Efficient Computation of Sparse Structures[*]

David G. Harris[**], Ehab Morsy[***], Gopal Pandurangan[†], Peter Robinson[‡], and Aravind Srinivasan[§]

Abstract. Basic graph structures such as maximal independent sets (MIS's) have spurred much theoretical research in randomized and distributed algorithms, and have several applications in networking and distributed computing as well. However, the extant (distributed) algorithms for these problems do not necessarily guarantee fault-tolerance or load-balance properties. We propose and study "low-average degree" or "sparse" versions of such structures. Interestingly, in sharp contrast to, say, MIS's, it can be shown that checking whether a structure is sparse, will take substantial time. Nevertheless, we are able to develop good sequential/distributed (randomized) algorithms for such sparse versions. We also complement our algorithms with several lower bounds. Randomization plays a key role in our upper and lower bound results.

Keywords: Graph algorithms, Distributed algorithms, Randomization, Approximation algorithms, Maximal Independent Set, Lower Bounds.

1 Introduction

Graph-theoretic structures such as maximal independent sets (MIS's) and minimal dominating sets (MDS's) are fundamental to graph theory, and their efficient computation is especially useful in the context of distributed computing and networks [13]. MIS, for example, is a basic building block in distributed computing and is useful in basic tasks such as monitoring, scheduling, routing, and clustering [12, 14]; furthermore, the development of fast parallel/distributed algorithms for it has spurred fundamental progress in randomized algorithms and in derandomization [1, 7, 11]. Extensive research has gone into designing fast distributed algorithms for these problems since the early eighties: see [9, 18] and the references therein. We now know that problems such as MIS are quite *local*, i.e., that they admit distributed algorithms that run in a small number of *rounds*:

[*] A preliminary version of this work appears in the *Proc. International Colloquium on Automata, Languages, and Programming* (ICALP), 2013.

[**] Department of Applied Mathematics, University of Maryland, College Park, MD 20742. davidgharris29@hotmail.com. Supported in part by by NSF Awards CNS-1010789 and CCF-1422569.

[***] Division of Mathematical Sciences, Nanyang Technological University, Singapore 637371 and Department of Mathematics, Suez Canal University, Ismailia 22541, Egypt. ehabmorsy@gmail.com.

[†] Department of Computer Science, University of Houston, Houston, TX 77204, USA. Work done while the author was affiliated with the Division of Mathematical Sciences, Nanyang Technological University, Singapore 637371 and Department of Computer Science, Brown University, Providence, RI 02912, USA. gopalpandurangan@gmail.com. Supported in part by Nanyang Technological University grant M58110000, Singapore Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 2 grant MOE2010-T2-2-082, Singapore MOE AcRF Tier 1 grant MOE2012-T1-001-094, and by the United States-Israel Binational Science Foundation grant 2008348.

[‡] Queen's University Belfast, United Kingdom. p.robinson@qub.ac.uk. Supported in part by Nanyang Technological University grant M58110000, Singapore Ministry of Education (MoE) Academic Research Fund (AcRF) Tier 2 grant MOE2010-T2-2-082, and MOE2011-T2-2-042 "Fault-tolerant Communication Complexity in Wireless Networks" from the Singapore MoE AcRF-2.

[§] Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742. srin@cs.umd.edu. Supported in part by NSF Awards CNS 1010789 and CCF 1422569.

typically (poly-)logarithmic in the network size n (n will denote the number of nodes in the network throughout unless specified otherwise). However, one main drawback of these algorithms is that there is no guarantee on the *quality* of the structure output. For example, the classical distributed MIS algorithms of Alon, Babai & Itai [1] and Luby [11] compute an MIS in $O(\log n)$ rounds with high probability; their focus is not on additional properties of the output MIS. In this paper, we initiate a systematic study of "sparse" versions of these structures, i.e., the *average degree* – in the original graph – of the nodes belonging to the structure is "small"; this study is motivated by both theoretical and practical considerations.

We are not aware of previous work on adding additional sparse constraints to such classic graph structures. Closest to our work, is the work of [4], which consider algorithms that compute "fair" MIS on certain graph classes. In contrast to the classic MIS algorithms (cf. [1, 11]), a fair algorithm must ensure that all nodes have roughly the same probability of entering the MIS.

