.k.a. the *graph reconstruction* problem), cf. [7, 9, 10, 11, 12, 15, 16, 17, 18, 20]. Other works such as [6, 8, 13] studied interesting applications in the context of social networks.

To the best of our knowledge, the MAXNDEG and MINNDEG realization problems have not been explored so far. There are only two related problems that we are aware of. The first is the *shotgun assembly* problem [14], where the characteristic associated with the vertex i is some description of its neighborhood up to radius r. The second is the *neighborhood degree lists* problem [5], where the characteristic associated with the vertex i is the list of degrees of all vertices in i's neighborhood. We point out that in contrast to these studies, our MAXNDEG and MINNDEG problem applies to a more restricted profile (with a single number characterizing each vertex), and the techniques involves are totally different from those of [5, 14]. Several other realization problems are surveyed in [2, 4].

2 Preliminaries

Let H be an undirected graph. We use V(H) and E(H) to respectively denote the vertex set and the edge set of graph H. For a vertex $x \in V(H)$, let $deg_H(x)$ denote the degree of x in H. Let $N_H[x] = \{x\} \cup \{y \mid (x,y) \in E(H)\}$ be the (closed) neighborhood of x in H. For a set $W \subseteq V(H)$, we denote by $N_H(W)$, the set of all the vertices lying outside set W that are adjacent to some vertex in W, that is, $N_H(W) = (\bigcup_{w \in W} N[w]) \setminus W$. Given a vertex v in H, the minimum (resp. maximum) degree in the neighborhood of v, namely $\operatorname{MINNDEG}_H(v)$ (resp. $\operatorname{MAXNDEG}_H(v)$), is defined to be the maximum over the degrees of all the vertices in the neighborhood of v. Given a set of vertices A in a graph H, we denote by H[A] the subgraph of H induced by the vertices of A. For a set A and a vertex $x \in V(H)$, we denote by $A \cup x$ and $A \setminus x$, respectively, the sets $A \cup \{x\}$ and $A \setminus \{x\}$. When the graph is clear from context, for simplicity, we omit the subscripts H in all our notations. Finally, given two integers $i \leq j$, we define $[i,j] = \{i,i+1,\ldots,j\}$.

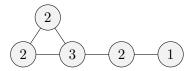


Figure 1: A MAXNDEG realization of $(3^4, 2^1)$ and a MINNDEG realization of $(2^3, 1^2)$.

A profile $\sigma=(d_\ell^{n_\ell},\cdots,d_1^{n_1})$ satisfying $d_\ell>d_{\ell-1}>\cdots>d_1>0$ is said to be MINNDEG realizable (resp. MAXNDEG realizable) if there exists a graph G on $n=n_1+\cdots+n_\ell$ vertices that for each $i\in[1,\ell]$ contains exactly n_i vertices whose MINNDEG (resp. MAXNDEG) is d_i . Equivalently, $|\{v\in V(G): \text{MINNDEG}(v)=d_i\}|=n_i$ (resp. $|\{v\in V(G): \text{MAXNDEG}(v)=d_i\}|=n_i$). The figure depicts a MAXNDEG realization of $(3^4,2^1)$ and a MINNDEG realization of $(2^3,1^2)$. (The numbers in the vertices represent their degrees.) Note that in the open neighborhoods model, the corresponding MAXNDEG and MINNDEG profiles become $(3^3,2^2)$ and $(2^4,1^1)$, respectively.

3 Realizing minimum neighborhood degree profiles

3.1 Leaders and followers

Let G=(V,E) be any graph. For any vertex $v\in V$, we define leader(v) to be a vertex in N[v] of minimum degree, if there are more than one choices we pick the leader arbitrarily. In other words, $leader(v)=\arg\min\{deg(w)\mid w\in N[v]\}$. Next let $\sigma=(d_\ell^{n_\ell}\cdots d_1^{n_1})$ be the min-degree sequence of G. We define V_i to be set of those vertices in G whose minimum-degree in the closed neighborhood is exactly d_i , so $|V_i|=n_i$. Also, let L_i be set of those vertices in G who are leader of at least one vertex in V_i , equivalently, $L_i=\{leader(v)\mid v\in V_i\}$, and denote by $L=\cup_{i=1}^\ell L_i$ the set of all the leaders in G. Observe that the sets V_1,\ldots,V_ℓ forms a partition of the vertex-set of G.

A vertex v in G is said to a *follower*, if $leader(v) \neq v$. Let $F_i = \{v \in V_i \mid v \neq leader(v)\}$ be the set of all the followers in V_i . Finally we define $R = V \setminus L$ to be the set of all the non-leaders, and $F = \bigcup_{i=1}^{\ell} F_i$ to be the set of all the followers.

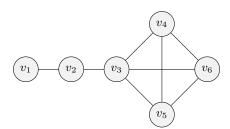


Figure 2: The unique MINNDEG-realization of the sequence $\sigma=(3^32^11^2)$. Observe that $min\text{-}deg(v_1)=min\text{-}deg(v_2)=deg(v_1)=1$, $min\text{-}deg(v_3)=deg(v_2)=2$, and $min\text{-}deg(v_i)=3$, for $i\in\{4,5,6\}$. Since $leader(v_2)=v_1$ and $leader(v_3)=v_2$, here v_2 is a leader as well as a follower.

We point here that there exist realizable sequences σ for which any graph G realizing σ and any leader function over G, the sets L and F have non-empty intersection. For example, consider the sequence $\sigma=(1^22^13^3)$ in Figure 2. It can be easily checked that σ has only one realizing graph, and in this graph, the leader-set and the follower-set are not disjoint.

We classify the sequences that admit disjoint leader and follower sets as follows.

Definition 1. A sequence $\sigma = (d_{\ell}^{n_{\ell}} \cdots d_{1}^{n_{1}})$ is said to admit a Disjoint Leader-Follower (DLF) MINNDEGrealization if there exists a graph G realizing σ and a leader function under which the sets L and F are mutually disjoint, that is, $L \cap F = \emptyset$.

3.2 Realizing uniform sequences

Lemma 1. For a sequence $\sigma = (d_{\ell}^{n_{\ell}} \cdots d_{1}^{n_{1}})$ to be MINNDEG-realizable it is necessary that $d_{1} + 1 \leq n_{1}$.

Proof. Suppose σ is MINNDEG-realizable by a graph G, then there exists at least one vertex, say w, of degree exactly d_1 in G. Now $|N[w]| = d_1 + 1$, this implies that the number of vertices v in graph G with $min\text{-}deg(v) = d_1$ must be at least $d_1 + 1$. Thus $n_1 \ge d_1 + 1$.

Lemma 2. The sequence $\sigma = (d^n)$, is MINNDEG-realizable if and only if $n \ge d + 1$.

Proof. By Lemma 1, if the sequence $\sigma=(d^n)$ is realizable then n must be at least d+1. To prove the converse, we give a realization for σ assuming $n\geq d+1$. Let $q\geq 1$ and $r\in [0,d]$ be integers satisfying n=(q)(d+1)-r. Take a set A of q vertices, namely L_i $(i\in [1,q])$, and another set B of dq vertices, namely b_{ij} $(i\in [1,q],j\in [1,d])$. Connect each L_i to vertices b_{i1},\ldots,b_{id} . So vertices in A have degree exactly d and vertices in B have in their neighborhood a vertex of degree d. Next if r>0, then we merge b_{1j} with b_{2j} , for $j\in [1,r]$, thereby reducing r vertices in B. (Notice that b_{1j} and b_{2j} exists because r>0 only if $q\geq 2$). Thus |A|+|B|=n and each vertex in A still has degree exactly d. So $|A|=\frac{n+r}{d+1}=\left\lceil\frac{n}{d+1}\right\rceil$ and $|R|=n-|A|=\left\lfloor\frac{nd}{d+1}\right\rfloor\geq d$. Finally, we add edges between each pair of vertices in B to make it a clique of size at least d; this will imply that the vertices in set B have degree at least d. It is easy to check that min-deg(v) for each $v\in A\cup B$ in our constructed graph is d.

Remark 3. Henceforth, we will use GRAPH(n,d,A,B) to denote the function that returns the edges of the graph constructed by Lemma 2 whenever $n \ge d+1$ and $|A| = \left \lceil \frac{n}{d+1} \right \rceil$, and $|R| = \left \lfloor \frac{nd}{d+1} \right \rfloor$.

3.3 Necessary and sufficient conditions for MINNDEG profiles

We start with the following theorem.

Theorem 1 (Sufficient condition SC). Any sequence $\sigma = (d_{\ell}^{n_{\ell}} \cdots d_{1}^{n_{1}})$ satisfying $d_{i} \leq \sum_{j=1}^{i} \left\lfloor \frac{n_{j}d_{j}}{d_{j}+1} \right\rfloor$, for $i \in [1,\ell]$, is MINNDEG-realizable by a graph G such that $L \cap F = \emptyset$ with respect to some leader function defined over G.

Proof. We initialize G to be an empty graph. Our algorithm proceeds in ℓ rounds. (See Algorithm 1 for a pseudo-code). In each round, we first add to G a set V_i of n_i new vertices and partition V_i into two sets L_i and R_i of sizes respectively $\left\lceil \frac{n_i}{d_i+1} \right\rceil$ and $\left\lfloor \frac{n_i d_i}{d_i+1} \right\rfloor$. Now if $n_i > d_i+1$, then we solve this round independently by adding to G all the edges returned by GRAPH (n_i, d_i, L_i, R_i) . Notice that if $n_i \leq d_i+1$, then L_i will contain only one vertex, say a_i . In such a case, we add edges between a_i and all the vertices in set R_i . Also, we add edges between a_i and any arbitrarily chosen d_i+1-n_i vertices in $\cup_{j< i} R_j$. This is possible since $d_i+1-n_i=d_i-\left\lfloor \frac{n_i d_i}{d_i+1} \right\rfloor \leq \sum_{j=1}^{i-1} \left\lfloor \frac{n_j d_j}{d_j+1} \right\rfloor = \sum_{j=1}^{i-1} |R_j|$. Finally, after the ℓ rounds are completed, we add edges between each pair of vertices in set $R=\cup_{i=1}^{\ell} R_i$ to make it a clique.

Let us now show bounds on the degree of vertices in sets L_i and R_i .

- 1. Each vertex in L_i has degree exactly d_i : Recall we add edges to vertices in L_i only in the i^{th} iteration of for loop. If $n_i > d_i + 1$, then by Lemma 2, the degree of each vertex in L_i is exactly d_i . If $|L_i| = 1$, or equivalently, $n_i \le d_i + 1$, then $|R_i| = n_i |L_i| = n_i 1$, and so degree of vertex $a_i \in L_i$ is $(n_i 1) + (d_i + 1 n_i) = d_i$.
- 2. Vertices in R have degree at least d_ℓ : For any $i \in [1,\ell]$, if $n_i > d_i + 1$, then by Lemma 2, $|R_i| = \left\lceil \frac{n_i d_i}{d_i + 1} \right\rceil$, and even in the case $n_i \le d_i + 1$, we have $|R_i| = n_i |L_i| = n_i \left\lceil \frac{n_i}{d_i + 1} \right\rceil = \left\lceil \frac{n_i d_i}{d_i + 1} \right\rceil$. Thus $|R| = \sum_{i=1}^{\ell} |R_i| = \sum_{i=1}^{\ell} \left\lceil \frac{n_i d_i}{d_i + 1} \right\rceil$ which is bounded below by d_i . Since $|R| \ge d_\ell$, and each vertex in R is adjacent to at least one vertex in $\cup_i L_i$, the degree of vertices in R is at least d_ℓ .

```
Input: A sequence \sigma = (d_\ell^{n_\ell} \cdots d_1^{n_1}) satisfying d_i \leq \sum_{j=1}^i \lfloor \frac{n_j d_j}{d_j + 1} \rfloor, for 1 \leq i \leq \ell.

1 Initialize G to be an empty graph.
2 for i = 1 to \ell do
3 | Add to G a set V_i of n_i new vertices.
4 | Partition V_i in two sets L_i, R_i such that |L_i| = \lceil \frac{n_i}{d_i + 1} \rceil and |R_i| = \lfloor \frac{n_i d_i}{d_i + 1} \rfloor.
5 | if (n_i > d_i + 1, or equivalently, |L_i| > 1) then
6 | Add to G all the edges returned by GRAPH(n_i, d_i, L_i, R_i).
7 | else if (|L_i| = 1) then
8 | Let a_i be the only vertex in L_i.
9 | Connect a_i to all vertices in R_i, and any arbitrary d_i + 1 - n_i vertices in \cup_{j < i} R_j.
10 Add edges between each pair of vertices in R_i to make it a clique.
11 Output G.
```

Algorithm 1: Computing a MINNDEG-realization for a given special σ .

We next show that for any vertex $v \in V_i$, $min\text{-}deg(v) = d_i$, where $i \in [1, \ell]$. If $v \in L_i$, then $min\text{-}deg(v) = d_i$, since each vertex in L_i has degree d_i , and is adjacent to only vertices in R which have degree at least $d_\ell \geq d_i$. If $v \in R_i$, then also $min\text{-}deg(v) = d_i$, since each vertex in R_i is adjacent to at least one vertex in L_i , and N[v] is contained in the set $R \cup (\cup_{j \geq i} L_j)$, whose vertices have degree at least d_i .

The leader function over V is as follows. For each $v \in \bigcup_{i=1}^{\ell} L_i$, we set leader(v) = v, and for each $v \in R_i$, we set leader(v) to any arbitrary neighbour of v in L_i . Since each vertex in $L = \bigcup_{i=1}^{\ell} L_i = \{leader(v) \mid v \in V\}$ is a leader of itself, the set L of leader and the set F of followers must be mutually disjoint.

We now provide a lower bound on the size of the leader set L_i .

Lemma 3. For each
$$i \in [1, \ell]$$
, we have $|L_i| \geq \left\lceil \frac{n_i}{d_i + 1} \right\rceil$.

Proof. Consider any vertex $a \in L_i$. Since $|N(a)| = d_i + 1$, vertex a can serve as leader for at most $d_i + 1$ vertices. This shows that $|L_i| \ge \frac{n_i}{d_i + 1}$. The claim follows from the fact that $|L_i|$ is an integer.

Theorem 2 (Necessary condition). For any MINNDEG-realizable sequence $\sigma=(d_\ell^{n_\ell}\cdots d_1^{n_1})$, we have

(NC1)
$$d_i \leq \left(\sum_{j=1}^i n_j\right) - 1, \text{ for } i \in [1, \ell]$$

(NC2)
$$d_{\ell} \leq \sum_{i=1}^{\ell} \left\lfloor \frac{n_i d_i}{d_i + 1} \right\rfloor.$$

Proof. Let G be a realization for σ . Let w be any vertex in G such that $deg(w) = d_i$. Then w as well as all the neighbours of w must be contained in $\bigcup_{j=1}^{i} V_j$, therefore, $d_i + 1 = |N[w]| \le |\bigcup_{j=1}^{i} V_j| = \sum_{j=1}^{i} n_j$, implying condition (NC1).

To prove condition (NC2), suppose w is a vertex in G such that $\min\text{-}deg(w) = d_\ell$. Then N[w] cannot contain vertices of degree less than d_ℓ , so $N[w] \cap L_i = \emptyset$, for each $i < \ell$. Therefore, $|N[w]| \le n - \sum_{i=1}^{\ell-1} |L_i|$. Also deg(w) must be at least d_ℓ . We thus get,

$$d_{\ell} + 1 \le |N[w]| \le n - \sum_{i=1}^{\ell-1} |L_i| = n_{\ell} + \sum_{i=1}^{\ell-1} (n_i - |L_i|) \le n_{\ell} + \sum_{i=1}^{\ell} \left\lfloor \frac{n_i d_i}{d_i + 1} \right\rfloor,$$

where the last inequality follows from Lemma 3.

If $n_{\ell} \leq d_{\ell}$, then $n_{\ell} - 1 = \lfloor \frac{n_{\ell}d_{\ell}}{d_{\ell}+1} \rfloor$, and so $d_{\ell} \leq \sum_{i=1}^{\ell} \lfloor \frac{n_{i}d_{i}}{d_{i}+1} \rfloor$. If $n_{\ell} \geq d_{\ell} + 1$, then $\frac{n_{\ell}d_{\ell}}{d_{\ell}+1} \geq d_{\ell}$ which implies $d_{\ell} \leq \lfloor \frac{n_{\ell}d_{\ell}}{d_{\ell}+1} \rfloor$ since d_{ℓ} is integral.

As a corollary of the above results, the following is immediate.

Corollary 1. The sequence $\sigma = (d_2^{n_2} d_1^{n_1})$ is MINNDEG-realizable if and only if $d_1 \leq \lfloor \frac{n_1 d_1}{d_1 + 1} \rfloor$ and $d_2 \leq \lfloor \frac{n_1 d_1}{d_1 + 1} \rfloor$ $\left| \frac{n_1 d_1}{d_1 + 1} \right| + \left| \frac{n_2 d_2}{d_2 + 1} \right|$.

Proof. Suppose $\sigma=(d_2^{n_2}d_1^{n_1})$ is realizable. Then Theorem 2 implies (i) $n_1\geq d_1+1$ which implies $d_1\leq \left\lfloor\frac{n_1d_1}{d_1+1}\right\rfloor$, and (ii) $d_\ell=d_2\leq \left\lfloor\frac{n_1d_1}{d_1+1}\right\rfloor+\left\lfloor\frac{n_2d_2}{d_2+1}\right\rfloor$. The converse follows from Theorem 1.

For a sequence $\sigma = (d_{\ell}^{n_{\ell}}, \cdots, d_{1}^{n_{1}})$, let $\gamma = (d_{1}+1)/d_{1}$. As $\lfloor \frac{\gamma n_{1}d_{1}}{d_{1}+1} \rfloor + \ldots + \lfloor \frac{\gamma n_{i}d_{i}}{d_{i}+1} \rfloor \geq n_{1}+\cdots+n_{i} \geq d_{i}$, we also have the following

Corollary 2. For any sequence $\sigma = (d_{\ell}^{n_{\ell}}, \cdots, d_{1}^{n_{1}})$ satisfying the first necessary condition (NC1), the sequence $\sigma^{\gamma} = (d_{\ell}^{\lceil \gamma n_{\ell} \rceil}, \dots, d_{1}^{\lceil \gamma n_{1} \rceil})$ satisfies the sufficient condition (SC).

3.4 MINNDEG realization of tri-sequences

We here consider the scenario when a sequence has only three distinct degrees. Specifically, we provide a complete characterization of sequences $\sigma = (d_3^{n_3} d_2^{n_2} d_1^{n_1})$.

Theorem 3. The necessary and sufficient conditions for MINNDEG-realizability of the sequence $\sigma =$ $(d_{\ell}^{n_{\ell}},\cdots,d_{1}^{n_{1}})$ when $\ell=3$ is

- 1. $d_1 + 1 \le n_1$,
- 2. $d_2 + 1 \le n_1 + n_2$.
- 3. $d_3 \leq \left| \frac{n_1 d_1}{d_1 + 1} \right| + \left| \frac{n_2 d_2}{d_2 + 1} \right| + \left| \frac{n_3 d_3}{d_2 + 1} \right|$, and

4. either
$$d_2 \leq \lfloor \frac{n_1 d_1}{d_1 + 1} \rfloor + \lfloor \frac{n_2 d_2}{d_2 + 1} \rfloor$$
, or $d_3 + 1 \leq n_1 + n_2 + n_3 - \left(1 + \left\lceil \frac{d_2 - n_2}{d_1} \right\rceil\right)$.

Proof. Suppose $\sigma=(d_3^{n_3}d_2^{n_2}d_1^{n_1})$ is realizable, then by Theorem 2, it follows that the first three conditions stated above are necessary.

To prove that all four conditions are necessary, we are left to show that if $d_2 \ge \left\lfloor \frac{n_1 d_1}{d_1 + 1} \right\rfloor + \left\lfloor \frac{n_2 d_2}{d_2 + 1} \right\rfloor$, then $d_3+1 \leq n_1+n_2+n_3-\left(1+\left\lceil\frac{d_2-n_2}{d_1}\right\rceil\right)$. We consider a graph G that realizes σ . Let V_1,V_2,V_3 be the partition of V(G) as defined in Section 3. Consider a vertex $w\in V_2$. Observe that leader(w) must lie in V_1 , because if $L_2\cap V_2$ is non-empty, then Lemma 4 implies $d_2\leq \left\lfloor\frac{n_1d_1}{d_1+1}\right\rfloor+\left\lfloor\frac{n_2d_2}{d_2+1}\right\rfloor$. We first show that $|L_1| \ge \left\lceil \frac{d_2 - n_2}{d_1} \right\rceil$. The set $N(w) \cap V_1$ has size at least $d_2 - n_2$. Each vertex $x \in L_1$ can serve as a leader of at most d_1 vertices in open-neighborhood of w. Indeed, if $x \in N(w)$ then it can not count w (lying outside N(w)), and if $x \notin N(w)$ then it can not count itself (again lying outside N(w)). Thus to cover the set $N(w) \cap V_1$ at least $\left\lceil \frac{d_2 - n_2}{d_1} \right\rceil$ leaders are required, thereby, showing $|L_1| \ge \left\lceil \frac{d_2 - n_2}{d_1} \right\rceil$. Now consider a vertex $y \in V_3$, note that N[y] excludes w (as degree of w is d_2), as well as L_1 (as vertices in L_1 have degree d_1). Therefore, we obtain the following relation.

$$d_3 + 1 = |N[y]| \le |V_1 \setminus L_1| + |V_2 \setminus w| + |V_3| \le n_1 + n_2 + n_3 - \left(1 + \left\lceil \frac{d_2 - n_2}{d_1} \right\rceil\right)$$

We now prove the sufficiency claims. If $d_2 \leq \left\lfloor \frac{n_1d_1}{d_1+1} \right\rfloor + \left\lfloor \frac{n_2d_2}{d_2+1} \right\rfloor$, then the conditions 1-4 are sufficient by Theorem 1. So let us focus on the scenario when $d_2 \geq \left\lfloor \frac{n_1d_1}{d_1+1} \right\rfloor + \left\lfloor \frac{n_2d_2}{d_2+1} \right\rfloor$. Let $N = n_1 + n_2 + n_3 - \left(1 + \left\lceil \frac{d_2-n_2}{d_1} \right\rceil\right)$. The vertex-set of our realized graph G = (V, E) will be a union of three disjoint sets $L_1, L_2 = \{w\}$, and Z of size respectively $\left\lceil \frac{d_2-n_2}{d_1} \right\rceil$, 1, and N. Initially, the edge-set E is an empty-set. Between vertex pairs in Z, we add edges so that the induced graph G[Z] is identical to $\operatorname{GRAPH}(N,d_3,\left\lceil \frac{N}{d_3+1}\right\rceil,\left\lfloor \frac{Nd_3}{d_3+1}\right\rfloor)$. This step is possible since $d_3+1\leq N$, and ensures that $\operatorname{MINNDEG}_{G[Z]}(z)=d_3$, for $z\in Z$. Let L_3 denote the set of those vertices in Z whose degree is equal to d_3 . We connect w to arbitrary $N-n_3=n_2+(n_1-|L_1\cup L_2|)$ vertices in $Z\setminus L_3$, and any arbitrary $\alpha:=d_2-(n_1+n_2-|L_1\cup L_2|)$ vertices in L_1 . Since $\deg_G(w)=d_2$, this step ensures that $\operatorname{MINNDEG}$ of exactly n_2 vertices in Z decreases to d_2 . Let Z be a subset of arbitrary Z and each Z be a subset of arbitrary Z and each Z be a representation of Z be a subset of arbitrary Z. In each Z be a subset of arbitrary Z be a subset of arbitrary Z, and each Z be a representation of Z. Finally, we connect each Z be a subset of arbitrary Z be a subset of

Looking at the complexity of the above characterization, we leave it as an open question to solve the problem in general.