1.1 Problems Addressed

We consider an undirected simple graph $G = (V, E)$ with n nodes and m edges. We denote the average degree of G by $d = d(G) = \frac{2m}{n}$; *this parameter will play a key role in our results.* More generally, given any subset $S \subseteq V$, we define the average degree of S, denoted by d_S, as the total degree (in G) of the vertices of S divided by the number of vertices in S, i.e., $d_S = \frac{\sum_{v \in S} d_v}{|S|}$, where d_v is the degree of node v in G. We assume that G has no isolated vertices for convenience; this assumption can be easily removed.

Recall the following fundamental graph structures:

- a Maximal Independent Set (MIS) is an inclusion-maximal vertex subset $S \subseteq V$ such that no two vertices in S are neighbors;
- a Minimal Dominating Set (MDS) is an inclusion-minimal vertex subset $S \subseteq V$ such that every vertex in G is either in S or is a neighbor of a vertex in S; and
- a Minimal Vertex Cover (MVC) is an inclusion-minimal vertex subset $S \subseteq V$ such that every edge in G has at least one endpoint in S.

This paper is concerned with the "sparse" versions of these problems:

1. Sparse Maximal Independent Set (SMIS): Given an undirected graph G, a SMIS is an MIS S in G that minimizes d_S. In other words, the SMIS has the minimum average degree (in G) among all MIS's in G.
2. Sparse Minimal Dominating Set (SMDS): Given an undirected graph G, a SMDS is an MDS D in G that minimizes d_D.
3. Sparse Minimal Vertex Cover (SMVC): Given an undirected graph G, a SMVC is an MVC C in G that minimizes d_C.

We note that the *maximum* independent set in a graph G – a well-studied NP-hard problem [6] – is not necessarily a SMIS in G. Consider the graph G that contains a complete graph K_p (assume p is even), and a complete bipartite graph $K_{A,B}$ with $|A| = 2$ and $|B| = 3$. Each vertex in A is connected to a different half of the set of vertices in K_p (i.e., one vertex of A is connected to one half of vertices of K_p and the second vertex of A is connected to the other half of K_p), and each vertex in B is connected to all vertices in K_p. Clearly, B is the maximum independent set in G and has average degree $p + 2$, while A is a SMIS in G since its average degree is $p/2 + 3$. Thus SMIS is quite different compared to the maximum independent set problem: in sharp contrast to the standard MIS, a maximum independent set may be very different from any SMIS.

1.2 Motivations

One key motivation for our work is understanding the complexity of local computation of globally optimal (or near optimal) fundamental structures. The correctness of structures such as MIS or MDS can be verified *strictly locally* by a distributed algorithm.[1] In the case of MIS, for example, each node can check the MIS property by communicating only with its neighbors; if there is a violation at least one node will raise an alarm. On the other hand, it is not difficult to show that the correctness of sparse structures such as SMIS cannot be locally verified (in the above sense) as the SMIS refers to a "global" property: nodes have to check the small average degree property, in addition to the MIS property. In fact, it can be shown that, for any D such that $D \in o(n)$, there is a graph of diameter D, where it takes at least $\Omega(D)$ rounds to check whether a given MIS is (a constant approximation of) a SMIS: Consider the graph consisting of a line $L = (u_1, \ldots, u_D)$ and a cycle C of $(n-D)$ nodes, and assume that node u_D has an edge to each node in C. One way to select an MIS is to select nodes $u_1, u_3, \ldots, u_{D-1}$ and $\Theta(n-D)$ nodes from C: this yields an MIS S with constant average degree. On the other hand, the MIS S' formed by selecting $u_1, u_3, \ldots, u_{D-3}, u_D$ has an average degree of $\Theta(n/D) \in \omega(1)$. For node u_1, however, it takes D time to distinguish between S and S'.

Moreover, we prove that SMIS is an NP-hard problem and hence the optimality of the structure is not easy to check even in a centralized setting. A key issue that we address here is whether one can compute near-optimal local (distributed) solutions to sparse global structures such as SMIS. A main result of this paper is that despite the global nature, we can design fast *distributed* algorithms that output high quality sparse structures.

Our work is also a step toward understanding the algorithmic complexity of a few basic sparse problems. While every MIS is an MDS, these two differ significantly in their balanced versions. In particular, we show that there exist graphs for which no MIS is a good SMDS. Hence we need a different approach to compute a good SMDS as compared to a good SMIS. Even for SMIS, we show that while one can (for example) use Luby's algorithm [11] to efficiently compute an MIS, the same approach fails to compute a good quality SMIS. We present new algorithms for computing such (approximately) sparse structures.