Open Question. Does there exist a polynomial-time algorithm, or a closed-form characterization, for realizing MINNDEG profiles for general graphs?

3.5 Complete characterization for sequences admitting disjoint leader-follower sets

We conclude by providing a complete characterization for special class of MINNDEG-sequences that admit a disjoint leader-follower sets.

Lemma 4. Let G be a graph and $\sigma(G) = (n_{\ell}^{d_{\ell}} \dots d_{1}^{n_{1}})$. For any leader function defined over G and for any $i \in [1, \ell]$, if $L_{i} \cap V_{i}$ is non-empty then $d_{i} \leq \sum_{j=1}^{i} \left| \frac{n_{j}d_{j}}{d_{j}+1} \right|$.

Proof. Let w be any vertex lying in $L_i \cap V_i$, so $min\text{-}deg(w) = deg(w) = d_i$. Recall for each j < i, vertices in the set L_j have degree strictly less than d_i . Since N[w] cannot contain vertices of degree less than d_i , thus for each j < i, $N[w] \cap L_j = \emptyset$. Also vertices in $V_{i+1} \cup \ldots \cup V_\ell$ cannot be adjacent to any vertex in $\{w\} \cup \left(\bigcup_{j=1}^{i-1} L_j\right)$, therefore, N[w] as well as $\bigcup_{j=1}^{i-1} L_j$ are contained in union $\bigcup_{j=1}^{i} V_j$. We thus get,

$$d_i + 1 = |N[w]| \le \Big|\bigcup_{j=1}^i V_j\Big| - \Big|\bigcup_{j=1}^{i-1} L_j\Big| = n_i + \sum_{j=1}^{i-1} (n_i - |L_j|) \le n_i + \sum_{j=1}^{i-1} \Big\lfloor \frac{n_j d_j}{d_j + 1} \Big\rfloor,$$

where the last inequality follows from Lemma 3. If $n_i \leq d_i$, then $n_i - 1 = n_i - \left \lceil \frac{n_i d_i}{d_i + 1} \right \rceil = \left \lfloor \frac{n_i d_i}{d_i + 1} \right \rfloor$, and so $d_i \leq \sum_{j=1}^i \left \lfloor \frac{n_j d_j}{d_j + 1} \right \rfloor$. If $n_i \geq d_i + 1$, then the bound trivially holds since $\frac{n_i d_i}{d_i + 1} \geq d_i$ which from the fact that d_i is integral implies $d_i \leq \left \lfloor \frac{n_i d_i}{d_i + 1} \right \rfloor$.

Theorem 4. A sequence $\sigma = (n_{\ell}^{d_{\ell}} \dots d_1^{n_1})$ is MINNDEG-realizable by a graph G having disjoint leader-set (L) and follower-set (F) with respect to some leader function, if and only if, for each $i \in [1, \ell]$, $d_i \leq \sum_{j=1}^{i} \left\lfloor \frac{n_j d_j}{d_j + 1} \right\rfloor$.

Proof. Let us suppose there exists a leader function over G for which $L \cap F = \emptyset$, then for each $i \in [1, \ell]$, $L_i \subseteq V_i$. This is because if for some i, there exists $w \in L_i \setminus V_i$, then $deg(w) = d_i \neq min\text{-}deg(d_i)$, which implies that w is a leader as well as a follower. Since $L_i \subseteq V_i$, by Lemma 4, $d_i \leq \sum_{j=1}^i \left\lfloor \frac{n_j d_j}{d_j + 1} \right\rfloor$, for each $i \in [1, \ell]$. The converse claim follows from Theorem 1.

4 Realizing maximum neighborhood degree profiles

In this section, we provide a complete characterization of MAXNDEG profiles. For simplicity, we first discuss the uniform scenario of $\sigma=(d^k)$. Observe that a star graph $K_{1,d}$ is MAXNDEG realization of the profile (d^{d+1}) . We show in the following lemma that, by identifying together vertices in different copies of $K_{1,d}$, it is always possible to realize the profile (d^k) , whenever $k \ge d+1$.

Lemma 5. For any positive integers d and k, the profile $\sigma = (d^k)$ is MAXNDEG realizable whenever $k \geq d+1$. Moreover, we can always compute in O(k) time a connected realization that has an independent set, say S, of size d such that all vertices in S have degree at most d, and at least two vertices in d have degree d.

Proof. Let α be the smallest integer such that $k \leq 2 + \alpha(d-1)$. We first construct a caterpillar α as follows. Take a path $P = (s_0, s_1, \dots, s_\alpha, s_{\alpha+1})$ of length $\alpha + 1$. Connect each internal vertex s_i (here $i \in [1, \alpha]$) with a set of d-2 new vertices, so that the degree of s_i is d. (See Figure 3). Note that the MAXNDEG of each vertex $v \in T$ is d.

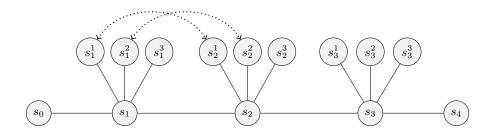


Figure 3: A caterpillar for d=5 and $\alpha=3$. If k=12, then r=2, and we merge (i) s_1^1 and s_2^1 , and (ii) s_1^2 and s_2^2 .

Now if $k=2+\alpha(d-1)$, then T serves as our required realizing graph. If $k<2+\alpha(d-1)$, then $\alpha\geq 2$ since $k\geq d+1$. The tree T is "almost" a realizing graph for the profile, except that it has too many vertices. Let $r=2+\alpha(d-1)-k$ denote the number of excess vertices in T that need to be *removed*. The r vertices can be removed as follows. Take any two distinct internal vertices s_i and s_j on P, and let s_i^1,\ldots,s_i^{d-2} and s_j^1,\ldots,s_j^{d-2} , respectively, denote the neighbors of s_i and s_j not lying on P. Let G be the graph obtained by merging vertices s_i^ℓ and s_j^ℓ into a single vertex for $\ell\in[1,r]$. (See Figure 3). Since the number of vertices was decreased by r, G now contains exactly n vertices. The degree of vertices $s_1,s_2,\ldots,s_{\alpha}$ remains d, and the degree of all other vertices is at most 2, therefore MAXNDEG(v)=d for each $v\in G$, so G is a realization of the profile σ .

Finally, in the resultant graph G, the end points of P (i.e. s_0 and $s_{\alpha+1}$) have degree 1, and there are d-2 other vertices, namely s_i^1,\ldots,s_i^{d-2} (or s_j^1,\ldots,s_j^{d-2}), that have degree bounded by 2. Therefore we set S to these d vertices. It is easy to verify that S is indeed an independent set. \square

²A caterpillar is a tree in which all the vertices are within distance one of a central path.

4.1 An incremental procedure for computing MAXNDEG realizations

We explain here our main building block, procedure ADDLAYER, that will be useful in incrementally building graph realizations in a decreasing order of maximum degrees. Given a partially computed connected graph H and integers d and k satisfying $d \geq 2$ and $k \geq 1$, the procedure adds to H a set W of k new vertices such that MAXNDEG(w) = d, for each $w \in W$. The reader may assume that $\text{MAXNDEG}(v) \geq d$, for each existing vertex $v \in V(H)$. The procedure takes in as an input a sufficiently large vertex list L (of size d-1) that forms an independent set in H, and whose vertices have small degree (that is, at most d-1). Moreover, in order to accommodate its iterative use, each invocation of the procedure also generates and outputs a new list, to be used in the further iterations.

Procedure ADDLAYER The input to procedure ADDLAYER (H, L, k, d) is a connected graph H and a list $L = (a_1, \ldots, a_{d-1})$ of vertices in H whose degree is bounded above by d-1. The first step is to add to H a set of k new vertices $W = \{w_1, w_2, \ldots, w_k\}$. Next, the new vertices are connected to the vertices of L and to themselves so as to ensure that MAXNDEG(w) = d for every $w \in W$. Depending upon whether or not k < d, there are two separate cases. (Refer to Algorithm 2 for pseudocode).

Let us first consider the case $k \leq d-1$. In this case we add edges from vertices in W to a subset of vertices from L such that those vertices in L will have degree d and therefore will imply MAXNDEG(w) = d, for every $w \in W$. We initialize two variables, count and i, respectively, to k and d-1. The variable count holds, at any instant of time, the number of vertices in W that still need to be connected to vertices in L. While count > 0, the procedure performs the following steps: (i) compute $r = \min\{d - deg(a_i), count\}$, the maximum number of vertices in W that can be connected to vertex a_i ; (ii) connect a_i to following r vertices in W: $w_{count-(r-1)}, w_{count-(r-2)}, \ldots, w_{count-1}, w_{count}$; and (iii) decrease count by r, and i by 1.

When count = 0, the vertices $a_i, a_{i+1}, \ldots, a_{d-1}$ are connected to at least one vertex in W (this implies $d-i \leq k$). It is also easy to verify that at this stage, $deg(a_{d-1}) = deg(a_{d-2}) = \cdots = deg(a_{i+1}) = d$, and $deg(a_i) \leq d$. Since the input graph H was connected, in the beginning of the execution $deg(a_i) \geq 1$, and by connecting a_i to at least one vertex in W, specifically to w_1 , its degree is increased at least by one. So at most d-2 edges need to be added to a_i to ensure that its degree is exactly d. The procedure performs the following operation for each $j \in [d-1, d-2, \ldots, 2, 1]$ (in the given order) until $deg(a_i) = d$: (i) if j < i then add edge (a_j, a_i) to H, and (ii) if j > i then add an edge between a_i and an arbitrary neighbor of a_j lying in W. Since $deg(a_i) = deg(a_{i+1}) = \cdots = deg(a_{d-1}) = d$, and $deg(w) \leq 2$ for every $w \in W$, it follows that MAXNDEG(w) = d, for each $w \in W$. In the end, we set a new list L containing the first d-2 vertices in the sequence $(w_1, w_2, \ldots, w_k, a_1, a_2, \ldots, a_{i-1})$. This is possible since $k+i-1 \geq d-2$ due to the fact that $d-i \leq k$. (Later on we bound the degrees of the vertices in the new list.)

Now we consider the case $k \geq d$. The procedure uses Lemma 5 to compute over the independent set $W \cup \{a_1\}$ a graph \bar{H} realizing the profile (d^{k+1}) such that $deg_{\bar{H}}(a_1) = 1$. Notice that in the beginning of the execution, $deg(a_1) \in [1, d-1]$, and it is increased by one by adding \bar{H} over the set $W \cup \{a_1\}$. So now $deg(a_1) \in [2, d]$. To ensure $deg(a_1) = d$, at most d-2 more edges need to be added to a_1 . Edges are added between a_1 and any arbitrary $d - deg(a_1)$ vertices in set $\{a_2, a_3, \dots, a_{d-1}\}$. This ensures that every $w \in W$ has MAXNDEG(w) = d. By Lemma 5, $\bar{H} \setminus \{a_1\}$ contains an independent set of d-1 vertices, say b_1, \dots, b_{d-1} , such that $1 = deg_{\bar{H}}(b_1) \leq deg_{\bar{H}}(b_2) \leq \dots \leq deg_{\bar{H}}(b_{d-1}) \leq 2$. In the end, the procedure creates a new list $L = (b_1, b_2, \dots, b_{d-2})$.

For sake of better understanding, in the rest of paper, we denote by H_{old}, L_{old} and H_{new}, L_{new} respectively the graph and the list before and after the execution of Procedure ADDLAYER. Observe that $V(H_{new}) = V(H_{old}) \cup W$.

The following two lemmas follow from the description of algorithm.

Lemma 6. Each $w \in W$ satisfies MAXNDEG(w) = d, and $N(w) \subseteq W \cup L_{old}$.

```
1 Let the list L be (a_1, a_2, \ldots, a_{d-1}).
 2 Add to H a set W = \{w_1, \dots, w_k\} of k new vertices.
 3 case (k < d) do
        Set count = k and i = d - 1.
        while (count \neq 0) do
            Let r = \min\{d - deg(a_i), count\}.
            Add edges (a_i, w_{count-t}) to H for t \in [0, r-1].
            Decrement i by 1 and count by r.
8
        foreach j \in [d-1, ..., 2, 1] do
            If deg(a_i) = d then break the for loop.
10
            If (j < i) then add edge (a_i, a_i) to H.
11
           If (j > i) then add an edge between a_i and an arbitrary vertex in N(a_j) \cap W.
12
       Set L to be prefix of (w_1, w_2, \dots, w_k, a_1, a_2, \dots, a_{i-1}) of size d-2.
14 case (k \geq d) do
       Use Lemma 5 to compute over independent set (W \cup \{a_1\}) the graph, say \bar{H}, realizing the
         profile (d^{k+1}) such that deg_{\bar{H}}(a_1) = 1.
        Add edges between a_1 and any arbitrary d - deg(a_1) vertices in set \{a_2, a_3, \dots, a_{d-1}\}.
       Let b_1, \ldots, b_{d-1} \in \overline{H} \setminus a_1 be such that 1 = \deg_{\overline{H}}(b_1) \leq \cdots \leq \deg_{\overline{H}}(b_{d-1}) \leq 2.
17
       Set L = (b_1, b_2, \dots, b_{d-2}).
19 Output L.
```

Algorithm 2: ADDLAYER (H, L, k, d)

```
Input: A sequence \sigma = (d_\ell^{n_\ell}, \cdots, d_1^{n_1}) satisfying d_\ell \leq n_\ell - 1 and d_1 \geq 2.

1 Initialize G to be the graph obtained from Lemma 5 that realizes the profile (d_\ell^{n_\ell}).

2 Let L_{\ell-1} be a valid list in G of size d_{\ell-1} - 1.

3 for (i = \ell - 1 \text{ to } 1) do

4 L_{i-1} \leftarrow \text{AddLayer}(G, L_i, n_i, d_i).

5 Truncate list L_{i-1} to contain only the first d_{i-1} - 1 \leq d_i - 2 vertices.

6 Output G.
```

Algorithm 3: MAXNDEG realization of $\sigma = (d_{\ell}^{n_{\ell}}, \dots, d_{1}^{n_{1}})$

Lemma 7. Each $a \in L_{old} \setminus L_{new}$ satisfies $deg_{H_{new}}(a) \le d$, and each $a \in L_{old} \cap L_{new}$ satisfies $deg_{H_{new}}(a) \le deg_{H_{old}}(a) + 1$.

It is also easy to verify that the total execution time of Procedure ADDLAYER is O(k+d).

The Inheritance Property Till now, we showed that given an independent list of d-1 vertices of degree at most d-1 in a graph H, we can add $k \geq 1$ vertices to H such that the MAXNDEG of these k vertices is d. In order to iteratively use this algorithm to add vertices of smaller MAXNDEG values ($\leq d$) we require that the list L_{new} computed by Procedure ADDLAYER should satisfy following three constraints: (i) The size of L_{new} should be d-2; (ii) the vertices of L_{new} should form an independent set; and most importantly, (iii) the vertices in L_{new} should have degree at most d-2.

In order to ensure these constraints on L_{new} , we further impose the constraint that the list L_{old} is a valid

list; this is formally defined as below.

Definition 2 (Valid List). A list $L = (a_1, a_2, ..., a_t)$ in a graph G is said to be "valid" with respect to G if the following two conditions hold: (i) for each $i \in [1, t]$, $deg(a_i) \leq i$, and (ii) the vertices of L form an independent set in G.

We next prove the inheritance property of our procedure.

Lemma 8 (Inheritance property). If the input list L_{old} in Procedure ADDLAYER is valid, then the output list L_{new} is valid as well.

Proof. We first consider the case $k \leq d-1$. Let i be the smallest index such that vertices $a_i, a_{i+1}, \ldots, a_{d-1}$ are adjacent to some vertex of W in H_{new} . (That is, i is the index when Procedure ADDLAYER exits the while loop). Recall that in the graph H_{new} , $w_1 \in W$ is a neighbor of a_i . Also, to increase the degree of a_i to d, we connect a_i to some/all vertices in a_1, \ldots, a_{i-1} , and some/all neighbors of a_{i+1}, \ldots, a_{d-1} lying in W. Therefore the vertex set $W \cup \{a_1, \ldots, a_{i-1}\}$ is independent in H_{new} . Also, its size at least d-1, as we showed that $k \geq d-i$. Since the list $L_{old} = (a_1, a_2, \ldots, a_{d-1})$ is valid in the beginning of the execution of Procedure ADDLAYER, it follows that in H_{old} , $deg(a_j) \leq j$ for $j \in [1, d-1]$. So by Lemma 7, in H_{new} , (i) $deg(a_j) \leq j+1$ for $j \in [1,i-1]$, (ii) $deg(w_1) = 1$, and (iii) the degree of each other vertex in $W \setminus w_1$ is at most 2. Consequently, (w_1, \cdots, w_k) is a valid list of length at least d-i. Since $deg(a_j) \leq j+1$ for $i \in [1,i-1]$, the list $i \in [1,i-1]$ is valid and has length at least $i \in [1,i-1]$. Truncating this to length $i \in [1,i-1]$ again gives us a valid list.

We now consider the case $k \geq d$. By Lemma 5, $H[W \cup \{a_1\}] = H$ contains an independent set $\{b_1, b_2, \ldots, b_{d-1}\} \subseteq W$ such that $deg(b_1) = 1$ and $deg(b_j) \leq 2$ for $j \in [2, d-1]$. Therefore, $(b_1, b_2, \ldots, b_{d-2})$ is a valid list of length d-2 in H_{new} .

The following proposition summarizes the above discussion.

Proposition 1. For any integers $d \geq 2$, $k \geq 1$, and any connected graph H containing a valid list L of size d-1, procedure ADDLAYER adds to H in O(k+d) time, a set W of k new vertices such that MAXNDEG(w) = d, for every $w \in W$. All the edges added to H lie in $W \times (W \cup L)$. Moreover, $deg_H(a) \leq d$, for every $a \in L$, and the updated graph remains connected and contains a new valid list of size d-2.

4.2 The main algorithm

We now present the main algorithm for computing the realizing graph using Procedure ADDLAYER.

Let $\sigma=(d_\ell^{n_\ell},\cdots,d_1^{n_1})$ be any profile satisfying $d_\ell \leq n_\ell-1$ and $d_1 \geq 2$. The construction of a connected graph realizing σ is as follows (refer to Algorithm 3 for pseudocode). We first use Lemma 5 to initialize G to be the graph realizing the profile $(d_\ell^{n_\ell})$. Recall G contains an independent set, say $W=\{w_1,w_2,\ldots,w_{d_\ell}\}$, satisfying the condition that the degree of the first two vertices is one, and the degree of the remaining vertices is at most two. Set $L_{\ell-1}=(w_1,w_2,\ldots,w_{d_{\ell-1}-1})$ (notice that $d_{\ell-1}-1\leq d_\ell$). It is easy to verify that this list is valid. Next, for each $i=\ell-1$ to 1, perform the following steps:

- (i) Taking as input the valid list L_i of size $d_i 1$, execute Procedure ADDLAYER (G, L_i, n_i, d_i) to add n_i new vertices to G. The procedure returns a valid list L_{i-1} of size $d_i 2$.
- (ii) Truncate the list L_{i-1} to contain only the first $d_{i-1} 1 \le d_i 2$ vertices. The truncated list remains valid since any prefix of a valid list is valid.

Proof of Correctness Let V_{ℓ} denote the set of vertices in graph G initialized in step 1, and for $i \in [1, \ell - 1]$, let V_i denote the set of n_i new vertices added to graph G in iteration i of the for loop. Also for $i \in [1, \ell]$, let G_i be the graph induced by vertices $V_i \cup \cdots \cup V_{\ell}$. The following lemma proves the correctness.

Lemma 9. For any $i \in [1, \ell]$, graph G_i is a MAXNDEG realization of profile $(d_{\ell}^{n_{\ell}}, \dots, d_{i}^{n_{i}})$, and for any $j \in [i, \ell]$ and any $v \in V_j$, $deg_{G_i}(v) \leq MAXNDEG_{G_i}(v) = d_j$.

Proof. We prove the claim by induction on the iterations of the for loop. The base case is for index ℓ , and by Lemma 5 we have that $deg_{G_{\ell}}(v) \leq \text{MAXNDEG}_{G_{\ell}}(v) = d_{\ell}$, for every $v \in V_{\ell}$. For the inductive step, we assume that the claim holds for i+1, and prove the claim for i. Consider any vertex v in G_i . We have two cases.

- 1. $v \in V_i$: In this case by Proposition 1 we have that $deg_{G_i}(v) \leq MAXNDEG_{G_i}(v) = d_i$.
- 2. $v \in V_j$, for j > i: We first show that for any vertex $w \in N_{G_i}[v]$, $deg_{G_i}(w) \leq d_j$. If $w \in V_i$, then we already showed $deg_{G_i}(w) \leq d_i$. So let us consider the case $w \in V_{i+1} \cup \cdots \cup V_\ell$. Now if $w \in L_i$ participates in Procedure ADDLAYER (G, L_i, n_i, d_i) , then by Proposition 1, in the updated graph $deg_{G_i}(w) \leq d_i \leq d_j$. If $w \notin L_i$, then the degree of w is unaltered in the i^{th} iteration, and thus $deg_{G_i}(w) = deg_{G_{i+1}}(w) \leq \text{MAXNDEG}_{G_{i+1}}(v) = d_j$ by the inductive hypothesis. It follows that MAXNDEG(v) remains unaltered due to iteration i, and thus $\text{MAXNDEG}_{G_i}(v) = \text{MAXNDEG}_{G_{i+1}}(v) = d_j$.

The execution time of the algorithm is $O\left(\sum_{i=1}^{\ell}(n_i+d_i)\right)$. This is also optimal. Indeed, any connected graph realizing σ must contain $\Omega(n_1+n_2+\cdots+n_\ell)$ edges as the degrees of all vertices must be non-zero. Also, the graph must contain at least one vertex of each of the degrees d_1,d_2,\ldots,d_ℓ , and therefore must have $\Omega(d_1+d_2+\cdots+d_\ell)$ edges. In other words, any realizing graph must contain $\Omega\left(\sum_{i=1}^{\ell}(n_i+d_i)\right)$ edges, and thus the computation time must be at least $\Omega\left(\sum_{i=1}^{\ell}(n_i+d_i)\right)$. The following theorem is immediate from the above discussions.

Theorem 5. There exists an algorithm that given any profile $\sigma = (d_{\ell}^{n_{\ell}}, \dots, d_{1}^{n_{1}})$ satisfying $d_{\ell} \leq n_{\ell} - 1$ and $d_{1} \geq 2$ computes in optimal time a connected MAXNDEG realization of σ .

4.3 A complete characterization for MAXNDEG realizable profiles

The necessary conditions for MAXNDEG realizability is as follows.

Lemma 10. A necessary condition for a profile $\sigma = (d_{\ell}^{n_{\ell}}, \cdots, d_{1}^{n_{1}})$ to be MAXNDEG realizable is $d_{\ell} \leq n_{\ell} - 1$.

Proof. Suppose σ is MAXNDEG realizable by a graph G. Then G must contain a vertex, say w, of degree d_{ℓ} in G. Since d_{ℓ} is the maximum degree in G, the MAXNDEG of all the $d_{\ell}+1$ vertices in N[w] must be d_{ℓ} . Thus $n_{\ell} \geq d_{\ell}+1$.