In distributed networks, especially in resource-constrained networks such as ad hoc, sensor and mobile networks, it is important to load-balance tasks among nodes. This is crucial in extending the lifetime of the network (see e.g., [20] and the references therein). For example, in a typical application, an MIS (or an MDS) can be used to form clusters with low diameter, with the nodes in the MIS being the "clusterheads" [12]. Each clusterhead is responsible for monitoring the nodes that are adjacent to it. Having an MIS with low degree is useful in a resource/energy-constrained setting since the number of nodes monitored *per* node in the MIS will be low (on average). This can lead to better load balancing, and consequently less resource or energy consumption per node, which is crucial for ad hoc and sensor networks, and help in extending the lifetime of such networks while also leading to better fault-tolerance. For example, in an n-node star graph, the above requirements imply that it is better for the leaf nodes to form the MIS rather than the central node alone. In fact, the average degree of the MIS formed by the leaf nodes – which is 1 – is within a constant factor of the average degree of a star (which is close to 2), whereas the average degree, $n-1$, of the MIS consisting of the central node alone is much larger.

[1] As is common in distributed verification (e.g., [5]), we require that all nodes output "yes" when given a valid instance; otherwise at least one node must output "no".

Obtaining fast distributed algorithms for sparse structures is an important focus of our work. As the extensive research (e.g., [1, 11]) on obtaining fast distributed algorithms for the classical MIS problem shows, obtaining fast distributed algorithms for the sparse versions of these problems are both well-motivated and challenging. Unlike MIS, the additional challenge is to obtain solutions that also obey the sparse constraint and the goal is to accomplish this using a localized algorithm, i.e., that takes only a small number of communication rounds. We note that a distributed algorithm that takes k rounds requires each node to get information within only its k-neighborhood; hence when k is small (e.g., logarithmic in the network size), it implies that the problem can be solved using a small amount of local information. In this paper, we present a fast distributed algorithm for the SMIS problem. A similar algorithm for the SMDS problem was left open. In a subsequent work [10], a fast distributed algorithm for the SMDS problem was presented; this algorithm is based on an efficient distributed implementation of the centralized algorithm of the present paper.

1.3 Our Results

Randomization is a vital component of our positive and negative results. We first note that the trivial lower bound is d for all sparse problems which follows from the example of a regular graph (where all nodes have the same degree). Hence, in general, the average degree of a balanced structure cannot be guaranteed to be less than d. On the other hand, there exist graphs where the average degree of the SMIS is significantly smaller than d (e.g., consider a graph in which $n/2$ nodes form a complete subgraph, with these nodes connected by a perfect matching to the remaining $n/2$). This leads us to two basic questions: (i) In every given graph G, does there always exist a SMIS whose average degree is at most d? and (ii) Can question (i) be answered for a specific graph G in polynomial time? We answer both questions in the negative.

The well-known probabilistic proof of Turán's theorem on independent sets [2, 19, 17] motivates question (i), and sheds light in an interesting way on the SMIS problem. Recall that in this probabilistic approach, we construct an independent set in G as follows: randomly permute the vertices, and construct an independent set I in which we put a vertex v iff no neighbor of v precedes v in the permutation constructed. Note that $P(v \in I) = 1/(d_v + 1)$. Thus, $E[|I|] = \sum_v 1/(d_v + 1)$, which is at least $n/(d + 1)$ by convexity; and, letting T denote the total degree (in G) of I, we have

$$E[T] = \sum_v \frac{d_v}{d_v + 1} < n.$$

Thus, heuristically "$E\left[\frac{T}{|I|}\right] \leqslant O(d)$"; this is also true rigorously at least in the case where all degrees are small, in which case we can show that $|I|$ is concentrated around its mean (e.g., by a second-moment calculation). That is, there is an independent set I with $d_I = O(d)$. Note, however, that I is an independent set, not necessarily an MIS. Nevertheless, this argument appears to suggest that there is an MIS S with $d_S = O(d)$ for all graphs. Our theorems contradict this, show that "$O(d^2)$" is the truth here instead of $O(d)$, and also develop good distributed versions of this result.

We show that unlike MIS, its balanced version, SMIS, is NP-hard. In particular, we show that the following *decision version* of the problem is NP-complete (cf. Theorem 7 in Section 2.4): *"Given a graph G, is there an MIS in G with average degree at most d?"* In fact we show that the optimization version SMIS is hard to approximate in polynomial time (unless NP=ZPP) to within a factor of $\Omega(\sqrt{n})$ (cf. Theorem 8 in Appendix 2.5).