Consider a profile $\sigma=(d_\ell^{n_\ell},\cdots,d_1^{n_1})$ realizable by a connected graph. If $d_1=1$, then the graph must contain a vertex, say v, of degree 1, and the vertices in N[v] must also have degree 1. The only possibility for such a graph is a single edge graph on two vertices. Thus in this case $\sigma=(1^2)$. If $d_1\geq 2$, then by Lemma 10, for σ to be realizable in this case we need that $n_\ell\geq d_\ell+1$. Also, by Theorem 5, under these two conditions σ is always realizable. We thus have the following theorem.

Theorem 6. For a profile $\sigma=(d_{\ell}^{n_{\ell}},\cdots,d_{1}^{n_{1}})$ to be MAXNDEG realizable by a connected graph the necessary and sufficient condition is that either (i) $n_{\ell} \geq d_{\ell} + 1$ and $d_{1} \geq 2$, or (ii) $\sigma=(1^{2})$.

Now if $d_1 = 1$, then n_1 must be even, since the vertices v with MAXNDEG(v) = 1 must form a disjoint union of exactly $n_1/2$ edges. So for general graphs we have the following theorem.

Theorem 7. For a profile $\sigma = (d_{\ell}^{n_{\ell}}, \cdots, d_{1}^{n_{1}})$ to be MAXNDEG realizable by a general graph the necessary and sufficient conditions are that $d_{\ell} \geq n_{\ell} - 1$, and either n_{1} is even or $d_{1} \geq 2$.

5 Realizing maximum open neighborhood-degree profiles

We start by formally defining the realizable profiles for maximum degree in open neighborhood.

Definition 3 (MAXNDEG⁻ realizable profile). A profile $\sigma = (d_{\ell}^{n_{\ell}}, \dots, d_{1}^{n_{1}})$ is said to be MAXNDEG⁻ realizable if there exists a graph G on $n = n_{1} + \dots + n_{\ell}$ vertices that for each $i \in [1, \ell]$ contains exactly n_{i} vertices whose MAXNDEG⁻ is d_{i} . Equivalently, $|\{v \in V(G) : \text{MAXNDEG}^{-}(v) = d_{i}\}| = n_{i}^{3}$.

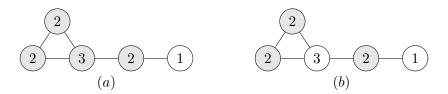


Figure 4: A comparison of the MAXNDEG realization of $(3^4, 2^1)$ and a MAXNDEG⁻ realization of $(3^3, 2^2)$.

Observe that in the case of MAXNDEG⁻ profiles, unfortunately, the nice *sub-structure property* (see Section 6) does not always hold. For example, for the graph considered in Figure 4, the profile $\sigma = (3^3, 2^2)$ is MAXNDEG⁻ realizable, however, the subsequence (3^3) is not MAXNDEG⁻ realizable.

5.1 Pseudo-valid List

We begin by stating the following lemmas that are an extension of Lemma 5 and Proposition 1 presented in Section 4 for MAXNDEG profiles.

Lemma 11. For any positive integers d and k, the profile $\sigma = (d^k)$ is MAXNDEG⁻ realizable whenever $k \geq d+2$. Moreover, we can always compute in O(k) time a connected realization that contains an independent set having (i) two vertices of degree 1, and (ii) d-2 other vertices of degree at most 2.

Proposition 2. For any integers $d \geq 2$, $k \geq 1$, and any connected graph H containing a valid list L of size d-1, procedure ADDLAYER adds to H in O(k+d) time, a set W of k new vertices such that $MAXNDEG^-(w) = d$, for every $w \in W$. All the edges added to H lie in $W \times (W \cup L)$. Moreover, $deg_H(a) \leq d$, for every $a \in L$, and the updated graph remains connected and contains a new valid list of size d-2.

It is important to note that though the Proposition 2 holds for the open-neighborhoods it can not be directly used to incrementally compute the realizations. This is due to the reason that for the profiles $\sigma = (d_\ell^{d_\ell+1})$ unlike the scenario of MAXNDEG realization, there is no MAXNDEG⁻ realization that contains a valid list (See Lemma 13 for further details).

This motivates us to define pseudo-valid lists.

³For a vertex v in H, the maximum degree in the open neighborhood $(N_H[v] \setminus v)$ of vertex v, namely MAXNDEG $^-_H(v)$ is defined to be the maximum over the degrees of all the vertices present in the open neighborhood of v.

Definition 4. A list $L = (a_1, a_2, \dots, a_t)$ in a graph H is said to be "pseudo-valid" with respect to H if (i) for each $i \in [1, t]$, $deg(a_i) = 2$, and (ii) the vertices of L form an independent set.

Note that the only deviation that prevents L from being a valid list is that $deg(a_1)$ is 2 instead of 1.

We next state two lemmas that are crucial in obtaining MAXNDEG⁻ realizations in the scenarios $n_{\ell} = d_{\ell}$ and $n_{\ell} = d_{\ell} + 1$.

Lemma 12. For any integers $d > \bar{d} \ge 2$, the profile $\sigma = (d^d, \bar{d}^1)$ is MAXNDEG⁻ realizable. Moreover, in O(d) time we can compute a connected realization that contains a valid list of size d-1.

Proof. The construction of G is as follows. Take a vertex z and connect it to d-1 other vertices v_1,\ldots,v_{d-1} . Next take another vertex y and connect to $v_1,\ldots,v_{\bar{d}-1}$ (recall $2\leq \bar{d}< d$). Also connect z to y. In the resulting graph G, deg(z)=d, $deg(y)=\bar{d}$, and $deg(v_i)\leq 2$ for $i\in [1,d-1]$. Also, v_{d-1} is not adjacent to y as $\bar{d}< d$, thus $deg(v_{d-1})=1$. Therefore, MAXNDEG $^-(z)=\bar{d}$, MAXNDEG $^-(y)=d$, and MAXNDEG $^-(v_i)=d$, for $i\in [1,d-1]$. It is also easy to verify that $(v_{d-1},\ldots,v_{\bar{d}-1},\ldots,v_2,v_1)$ is a valid list in G.

Lemma 13. For any integer $d \geq 2$, the profile $\sigma = (d^{d+1})$ is MAXNDEG⁻ realizable. Moreover, a connected realization that contains an independent set having d-1 vertices of degree 2 can be compute in O(d) time. However, none of the graphs realizing σ can contain a vertex of degree 1.

Proof. The construction of graph G realizing σ is very similar to the previous lemma. Take two vertex-sets, namely, $U = \{u_1, u_2\}$ and $W = \{w_1, \dots, w_{d-1}\}$. Add to G the edge (u_1, u_2) , and for each $i \in [1, d-1]$, add to G the edges (u_1, w_i) and (u_2, w_i) . This ensures that $deg(u_1) = deg(u_2) = d$ and $deg(w_i) = 2$ for $i \in [1, d-1]$. So G contains d+1 vertices with MAXNDEG⁻ equal to d. Also, W is an independent set of size d-1 in G and $deg(w_i) = 2$, for every vertex $w_i \in W$.

Next, let H be any MAXNDEG $^-$ realizing graph of σ . Then H must contain two vertices, say x and y, of degree d, since a single vertex of degree d in H can guarantee MAXNDEG $^-$ = d for at most d vertices. Next notice that N[x] = N[y], because otherwise H will contain more than d+1 vertices. This implies that all the vertices in H, other than x and y, are adjacent to both x and y. Therefore, each of the vertices in H must have degree at least two.

The next lemma shows that ADDLAYER outputs a valid list, even when the input list is pseudo-valid.

Lemma 14. In procedure ADDLAYER, the list L_{new} is valid even when the list L_{old} is pseudo-valid and the parameter d satisfies $d \ge 3$.

Proof. We borrow notations from the proof of Lemma 8. As before, we have two separate cases depending on whether or not k < d. We first consider the case $k \leq d-1$. We showed in Lemma 8 that $(w_1, \cdots, w_k, a_1, \ldots, a_{i-1})$ is a valid list of length at least d-1 when $deg_{H_{old}}(a_1)=1$. We now consider the scenario when L_{old} is pseudo-valid, and $deg_{H_{old}}(a_1)=2$. The list L_{new} is still valid if $k \geq 2$, since the degree of a_1 in H_{new} is at most 3 and its position in L_{new} is also 3 or greater. So the non-trivial case is k=1. In such a case i=d-1, as the only vertex w_1 belonging to W is connected to a_{d-1} in Algorithm 2. Also, $deg_{H_{old}}(a_{d-1})=2$, and a_{d-1} is connected to vertex w_1 , so to ensure that $deg(a_{d-1})=d$, in the for loop in step 9 of Algorithm 2, it is connected to only d-3 vertices, namely, $a_2, a_3, \ldots, a_{d-2}$. Since a_{d-1} is never connected to vertex $a_1, deg_{H_{new}}(a_1)=deg_{H_{old}}(a_1)=2$. This shows that the sequence $(w_1, \cdots, w_k, a_1, \ldots, a_{i-1})=(w_1, a_1, \ldots, a_{d-2})$ is a valid list of length exactly d-1. Truncating it to length d-2 again yields a valid sequence. In case $k \geq d$, a_1 's degree does not play any role, so the argument from the proof of Lemma 8 works as is.

Remark 4. The condition $d \ge 3$ is necessary in Lemma 14 because in a pseudo-valid list all the vertices have degree 2. However, Procedure ADDLAYER works only in the case when the degree of each vertex in the list is at most d-1, which does not hold true for a pseudo-valid list when d=2. So we provide a different analysis for the profile $(d^{d+1}, 2^k)$.

5.2 MAXNDEG⁻ realization of the profile $\sigma = (d^{d+1}, 2^k)$

The following lemmas shows that $\sigma=(d^{d+1},2^1)$, for $d\geq 3$, is not MAXNDEG⁻ realizable when $d\geq 3$; and $\sigma=(d^{d+1},2^k)$ is MAXNDEG⁻ realizable when $d\geq 3$ and $k\geq 2$.

Lemma 15. For any integer $d \geq 3$, the profile $\sigma = (d^{d+1}, 2^1)$ is not MAXNDEG⁻ realizable.

Proof. Let us assume on the contrary that σ is MAXNDEG⁻ realizable by a graph G, and let $w \in V(G)$ be a vertex such that MAXNDEG⁻(w) = 2. The graph G must contain at least two vertices, say x and y, of degree d, since a single vertex of degree d can guarantee MAXNDEG⁻ of d for at most d vertices in the graph. Consider the following two cases.

- (i) N[x] = N[y]: In this case the MAXNDEG⁻ of all the vertices in N[x] = N[y] is at least $d \geq 3$, as they are adjacent to either x or y. Thus $w \notin N[x]$, which implies that $V(G) = N[x] \cup \{w\}$ since |N[x]| = d+1 and |V(G)| = d+2. Also, w cannot be adjacent to any vertex in N[x], because if w is adjacent to a vertex $w_0 \in N[x]$, then $deg(w_0)$ must be 3, in contradiction to the assumption MAXNDEG⁻(w) = 2. Thus the only possibility left is that w is a singleton vertex, which is again a contradiction.
- (ii) $N[x] \neq N[y]$: In this case the vertex set of G is equal to $N[x] \cup N[y]$ since size of $N[x] \cup N[y]$ must be at least d+2 (as $|N[x] \cap N[y]| \leq d$) and is also at most |V(G)| = d+2. This implies that all the vertices of G are adjacent to either x or y, which contradicts the fact that $MAXNDEG^{-}(w) = 2$, since $deg(x) = deg(y) = d \geq 3$.

Lemma 16. For any integers $d \ge 3$ and $k \ge 2$, the profile $\sigma = (d^{d+1}, 2^k)$ is MAXNDEG⁻ realizable. Moreover, we can compute a connected realization in O(d+k) time.