Henceforth, we focus on obtaining solutions for SMIS that are good *compared* to the average degree of the graph. We show that we can obtain near-tight solutions that compare well with d. The following are our main results:

Theorem 1. *There is a (centralized) deterministic algorithm that selects an MIS of average degree at most $d^2/8 + O(d)$ and runs in $O(m \log n)$ time.*

To prove this theorem, we show that Luby's MIS algorithm [11] returns an MIS with average degree at most $d^2/8 + O(d)$ with positive probability. This can be derandomized using the method of conditional expectations. However, this does not yield a fast distributed algorithm. Our average-degree bound here is near-optimal, as we show an almost-matching lower bound (thereby also answering question (i) posed above in the negative):

Observation 1 *For any real number $\alpha > 1$, there is a graph G with average degree at most α, but in which every MIS has average degree at least $\alpha^2/8 + 3\alpha/4 + 5/8$.*

We next consider *distributed* approximation algorithms for SMIS and show that we can output near-optimal solutions fast, i.e., solutions that are close to the lower bound. We consider the following standard model for our distributed algorithms where the given graph G represents a system of n nodes, with each node having a distinct ID [13]. Each node runs an instance of the distributed algorithm and the computation advances in synchronous *rounds*, where, in each round, nodes can communicate with their neighbors in G by sending messages of size $O(\log n)$. A node initially has only *local knowledge* limited to itself and its neighbors. We assume that local computation (performed by the node itself) is free as long it is polynomial in the network size. Each node u has local access to a special bit (initially 0) that indicates whether u is part of the output set. Our focus is on the *time complexity*, i.e., the number of rounds of the distributed computation.

We present two distributed algorithms for SMIS in Section 2.2, both running in polylogarithmic rounds with high probability.[2] The second algorithm gives a better bound on the average degree at the cost of somewhat increased run-time.

Theorem 2. *Consider a graph $G = (V, E)$ with average degree d.*

1. *There is a distributed algorithm that runs in $O(\log n)$ rounds and with high probability outputs an MIS with average degree $O(d^2)$.*
2. *For any $\epsilon > 0$, there is a distributed algorithm that runs in $O(\log^2 n)$ rounds and with high probability outputs an MIS with average degree $(1 + \epsilon)(d^2/4 + d)$.*

We also give a deterministic parallel algorithm.

Theorem 3. *Consider a graph $G = (V, E)$ with average degree d. There is a NC algorithm that runs in time $O(\log^2 n)$ and outputs an MIS with average degree $d^2/8 + O(d)$.*

Note that in general, due to the lower bound of Observation 1, the bounds provided by algorithms of the above two theorems are optimal up to constant factors.

We next present results on SMDS. Since an MIS is also an MDS, an algorithm for MIS can also be used to output an MDS. However, this can lead to a poor approximation guarantee, since there are graphs for which *every* MIS has a very large average degree compared to some MDS. This follows from the graph family used in Observation 1: while the average degree of *every* MIS – of

[2] We say an event occurs with high probability if it has probability $\geqslant 1 - n^{-\Omega(1)}$.

any graph in the family – is $\Omega(d^2)$, there exists an MDS with average degree only $O(d)$. Because an MIS is also an MDS, the results of Theorem 2 also hold for SMDS. Our next theorem shows that much better guarantees are possible for SMDS.

Theorem 4. *Any graph G with average degree d has a minimal dominating set with average degree at most $O(\frac{d \log d}{\log \log d})$. Furthermore, there is a sequential deterministic algorithm to find such an MDS in time $O(m)$.*

The next theorem shows that the bound of Theorem 4 is optimal in general up to constant factors:

Theorem 5. *For any real number $\alpha > 0$, there are graphs with average degree $\leqslant \alpha$, but for which any MDS has an average degree of $\Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$.*

Finally, we show that there cannot be any bounded approximation algorithm for SMVC:

Observation 2 *For any real number $\alpha > 2$, there are graphs for which the average degree is at most α, but for which the average degree of any MVC is arbitrarily large.*

2 Sparse Maximal Independent Set (SMIS)

We first prove Observation 1 which shows that there are graphs G for which the degree of every MIS is much larger than d. More importantly, the theorem gives a lower bound on the quality of SMIS in general: one cannot guarantee an MIS whose average degree is less than $\frac{d^2}{8} + \Theta(d)$.