Proof. The construction of G is as follows. Take a vertex u_1 and connect it to d other vertices v_1,\ldots,v_d . Next, take another vertex u_2 and connect it to vertices v_2,\ldots,v_d , and a new vertex v_{d+1} . Finally, take a path $(a_1,a_2,\ldots,a_{\alpha})$ on $\alpha=k-2$ new vertices, and connect a_1 to v_{d+1} . In the graph G, $deg(u_1)=deg(u_2)=d$, and $deg(v_i), deg(a_j) \leq 2$, for $i \in [1,d+1]$ and $j \in [1,k-2]$. Vertices u_1 and u_2 has maximum degree in their neighborhood 2, thus MAXNDEG $^-(u_1)=MAXNDEG^-(u_2)=2$. Each v_i is adjacent to u_1,u_2 , for $i \in [1,d+1]$, so its MAXNDEG $^-$ is d. And, the MAXNDEG $^-$ of vertices on the path $(a_1,a_2,\ldots,a_{\alpha})$ is 2, since they have a neighbour of degree 2.

5.3 Algorithm

We now explain the construction of a graph realizing the profile $\sigma=(d_\ell^{n_\ell},\cdots,d_1^{n_1})\neq (d_\ell^{d_\ell+1},2^1)$ that satisfies the conditions (i) $d_\ell \leq \min\{n_\ell,n-1\}$, and (ii) $d_1 \geq 2$, where $n=n_1+\cdots+n_\ell$. If σ is equal to $(d_\ell^{d_\ell+1},2^k)$, for some $k\geq 2$, we use Lemma 16 to realize σ . If not, then depending upon the value of n_ℓ , we initialize G differently as follows. (Refer to Algorithm 4 for the pseudocode).

1. If $n_\ell \ge d_\ell + 2$, we use Lemma 11 to initialize G to be a MAXNDEG⁻ realization of the profile $(d_\ell^{n_\ell})$. Recall G contains an independent set, say $W = \{w_1, w_2, \ldots, w_{d_\ell}\}$, satisfying the condition that the degree of first two vertices is one, and the degree of the remaining vertices is at most two. We set $L_{\ell-1}$ to be the list $(w_1, w_2, \ldots, w_{d_{\ell-1}-1})$ (notice $d_{\ell-1} - 1 < d_\ell$). It is easy to verify that this list is valid.

```
Input: A sequence \sigma = (d_\ell^{n_\ell}, \cdots, d_1^{n_1}) \neq (d_\ell^{d_\ell+1} 2^1) satisfying d_\ell \leq \min\{n-1, n_\ell\} and d_1 \geq 2.
1 if \sigma = (d_{\ell}^{d_{\ell}+1}, 2^k) for some k \geq 2 then
    Use Lemma 16 to compute a realization G for profile \sigma.
3 else
        case n_{\ell} \geq d_{\ell} + 2 do
              Initialize G to be the graph obtained from Lemma 11 that realizes the profile (d_{\ell}^{n_{\ell}}).
              Set L_{\ell-1} to be a valid list in G of size d_{\ell-1}-1.
        case n_{\ell} = d_{\ell} + 1 do
              Initialize G to be the graph obtained from Lemma 12 that realizes the profile (d_{\ell}^{d_{\ell}+1}).
              Set L_{\ell-1} to be a pseudo-valid list in G of size d_{\ell-1}-1.
        case n_{\ell} = d_{\ell} \operatorname{do}
10
              Initialize G to be the graph obtained from Lemma 13 that realizes the profile (d_{\ell}^{d_{\ell}}d_{\ell-1}).
11
12
              Set L_{\ell-1} to be a valid list in G of size d_{\ell-1}-1.
           Decrement n_{\ell-1} by 1.
13
         for (i = \ell - 1 \text{ to } 1) do
14
             L_{i-1} \leftarrow \text{AddLayer}(G, L_i, n_i, d_i).
15
             Truncate list L_{i-1} to contain only the first d_{i-1} - 1 (\leq d_i - 2) vertices.
16
17 Output G.
```

Algorithm 4: MAXNDEG⁻ realization of $\sigma = (d_{\ell}^{n_{\ell}}, \dots, d_{1}^{n_{1}})$

- 2. If $n_\ell = d_\ell + 1$, then a realization of $(d_\ell^{d_\ell+1})$ does not contains a valid list. So we use Lemma 13 to initialize G to be a MAXNDEG⁻ realization of the profile $(d_\ell^{d_\ell+1})$ that contains a pseudo-valid list. This is possible since we showed G contains an independent set, say $W = \{w_1, w_2, \ldots, w_{d_\ell-1}\}$, such that degree of each $w \in W$ is two. We set $L_{\ell-1}$ to be the list $(w_1, w_2, \ldots, w_{d_{\ell-1}-1})$ (again notice $d_{\ell-1}-1 < d_\ell-1$).
- 3. If $n_{\ell} = d_{\ell}$, then the sequence $d_{\ell}^{d_{\ell}}$ is not realizable (see Lemma 18). So we initialize G to be the graph realization of $(d_{\ell}^{n_{\ell}}, d_{\ell-1})$ as obtained from Lemma 12. We set $L_{\ell-1}$ be a valid list in G of size $d_{\ell-1} 1$. Also we decrement $n_{\ell-1}$ by one as G already contain a vertex whose MAXNDEG⁻ is $d_{\ell-1}$.

Next for each $i = \ell - 1$ to 1 we perform following steps. (i) We take as an input the valid list L_i of size $d_i - 1$, and execute Procedure ADDLAYER (G, L_i, n_i, d_i) to add n_i new vertices to G. The procedure returns a valid list L_{i-1} of size $d_i - 2$. (ii) Truncate list L_{i-1} to contain only the first $d_{i-1} - 1 (\leq d_i - 2)$ vertices. The truncated list remains valid since it is a prefix of a valid list.

Correctness. Let \bar{V}_{ℓ} denote the set of vertices in graph G initialized in steps 5, 8, or 11 of Algorithm 4, and for $i \in [1, \ell-1]$, let \bar{V}_i denote the set of new vertices added to graph G in iteration i of for loop. For $i \in [1, \ell]$, let G_i be the graph induced by vertices $\bar{V}_i \cup \cdots \cup \bar{V}_{\ell}$.

Recall that if $n_\ell = d_\ell$, then the graph is initialized in step 11 and contains $n_\ell + 1$ vertices, of which one vertex, say z, has MAXNDEG $^-(z) = d_{\ell-1}$, and the remaining vertices have MAXNDEG $^- = d_\ell$. If $n_\ell = d_\ell$, then let $Z = \{z\}$, otherwise let $Z = \emptyset$. We set $V_\ell = \bar{V}_\ell \setminus Z$, $V_{\ell-1} = \bar{V}_{\ell-1} \cup Z$, and $V_i = \bar{V}_i$ for $i \in [1, \ell-2]$. Thus $|V_i| = n_i$, for $i \in [1, \ell]$. The following lemma proves the correctness.

Lemma 17. For any $i \in [1, \ell]$, graph G_i is a MAXNDEG⁻ realization of profile $(d_\ell^{n_\ell}, \dots, d_i^{n_i})$, except for the case $n_\ell = d_\ell$ in which G_ℓ is MAXNDEG⁻ realization of profile $(d_\ell^{n_\ell}, d_{\ell-1})$. Moreover, for any $j \in [i, \ell]$,

we have

- 1. For every $v \in V_i \setminus Z$, $deg_{G_i}(v) \leq MAXNDEG^-_{G_i}(v) = d_i$.
- 2. If $n_{\ell} = d_{\ell}$, then $deg_{G_i}(z) = d_{\ell}$ and MAXNDEG $_{G_i}(z) = d_{\ell-1}$.

Proof. We prove the claim by induction on the iterations of the for loop. The base case is for index ℓ , and the claim follows from Lemmas 11, 12, and 13. Specifically, notice that every vertex $v \in V_{\ell}$ that is included in G in step 5, 8, or 11 of the algorithm has MAXNDEG $^-(v) = d_{\ell}$. In the case $n_{\ell} = d_{\ell}$, the vertex $z \in V_{\ell-1}$ included in step 11 of algorithm has MAXNDEG $^-(z) = d_{\ell-1}$. Also, in both the cases, $V_{\ell} \cup Z$ is the vertex set of G, and degree of all the vertices in this set is bounded by d_{ℓ} .

For the inductive step, we assume that the claim holds for i + 1, and prove the claim for i. Consider any vertex v in G_i . We have two cases.

- 1. $v \in V_i \setminus Z$: In this case by Proposition 2 and Lemma 14, $deg_{G_i}(v) \leq \text{MAXNDEG}^-_{G_i}(v) = d_i$.
- 2. $v \in V_j \setminus Z$, for j > i: In this case we first show that for any vertex $w \in N(v)$, $deg_{G_i}(w) \le d_j$. If $w \in V_i \setminus Z$, then we already showed $deg_{G_i}(w) \le d_i$. So we next consider the case $w \in (V_{i+1} \cup \cdots \cup V_\ell) \setminus Z$. Now if $w \in L_i$ participates in Procedure ADDLAYER (G, L_i, n_i, d_i) , then by Proposition 2 in the updated graph $deg_{G_i}(w) \le d_i \le d_j$. If $w \notin L_i$, then the degree of w is unaltered in the i^{th} iteration, and thus $deg_{G_i}(w) = deg_{G_{i+1}}(w) \le d_j$ by the inductive hypothesis. If $n_\ell = d_\ell$ and $w = z \in Z$, then also $deg_{G_i}(w) = deg_{G_{i+1}}(w)$ since vertex z never participates in procedure ADDLAYER. It follows that MAXNDEG $^-(v)$ remains unaltered due to iteration i, and thus MAXNDEG $^-(v)$ = MAXNDEG $^-(v)$ = MAXNDEG $^-(v)$ = d_j .

Now when $n_\ell=d_\ell$, then $deg_{G_\ell}(z)=d_\ell$ and MAXNDEG $^-_{G_\ell}(z)=d_{\ell-1}$. The degree of vertex z never changes since it does not participates in procedure ADDLAYER. The MAXNDEG $^-$ of z never changes from the same reasoning as above.

The execution time of algorithm takes $O(\sum_{i=1}^{\ell} (n_i + d_i))$ time, which can be easily shown to be optimal. The following theorem is immediate from the above discussions.

Theorem 8. There exists an algorithm that given any profile $\sigma = (d_{\ell}^{n_{\ell}}, \dots, d_{1}^{n_{1}}) \neq (d_{\ell}^{d_{\ell}+1}2^{1})$ with $n = n_{1}+\dots+n_{\ell}$ satisfying $d_{\ell} \leq \min\{n-1,n_{\ell}\}$ and $d_{1} \geq 2$, computes in optimal time a connected MAXNDEG⁻ realization of σ .

5.4 Complete characterization of MAXNDEG⁻ profiles.

We first give the sufficient conditions for a profile to be MAXNDEG⁻ realizable.

Lemma 18. A necessary condition for the profile $\sigma = (d_{\ell}^{n_{\ell}}, \cdots, d_{1}^{n_{1}})$ with $n = n_{1} + \cdots + n_{\ell}$ to be MAXNDEG⁻ realizable is $d_{\ell} \leq \min\{n_{\ell}, n-1\}$.

Proof. Suppose σ is MAXNDEG⁻ realizable by a graph H. Then there exists at least one vertex, say u, of degree exactly d_ℓ in H. Now $|N(u)| = d_\ell$ and $|N[u]| = d_\ell + 1$, which implies that the number of vertices in H whose MAXNDEG⁻ is d_ℓ must be at least d_ℓ , so $n_\ell \geq d_\ell$. Also, the number of vertices in the graph H, n, must be at least $d_\ell + 1$.

Consider a profile $\sigma=(d_\ell^{n_\ell},\cdots,d_1^{n_1})$ realizable by a connected graph. If $d_1=1$, then the realizing graph must contain a vertex, say u, such that each vertex in N(u) has degree 1. Let d=deg(u), and v_1,\ldots,v_d be the neighbours of u. Then $deg(v_1)=\cdots=deg(v_d)=1$. So in this case the realizing graph is a star graph $K_{1,d}$ with MAXNDEG⁻ profile $\sigma=(d^d,1^1)$. If $d_1\geq 2$, then by Lemma 18, for σ to be realizable in this case, we need that $d_\ell\leq \min\{n_\ell,n-1\}$. Also, Lemma 15 implies that σ must not be $(d^{d+1},2^1)$. By Theorem 8, under these conditions σ is always realizable. We thus have the following theorem.