Observation 1. *For any real number $\alpha > 1$, there is a graph G with average degree at most α, but in which every MIS has average degree at least $\alpha^2/8 + 3\alpha/4 + 5/8$.*

Proof. Consider the graph consisting of a copies of K_b, as well one copy of $K_{c,c}$, where $b = \lfloor \frac{3+\alpha}{2} \rfloor$ and $c = \lfloor \frac{1}{2}\sqrt{2ab(\alpha - b + 1) + \alpha^2} + \alpha \rfloor$.

The resulting graph has average degree $\frac{ab(b-1)+2c^2}{ab+2c} \leqslant \alpha$. Every MIS of this graph contains one vertex from each K_b, as well as one half of the vertices of $K_{c,c}$, for an average degree of $\frac{ab+c^2}{a+c}$. As a tends to infinity, such average degree increasingly approaches $\frac{(3+\alpha-b)b}{2} \geqslant \alpha^2/8 + 3\alpha/4 + 5/8$.

2.1 An Almost-Optimal Sequential SMIS Algorithm

In this section, we prove Theorem 1. To do so, we use Luby's algorithm for MIS [11], which goes through a number of rounds which can be described as follows. Every vertex v choose a rank ρ_v uniformly and independently from the real interval $[0, 1]$. Any vertex whose rank is lower than all its neighbors is then selected for the independent set. Such vertices and their neighbors are removed from the graph. After a sufficient number of rounds have passed, this process forms an MIS.

We show that if the original graph has average degree d, then Luby's algorithm may select an MIS of average degree $\approx d^2/8$, with a small but positive probability. We then derandomize this process to obtain an determinmistic algorithm to find such an MIS. Observation 1 shows that the average degree bound obtained is close to optimal.

One key technical tool for this proof is definining, for any independent set I and a "target degree" t, the random variable

$$\Psi_t(I) = \sum_{v \in I} (d_G(v) - t)$$

It is not hard to see that $\Psi_t(I) \leqslant 0$ iff $d_I \leqslant t$.

Lemma 1. *Suppose that I is the MIS obtained after running Luby's algorithm, and let $t = \frac{d^2}{8} + \frac{7d}{4}$. Then $\mathbf{E}[\Psi_t(I)] \leqslant -\Omega(1)$.*

Proof. Let B denote the set of vertices v such that $d_v > t$ (the "big" vertices), and let $S = V - B$ denote the set of vertices v such that $d_v \leqslant t$ (the "small" vertices.) Roughly speaking, our goal is to choose an MIS of degree $\leqslant t$, and for this end adding small vertices helps us while adding big vertices hurts us.

For each vertex v, let x_v (respectively y_v) denote the number of neighbors in S (respectively B) neighbors, so $x_v + y_v = d_v$.

We have

$$\begin{aligned}\mathbf{E}[\Psi_t(I)] = \sum_v (d_v - t)P(v \in I) &= \sum_{v \in S}(d_v - t)P(v \in I) + \sum_{v \in B}(x_v + y_v - t)P(v \in I) \\ &\leqslant \sum_{v \in S}(d_v - t)P(v \in I) + \mathbf{E}\big[\sum_{v \in B} y_v[v \in I]\big] + \\ &+ \quad \sum_{v \in B}(x_v - t)P(v \in I)\end{aligned}$$

where here the expression $[v \in I]$ is the Iverson notation, which is one if $v \in I$ and zero otherwise.

Let us consider these terms in turn. For $v \in S$, the expression $d_v - t$ is negative; hence to upper-bound $\mathbf{E}[\Psi_t(I)]$ we must *lower-bound* the probability that $v \in I$. To do so, we note that if v is selected in the first round, then $v \in I$; the event that v is selected in the first round has probability $\frac{1}{d_v+1}$.

Next, note that the expression $\sum_{v \in B} y_v[v \in I]$ counts the number of edges, both of whose endpoints are in B, that are adjacent to a vertex in I. As I is independent, each edge whose endpoints are in B may be counted at most once in this sum, so we have $\sum_{v \in B} y_v[v \in I] \leqslant \sum_{v \in B} y_v/2$ for any independent set I.

Finally, for $v \in B$, we claim that $(x_v - t)P(v \in I) \leqslant x_v/2$. This is obvious if $x_v \leqslant t$ (in which case the LHS is negative and the RHS is positive), so suppose $x_v \geqslant t$. In this case we will *upper-bound* the probability that $v \in I$.

To show this upper bound, we consider the probability that v was excluded from I in the first round. To do so, consider all the small neighbors of v. Suppose that, for $u \in N(v) \cap S$, the rank of vertex u is smaller than all its neighbors (including v), as well as all the other small neighbors of v.[3] This occurs with probability $\frac{1}{|(N(v) \cap S) \cup N(u)|} \geqslant \frac{1}{d_u + x_v}$. If this event occurs, then vertex u is selected and vertex v is unavailable for I. Furthermore, for each of the small neighbors of v, the corresponding events are mutually exclusive. Hence we get

$$\begin{aligned}P(v \in I) \leqslant P(v \text{ available after 1st round}) &\leqslant 1 - \sum_{u \in N(v) \cap S} \frac{1}{d_u + x_v} \\ &\leqslant 1 - \sum_{u \in N(v) \cap S} \frac{1}{t + x_v} \\ &= 1 - x_v \times \frac{1}{t + x_v} = \frac{t}{t + x_v}\end{aligned}$$

[3] The notation $N(v)$ denotes the neighborhood of v, that is, the set of all vertices with an edge to v.

So $(x_v - t)P(v \in I) \leqslant (x_v - t)\frac{t}{t+x_v}$; simple calculus show that this at most $0.172 x_v \leqslant x_v/2$ as claimed.

Now putting these three estimates into our bound on $\mathbf{E}[\Psi_t(I)]$:

$$\begin{aligned}\mathbf{E}[\Psi_t(I)] &\leqslant \sum_{v \in S} \frac{d_v - t}{d_v + 1} + \sum_{v \in B} y_v/2 + \sum_{v \in B} x_v/2 \\ &\leqslant \sum_{v \in S} \frac{d_v - t}{d_v + 1} + \sum_{v \in B} d_v/2\end{aligned}$$

Now let the averages degrees of S, B be d_S, d_B respectively. We have $|S| + |B| = n$ and $|S|d_s + |B|d_B = nd$. By concavity we have

$$\mathbf{E}[\Psi_t(I)] \leqslant n \frac{d_B d_S (d + 1 - d_S) - 2t(d_B - d) + d(d_B - 2d_S)}{2(d_S + 1)(d_B - d_S)}.$$

Routine calculus shows that this achieves its maximum value at $d_B = t$ and $d_S = \frac{\sqrt{2t}(t+1) - 3t}{t-2}$, yielding

$$\mathbf{E}[\Psi_t(I)] \leqslant n \frac{2\sqrt{2t}(d - t) + t(d - 1) + 2d}{2(t + 1)}.$$

For $t = d^2/8 + 7d/4$, we have

$$\mathbf{E}[\Psi_t(I)] \leqslant n(-\Omega(1/d)) \leqslant -\Omega(1).$$

This result immediately gives us a randomized, polynomial-time algorithm to find an MIS of degree $\leqslant d^2/8 + 7d/4$. But we can do better.

Theorem 1. *There is a (centralized) deterministic algorithm that selects an MIS of average degree at most $d^2/8 + \frac{7d}{4}$ and runs in time $O(m \log n)$.*

Proof. We will use the method of conditional expectations to derandomize Lemma 1. Although Lemma 1 discusses the full Luby algorithm, which has $\Theta(\log n)$ rounds, in fact our analysis of Luby's algorithm has been limited to its first round. Suppose that Luby's algorithm uses the ranks $\rho(v)$ in its first round, and I is the final MIS. We may define $\Psi'_t(\rho)$, which serves as a pessimistic estimator for $\Psi_t(I)$:

$$\begin{aligned}\Psi'_t(\rho) = &\sum_{v \in S} \Big[\bigwedge_{w \in N(v)} \rho(v) < \rho(w) \Big] (d_v - t) \\ &+ \sum_{v \in B} y_v/2 + \sum_{v \in B} (x_v - t)\Big(1 - \sum_{u \in S \cap N(v)} \Big[\bigwedge_{\substack{w \in N(u) \\ \cup (S \cap N(v))}} \rho(u) < \rho(w) \Big]\Big)\end{aligned}$$

Assuming that all the ranks ρ are distinct, we have already shown in the proof of Lemma 1 that $\Psi_t(I) \leqslant \Psi'_t(\rho)$, and $\mathbf{E}[\Psi'_t(\rho)] \leqslant -\Omega(1)$. So we really only need to derandomize a single round of Luby's algorithm, producing an set of distinct ranks ρ with $\Psi'_t(\rho) \leqslant 0$.