Theorem 9. The necessary and sufficient condition for a profile $\sigma = (d_{\ell}^{n_{\ell}}, \dots, d_{1}^{n_{1}}) \neq (d^{d+1}, 2^{1})$ with $n = n_{1} + \dots + n_{\ell}$ to be MAXNDEG⁻ realizable by a connected graph is (i) $d_{\ell} \leq \min\{n_{\ell}, n-1\}$ and $d_{1} \geq 2$; or (ii) $\sigma = (d^{d}, 1^{1})$ for some positive integer d > 1; or (iii) $\sigma = (1^{2})$.

For general graphs we have the following theorem.

Theorem 10. The necessary and sufficient condition for a profile σ to be MAXNDEG⁻ realizable by a general graph is that σ can be split into two profiles σ_1 and σ_2 such that (i) σ_1 has a connected MAXNDEG⁻ realization, and (ii) $\sigma_2 = (1^{2\alpha})$ or $\sigma_2 = (d^d, 1^{2\alpha+1})$ for some integers $d \geq 2$, $\alpha \geq 0$.

Proof. Suppose σ is realizable by graph G. Let $\mathcal{C}(G)$ be a set consisting of all those components in G that contain a vertex of MAXNDEG⁻ equal to 1 but is not an edge. As a long as $|\mathcal{C}(G)| > 1$, we perform following modifications to G. Take any two components $C_1, C_2 \in \mathcal{C}(G)$, and let σ_1 and σ_2 be their MAXNDEG⁻ profiles. For i=1,2, component C_i must be of form K_{1,δ_i} and contain $\delta_i(\geq 2)$ vertices of MAXNDEG⁻ equal to δ_i , and a single vertex of MAXNDEG⁻ equal to 1. Let us assume $\delta_2 \geq \delta_1$. We replace C_1 and C_2 in G by two different components, namely, an edge and (i) a connected MAXNDEG⁻ realization of profile $\delta_2^{\delta_1+\check{\delta_2}}$ if $\delta_2=\delta_1$, or (ii) a connected MAXNDEG⁻ realization of profile $(\delta_2^{\delta_2},\delta_1^{\delta_1})$ if $\delta_2 > \delta_1$. In each iteration we decrease $|\mathcal{C}(G)|$ by a value two. In the end if $\mathcal{C}(G)$ is non-empty we denote the only component in it by C_0 . Next let $\bar{C}_1, \dots, \bar{C}_k$ be all those components in G that contain only the vertices of MAXNDEG⁻ strictly greater than 1. Also let $\sigma_1, \ldots, \sigma_k$ be their MAXNDEG⁻ profiles. If k > 0, we replace the components $\bar{C}_1, \ldots, \bar{C}_k$ by a single connected component, say \bar{C}_0 , that realizes the profile $\sigma_1 + \cdots + \sigma_k$. It is easy to verify from Theorem 8 that $\sigma_1 + \cdots + \sigma_k$ will be MAXNDEG⁻ realizable. The final graph G contains (i) at most one component, namely \bar{C}_0 , having all vertices of MAXNDEG⁻ greater than 1, (ii) at most one component, namely C_0 , having exactly one vertex of MAXNDEG⁻ equal to 1, and (iii) a union of some $\alpha \geq 0$ disjoint edges. This shows that σ can be split into two profiles σ_1 and σ_2 such that (i) σ_1 has a connected MAXNDEG⁻ realization, and (ii) $\sigma_2 = (1^{2\alpha})$ or $\sigma_2 = (d^d, 1^{2\alpha+1})$ for some integers $d \geq 2, \alpha \geq 0$. To prove the converse notice that $\sigma_2 = (1^{2\alpha})$ is realizable by a disjoint union of $\alpha \geq 0$ edges, and $\sigma_2 = (d^d, 1^{2\alpha+1})$ is realizable by a disjoint union of α edges and the star graph $K_{1,d}$. Thus any σ that can be split into two profiles σ_1 and σ_2 such that (i) σ_1 has a connected MAXNDEG⁻ realization, and (ii) $\sigma_2 = (1^{2\alpha})$ or $\sigma_2 = (d^d, 1^{2\alpha+1})$ for some integers $d \ge 2, \alpha \ge 0$ is MAXNDEG⁻ realizable.

6 Concluding remarks on extremal neighborhood degree profiles

Our work focuses on two similar neighborhood profiles, MAXNDEG and MINNDEG, which capture two opposing extremes of the neighborhood, but yet exhibit a surprising difference in structure. The realizability of MAXNDEG profiles depends only on their prefix; in contrast, the realizability characterization of MINNDEG profiles is incomplete and depends on the entire profile. Let us conclude with a brief discussion exploring the reasons behind this structural difference.

Let us first consider the MAXNDEG profile $\sigma=(d^{n_\ell}_\ell,\cdots,d^{n_1}_1)$ for a graph G=(V,E). For $1\leq i\leq \ell$, let $W_i\subseteq V$ be the set of vertices whose MAXNDEG in G is at least d_i . Note that for any vertex $v\in W_i$, a vertex having maximum degree in $N_G[v]$ (say x) must be contained in W_i . Moreover, all the neighbors of x must also lie in W_i . It follows that the degree of x remains unaltered when restricted to the induced subgraph $G[W_i]$, and $\text{MAXNDEG}_G(v) = \text{MAXNDEG}_{G[W_i]}(v)$. Hence, MAXNDEG profiles satisfy the following nice substructure property, which also justifies the incremental algorithm for computing their realizations given in Section 4:

³ A profile $\sigma=(d_\ell^{n_\ell},\cdots,d_1^{n_1})$ is said to be split into two profiles $\sigma_1=(d_\ell^{p_\ell},\cdots,d_1^{p_1})$ and $\sigma_2=(d_\ell^{q_\ell},\cdots,d_1^{q_1})$ if $n_i=p_i+q_i$ for each $i\in[1,\ell]$.

Substructure Property. The induced graph $G_i = G[W_i]$ is a MAXNDEG realization of the partial profile $(d_\ell^{n_\ell}, \dots, d_i^{n_i})$, for each $1 \le i \le \ell$.

A natural question is whether a similar property holds for MINNDEG profiles. Unfortunately, in this case the answer is negative. To see why, consider the MINNDEG profile $\sigma=(d_\ell^{n_\ell},\cdots,d_1^{n_1})$ for G=(V,E). If MINNDEG $_G(v)$ is d_i (for some i and v), and $x=\arg\min\{deg(x)\mid x\in N[v]\}$ is a leader of v, then the MINNDEG of all vertices in N[x] is at most d_i . But if we take W_i to be the set of all vertices whose MINNDEG in G is at most d_i , and drop the vertices z with MINNDEG $_G(z)>d_i$, i.e., look at the induced graph $G[W_i]$, then the degrees of v's neighbors might decrease, so its leader might change. Hence the substructure property does not hold, which renders an incremental construction impossible, and contributes to the intricacy of realizing MINNDEG profiles.

Nevertheless, in this work we obtain a simple 2-approximate bound on the achievable n_i 's. The problem of obtaining an exact characterization for MINNDEG profiles is left as an interesting open question for future research.

References

- [1] Martin Aigner and Eberhard Triesch. Realizability and uniqueness in graphs. *Discrete Mathematics*, 136:3–20, 1994.
- [2] Amotz Bar-Noy, Keerti Choudhary, David Peleg, and Dror Rawitz. Realizability of graph specifications: Characterizations and algorithms. In *25th SIROCCO*, volume 11085 of *LNCS*, pages 3–13, 2018.
- [3] Amotz Bar-Noy, Keerti Choudhary, David Peleg, and Dror Rawitz. Efficiently realizing interval sequences. In 30th International Symposium on Algorithms and Computation, ISAAC 2019, December 8-11, 2019, Shanghai University of Finance and Economics, Shanghai, China, pages 47:1–47:15, 2019.
- [4] Amotz Bar-Noy, Keerti Choudhary, David Peleg, and Dror Rawitz. Graph profile realizations and applications to social networks. In *13th WALCOM*, volume 11355 of *LNCS*, pages 1–12, 2019.
- [5] Michael D. Barrus and Elizabeth A. Donovan. Neighborhood degree lists of graphs. *Discrete Mathematics*, 341(1):175–183, 2018.
- [6] Joseph K. Blitzstein and Persi Diaconis. A sequential importance sampling algorithm for generating random graphs with prescribed degrees. *Internet Mathematics*, 6(4):489–522, 2011.
- [7] Sheshayya A. Choudum. A simple proof of the Erdös-Gallai theorem on graph sequences. *Bulletin of the Australian Mathematical Society*, 33(1):67–70, 1991.
- [8] Brian Cloteaux. Fast sequential creation of random realizations of degree sequences. *Internet Mathematics*, 12(3):205–219, 2016.
- [9] Paul Erdös and Tibor Gallai. Graphs with prescribed degrees of vertices [hungarian]. *Matematikai Lapok*, 11:264–274, 1960.
- [10] S. Louis Hakimi. On realizability of a set of integers as degrees of the vertices of a linear graph –I. *SIAM J. Appl. Math.*, 10(3):496–506, 1962.
- [11] V. Havel. A remark on the existence of finite graphs [in Czech]. Casopis Pest. Mat., 80:477–480, 1955.

- [12] P.J. Kelly. A congruence theorem for trees. *Pacific J. Math.*, 7:961–968, 1957.
- [13] Milena Mihail and Nisheeth Vishnoi. On generating graphs with prescribed degree sequences for complex network modeling applications. *3rd Workshop on Approximation and Randomization Algorithms in Communication Networks*, 2002.
- [14] Elchanan Mossel and Nathan Ross. Shotgun assembly of labeled graphs. *CoRR*, abs/1504.07682, 2015.
- [15] Peter V. O'Neil. Ulam's conjecture and graph reconstructions. Amer. Math. Monthly, 77:35–43, 1970.
- [16] Gerard Sierksma and Han Hoogeveen. Seven criteria for integer sequences being graphic. *Journal of Graph Theory*, 15(2):223–231, 1991.
- [17] Amitabha Tripathi and Himanshu Tyagi. A simple criterion on degree sequences of graphs. *Discrete Applied Mathematics*, 156(18):3513–3517, 2008.
- [18] S.M. Ulam. A collection of mathematical problems. Wiley, 1960.
- [19] D.L. Wang and D.J. Kleitman. On the existence of n-connected graphs with prescribed degrees (n > 2). Networks, 3:225–239, 1973.
- [20] N.C. Wormald. Models of random regular graphs. Surveys in combinatorics, 267:239–298, 1999.

Appendix

A Counting the Number of Realizable MaxNDeg Sequences

We use the characterizations of the variants of MAXNDEG in order to count the number of realizable sequences. Our results are summarized in the following theorem.

Theorem 11. For $n \geq 5$

- There are 2^{n-3} realizable sequences with connected graph in the closed neighborhood model.
- There are $2^{n-2} 1$ realizable sequences with connected graph in the open neighborhood model.
- There are $\left\lceil \frac{2^{n-1}+(-1)^n}{3} \right\rceil$ realizable sequences with any graph in the closed neighborhood model.
- There are at least $\lceil \frac{2^n-2}{3} \rceil \lceil \frac{n-4}{2} \rceil$ realizable sequences and at most $2^{n-1}-1$ realizable sequences with any graph in the open neighborhood model.

There are $(n-1)^n$ unordered sequences (d_n, \ldots, d_1) of length n on the integers $1, \ldots, n-1$. We count the number of non-increasing such sequences denoted by S_n . Let f(i, j, k) be the number of non-increasing sequences of length k on the integers i, \ldots, j . By definition, $S_n = f(1, n-1, n)$.

Observation 1.
$$f(i, j, k) = \binom{k+j-i}{k}$$
.