As we have stated Luby's algorithm, each vertex selects a rank in the real interval $[0, 1]$. We claim that it suffices to select the ranks from the integer interval $\{0, 1, \ldots, 2^{10\lceil \log_2 n \rceil} - 1\}$. For, we may refine the potential function as

$$\Psi''_t(\rho) = \Psi'_t(\rho) + mn \sum_{\substack{v, v' \in V \\ v \neq v'}} [\rho(v) = \rho(v')]$$

It is not hard to see that $\Psi''_t \leqslant 0$ iff all the vertices have distinct values of ρ and $\Psi'_t(\rho) \leqslant 0$. Furthermore, $\mathbf{E}[\Psi''_t] \leqslant \mathbf{E}[\Psi'_t(I')] + mn \sum_{v,v'} P(\rho(v) = \rho(v')) \leqslant -\Omega(1) + n^2 n^{-10} \leqslant 0$.

Finally, to turn this into a deterministic algorithm, we need to select ρ such that $\Psi''_t(\rho) \leqslant 0$. We apply the method of conditional expectations, setting the values of $\rho(v)$ bit-by-bit.

For $i = 1, \dots, 10\lceil \log_2 n \rceil$, and for each vertex v in turn, we bisect the range of $\rho(v)$. Initially, each $\rho(v)$ is drawn uniformly and independently from the range $\{0, 1, \dots, 2^{10\lceil \log_2 n \rceil} - 1\}$; after stage i, each random variable $\rho(v)$ is drawn uniformly from an integer interval of length exactly $2^{10\lceil \log_2 n \rceil - i}$. To implement this, we must be able to calculate the expected value of $\mathbf{E}[\Psi''_t]$ when the random variables $\rho(v)$ are drawn uniformly from these intervals. The expected value of $\mathbf{E}[\Psi''_t]$ only depends on comparing the sizes of $\rho(v), \rho(w)$ where v, w are neighbors of each other. This only depends on the size of the overlaps of the ranges of $\rho(v), \rho(w)$ and the sizes of the ranges of $\rho(v), \rho(w)$ themselves. One can show that, in time $O(d_v)$, one can determine the change in $\mathbf{E}[\Psi''_t]$ when we fix a bit of $\rho(v)$.

Thus, the total time for each round i of this bisection procedure is $O(\sum_v d_v) = O(m)$. The total time for the bisection is $O(m \log n)$.

At the end of this procedure, $\rho(v)$ is determined exactly and $\Psi''_t \leqslant 0$. Thus, I' has been constructed deterministically such that $\Psi'_t(I') \leqslant 0$ and hence $\Psi_t(I) \leqslant 0$ and hence $d_I \leqslant d^2/8 + 7d/4$ as desired.

Theorem 3. *There is an NC (deterministic parallel) algorithm that selects an MIS of average degree at most $d^2/8 + \frac{7d}{4}$ and runs in time $O(\log^2 n)$.*

Proof. As we have shown in the proof of Theorem 1, it suffices to find a value for the ranks $\rho(v)$ such that $\Psi''_t(\rho) \leqslant 0$; furthermore, if the bits of ρ are selected at random then we have that $\mathbf{E}[\Psi''_t] \leqslant \mathrm{poly}(1/n)$.

Now observe that Ψ''_t can be written as a sum of a polynomial number of terms, each of which can be written as an indicator function of the form $[\rho(v) < \min_{u \in X} \rho(u)]$. Such an indicator function can be computed via a log-space function of ρ; essentially, for a given vertex v, one only needs to keep track of $\rho(v)$ and the current running minimum value of $\rho(u), u \in X$. As shown in [16], there is an NC algorithm, running in time $O(\log^2 n)$ and polynomial space to produce a probability distribution D, of polynomial support size, which "fools" such log-space statistical functions to within relative error n^{-a}, for any fixed value of a. That is, the expected value of $\Psi''_t(\rho)$, when $\rho \sim D$, differs by a factor of most n^{-a} from its expectation of unbiased bits. As $\mathbf{E}[\Psi''_t] \leqslant \mathrm{poly}(1/n)$, then there is some constant a sufficiently large such that $\mathbf{E}[\Psi''_t(\rho)] \leqslant 0$ for $\rho \sim D$. One may search the full support of this space to find ρ such that $\Psi''_t(\rho) \leqslant 0$ as desired.

2.2 Distributed Algorithms for SMIS

This section is devoted for designing different distributed algorithms for SMIS. In particular we will prove Theorem 2. The proposed algorithms do not require any global information of the original graph, not even knowledge of the network size n.