Proof. This is equivalent to counting the number of ways of placing k balls into j-i+1 ordered bins which is equivalent to inserting j-i dividers among a line of k balls.

Corollary 3.
$$S_n \approx \frac{4^{n-1}}{\sqrt{\pi n}}$$
.

Proof. $S_n = \binom{2n-2}{n}$ by Observation 1. Stirling's formula implies the following approximation for the central binomial coefficient $\binom{2n}{n} \approx \frac{4^n}{\sqrt{\pi n}}$. The corollary follows since $\frac{\binom{2n}{n}}{\binom{2n-2}{n}} = 4 + \frac{2}{n-1}$.

Theorem 11 and Corollary 3 imply that the number of realizable sequences in all variants is roughly $\Theta(\sqrt{S_n})$.

A.1 Connected Graphs in the Closed Neighborhood Model

Let CCON(n) be the number of length n sequences that are MAXNDEG realizable with a connected graph in the closed neighborhood model. Recall that by Theorem 6 the sequence $\sigma = (d_\ell^{n_\ell}, \dots, d_1^{n_1}) \in S_n$ can be realized with a connected graph in the closed neighborhood model if and only if one of the following holds: (i) n = 2: $\sigma = (1^2)$. (ii) $n \ge 3$: $n_\ell \ge d_\ell - 1$ and $d_1 \ge 2$.

Lemma 19.
$$CCon(2) = 1$$
 and $CCon(n) = 2^{n-3}$, for $n \ge 3$.

Proof. By the first part of the characterization, CCON(2) = 1. Assume $n \ge 3$. Let $d = d_\ell$. By the second part of the characterization, the first d+1 values in any realizable sequence must be equal to d. The suffix of length n-d-1 is a non-increasing sequence on the numbers $2, \ldots, d$. By the definition of f(i, j, k) with i = 2, j = d, and k = n - d - 1 and by Observation 1 the number of such sequences is

$$f(2,d,n-d-1) = {\binom{(n-d-1)+d-2}{n-d-1}} = {\binom{n-3}{d-2}}.$$

The value of d ranges from 2 to n-1. Hence, the total number of realizable sequences is

$$\mathrm{CCon}(n) = \sum_{d=2}^{n-1} \binom{n-3}{d-2} = \sum_{i=0}^{n-3} \binom{n-3}{i} = 2^{n-3} \; .$$

A.2 Connected Graphs in the Open Neighborhood Model

Let OCON(n) be the number of length n sequences that are MAXNDEG realizable with a connected graph in the open neighborhood model. Recall that by Theorem 9 a sequence $\sigma = (d_\ell^{n_\ell}, \dots, d_1^{n_1}) \in S_n$ can be realized with a connected graph in the open neighborhood model if and only if one of the following holds:

- (i) n=2: $\sigma=(1^2)$.
- (ii) $n \ge 3$: $\sigma = ((n-1)^{n-1}, 1^1)$.
- (iii) $n \ge 3$: $d_{\ell} \le n_{\ell}$, $d_1 \ge 2$, and $\sigma \ne ((n-2)^{n-1}, 2^1)$.

Note that the sequence in item 2 is the only sequence in which one vertex has a maximum degree 1 in its open neighborhood. It is realizable by the star graph.

Lemma 20.
$$OCon(2) = 1$$
, $OCon(3) = 2$, $OCon(4) = 4$, and $OCon(n) = 2^{n-2} - 1$, for $n \ge 5$.

Proof. Following the characterization, one could verify the following:

- (1,1) is the only realizable sequence of length 2. Therefore, OCon(2) = 1.
- (2,2,2) and (2,2,1) are the only realizable sequences of length 3. Therefore, OCon(2) = 2.
- (3, 3, 3, 3), (3, 3, 3, 2), (3, 3, 3, 1), and (2, 2, 2, 2) are the only realizable sequences of length 4. Therefore OCoN(4) = 4.

Assume $n \geq 5$. Let $d = d_{\ell}$. For the sake of counting, we assume that $((n-2)^{n-1}, 2^1)$ should be counted while $((n-1)^{n-1}, 1^1)$ should not. Hence, we need to count the sequences for which $d_{\ell} \leq n_{\ell}$ and $d_1 \geq 2$. It follows that the number of realizable sequences with $d = d_{\ell}$ is the number of sequences in which the first d values are equal to d and the suffix of length n-d is a non-increasing sequence on the numbers $2, \ldots, d$. By Observation 1 with i=2, j=d, and k=n-d the number of such sequences is

$$f(2,d,n-d) = \binom{(n-d)+d-2}{n-d} = \binom{n-2}{d-2}.$$

The value of d ranges from 2 to n-1. Hence, the total number of realizable sequences is

OCON(n) =
$$\sum_{d=2}^{n-1} {n-2 \choose d-2} = \sum_{i=0}^{n-3} {n-2 \choose i} = 2^{n-2} - 1$$
.

Observe that OCON $\approx 2 \cdot \text{CCON}(n)$. This is due to the more relaxed constraint on n_{ℓ} .

A.3 General Graphs in the Closed Neighborhood Model

Let CGEN(n) be the number of length n sequences that are MAXNDEG realizable with a general graph in the closed neighborhood model. By Theorem 7 the sequence $\sigma = (d_\ell^{n_\ell}, \dots, d_1^{n_1}) \in S_n$ can be realized with a general graph (without isolated vertices) in the closed neighborhood model if and only if the following holds for $n \geq 2$: $d_\ell \leq n_\ell - 1$, and either $d_1 \geq 2$ or n_1 is even.

Lemma 21. For
$$n \ge 2$$
, $CGEN(n) = (2^{n-1} + (-1)^n)/3$.

Proof. There are no realizable sequences of length 1 and therefore CGEN(1) = 0. The only realizable sequence of length 2 is (1,1) and therefore CGEN(2) = 1.

Assume $n \ge 3$. The first part of the characterization covers all the realizations with connected graphs while the second part of the characterization covers all the realizations with n-2 vertices plus an isolated edge. As a result, we get the following recursive formula,

$$CGen(n) = CGen(n-2) + CCon(n) = CGen(n-2) + 2^{n-3}.$$

We prove by induction that the lemma follows from this recursion. The claim holds for the two base cases n=1 and n=2 since $(2^0+(-1)^1)/3=0$ and $(2^1+(-1)^2)/3=1$. Assume that the claim is correct for n-2, that is that $CGEN(n-2)=(2^{n-3}+(-1)^{n-2})/3$. It follows that $CGEN(n)=(2^{n-3}+(-1)^{n-2})/3+2^{n-3}=(2^{n-1}+(-1)^n)/3$.

A.4 General Graphs in the Open Neighborhood Model

Let OGEN(n) be the number of length n sequences that are MAXNDEG realizable with a general graph in the open neighborhood model.

We do not know how to compute the exact value of OGEN(n) based on our complete characterization. The main reason is that we do not know how to avoid counting more than once a sequence that has several realizations with one star graph where in each realization the size of the star is different. For example, consider the sequence $(3^6, 2^2, 1^1)$. It can be realized with a 3-regular graph of size 6 whose MAXNDEG sequence is (3^6) and a star of size 3 whose MAXNDEG sequence is $(2^2, 1^1)$. It can also be realized by a cycle of size 4 that is connected to a vertex of degree 1 whose MAXNDEG sequence is $(3^3, 2^2)$ and a star of size 4 whose MAXNDEG sequence is $(3^3, 1^1)$. The problem is that the strategy of extracting the star and counting the number of realizations for the remaining sequence would count more than once sequences from which we can extract stars of different sizes.

Instead we provide characterizations for under and over counting. On one hand, we count most of the sequences that can be realized and on the other hand, we count all the realizable sequences, but also some sequences that cannot be realized. Specifically, we show two functions OGENL(n) and OGENU(n) such that $OGENL(n) \leq OGEN(n) \leq OGENU(n)$, for $n \geq 2$.

Let OGENL(n) be the number of sequences $\sigma = (d_\ell^{n_\ell}, \dots, d_1^{n_1}) \in S_n$ that can be realized with a general graph in the open neighborhood model if one of the following holds for $n \geq 2$: (i) $d_\ell \leq n_\ell$ and $d_1 \geq 2$. (ii) $d_\ell \leq n_\ell$, $d_1 = 1$, and n_1 is even.

Lemma 22. OGENL
$$(n) \leq OGEN(n)$$
, for every $n \geq 1$.

Proof. OGENL(n) counts all the realizations with one connected component and a collection of isolated edges. The connected component could be a star. However, sequences that can be realized with a connected component, a star and a collection of isolated edges are not counted.

Lemma 23. OGENL(2) = 1 and OGENL(n) =
$$\lceil (2^n - 2)/3 \rceil - \lceil (n - 4)/2 \rceil$$
, for $n \ge 3$.

Proof. One can verify the following:

- 1. The sequence (1,1) is the only realizable sequence and therefore OGENL(2) = 1.
- 2. The sequences (2,2,2) and (2,2,1) are realizable and therefore we can set OGENL(3) = 2.
- 3. The sequences (3,3,3,3), (3,3,3,2), (3,3,3,1), (2,2,2,2), and (1,1,1,1), are realizable and therefore we can set OGENL(4) = 5.

Lemma 22 implies the following recessive formula for $n \ge 4$,

$$OGEN(n) = OGENL(n-2) + OCON(n) = OGENL(n-2) + (2^{n-2} - 1)$$
.

We prove by induction that the lemma follows from this recursion. The claim holds for the two base cases n=3 and n=4 since $\lceil (2^3-2)/3 \rceil - \lceil (3-4)/2 \rceil = 2$ and $\lceil (2^4-2)/3 \rceil - \lceil (4-4)/2 \rceil = 5$. The induction hypothesis for n-2 implies that $\mathrm{OGENL}(n) = \lceil (2^{n-2}-2)/3 \rceil - \lceil (n-6)/2 \rceil + (2^{n-2}-1)$. For an even n, we have

$$\mathrm{OGENL}(n) = \frac{2^{n-2}-1}{3} - \frac{n-6}{2} + (2^{n-2}-1) = \frac{2^n-1}{3} - \frac{n-4}{2} = \left\lceil \frac{2^n-2}{3} \right\rceil - \left\lceil \frac{n-4}{2} \right\rceil \;,$$

and for an odd n,

$$OGENL(n) = \frac{2^{n-2}-2}{3} - \frac{n-5}{2} + (2^{n-2}-1) = \frac{2^n-2}{3} - \frac{n-3}{2} = \left\lceil \frac{2^n-2}{3} \right\rceil - \left\lceil \frac{n-4}{2} \right\rceil.$$

Let OGENU(n) be the number of non-increasing sequences $\sigma=(d_\ell^{n_\ell},\dots,d_1^{n_1})\in S_n$ that satisfy $d_\ell\leq n_\ell$ for $n\geq 2$.

Lemma 24. OGENU $(n) \ge OGEN(n)$, for every $n \ge 1$.

Proof. By Theorem 10 In any realizable sequence, d_{ℓ} cannot be larger than $\min\{n_{\ell}, n-1\}$.

Lemma 25. OGENU(2) = 1 and OGENU(n) = $2^{n-1} - 1$, for $n \ge 2$.

Proof. For n=2, (1,1) is the only sequence and therefore OGENU(2)=1. Assume $n\geq 2$. Let $d=d_\ell$. The first d values in any realizable sequence must be equal to d. The suffix of length n-d is a non-increasing sequence on the numbers $1,\ldots,d$. By Observation 1 with $i=1,\,j=d$, and k=n-d the number of such sequences is

$$f(1,d,n-d) = \binom{(n-d)+d-1}{n-d} = \binom{n-1}{d-1}.$$

The value of d ranges from 1 to n-1. Hence, the total number of realizable sequences is

$$\mathrm{OGENU}(n) = \sum_{d=1}^{n-1} \binom{n-1}{d-1} = \sum_{i=0}^{n-2} \binom{n-1}{i} = 2^{n-1} - 1 \; .$$

Observe that the ratio between the upper bound and the lower bound is about 3/2.