Theorem 2. *Consider a graph $G = (V, E)$ with average degree d.*

1. *There is a distributed algorithm that runs in $O(\log n)$ rounds and with high probability outputs an MIS with average degree $O(d^2)$.*
2. *For any $\epsilon > 0$, there is a distributed algorithm that runs in $O(\log^2 n)$ rounds and with high probability outputs an MIS with average degree $(1 + \epsilon)(d^2/4 + d)$.*

Proof of Part 1 of Theorem 2. We propose a distributed algorithm that constructs an MIS I of G such that the following two properties hold with high probability: (a) I has average degree at most $O(d^2)$, and (b) I is constructed within $O(\log n)$ rounds.

Our algorithm is based on Luby's algorithm for constructing an MIS, in which vertices v are marked independently with probabiltiy $\frac{1}{2d_v}$. However, if we apply Luby's algorithm directly, it is possible to select high-degree vertices early which invalidate many neighboring low-degree vertices. To remedy this, we will begin by marking the low-degree vertices, and gradually increase the degree of the vertices we allow to enter the MIS. This gives low-degree vertices a head-start compared to high-degree vertices.

We introduce the following definition which will be used throughout the proof. For any real number s, we let G_s denote the subgraph of G induced on the vertices of degree $\leqslant s$. This notation is used in describing Algorithm 1.

```
Repeat for rounds $i = 1, 2, 3, \ldots$:
    Phase I – Selecting vertices from $G_{2^i} - G_{2^{i-1}}$:
    Each vertex $v$ in $G_{2^i} - G_{2^{i-1}}$ marks itself independently with probability $1/(2d_v)$.
    If two adjacent nodes are marked, unmark the one with higher degree (breaking ties arbitrarily).
    Add any marked nodes to the independent set $I$. Remove their neighbors from $G$.
    Phase II – One round of Luby's algorithm on $G_{2^i}$:
    Each vertex $v$ in $G_{2^i}$ marks itself independently with probability $1/(2d'_v)$, where $d'_v$ represents the degree of
        vertex $v$ with respect to the residual graph $G_{2^i} - I - N(I)$.
    If two adjacent nodes are marked, unmark the one with higher degree $d'$ (breaking ties arbitrarily).
    Add any marked nodes to the independent set $I$. Remove their neighbors from $G$.
```

Algorithm 1: Distributed Algorithm for Approximating SMIS.

Here, i iterates over the natural numbers; but we will show that if the algorithm were terminated at $i = \Theta(\log n)$ then it successfully finds the MIS. We refer to each iteration of i as a *round.* Each round of this algorithm applies a single iteration of the Luby algorithm, respectively selecting vertices from $G_{2^i} - G_{2^{i-1}}$ and G_{2^i}. We refer to these as Phase I and Phase II respectively. In Phase I, the sampling probabilities are based on the degrees with respect to the original graph while in Phase II the sampling probabilities are based on the degrees in the residual graph (which may be smaller).

The following basic principle will be used in a variety of places in this proof:

Proposition 1. *Suppose a graph G has n vertices and average degree d. Suppose $s > 1$. Then the subgraph G_{sd} contains at least $n(1 - 1/s)$ vertices.*

Proof. Note that $\sum_v d_v = nd$. Suppose that G_{sd} has fewer than $n(1-1/s)$ vertices. Then there are more than n/s vertices with degree larger than sd. They contribute more than $sd \cdot n/s$ to the sum $\sum_v d_v$, which is a contradiction. □

Lemma 2. *Suppose $d \leqslant \sqrt{n}$. Let i be minimal such that $|V(G_{2^i})|/2^i \geqslant 0.1(n/d)$. Then with high probability, the independent set after Phase I of round i (i.e. selecting vertices in $G_{2^i} - G_{2^{i-1}}$) contains at least $0.01(n/d)$ vertices.*

Proof. Let $n' = |V(G_{2^{i-1}})|$ and let I' be the independent set produced after round $i-1$. By minimality of i, we may assume $n'/2^{i-1} \leqslant 0.1(n/d)$. We may also suppose $|I'| \leqslant 0.01(n/d)$ as otherwise we would be done.

Let S denote the set of vertices eligible to be selected in round i. These are all the vertices in $G_{2^i} - G_{2^{i-1}} - N(I')$. We thus have

$$\begin{aligned}
|S| &= |V(G_{2^i} - G_{2^{i-1}}) - N(I')| \\
&\geqslant |V(G_{2^i})| - |V(G_{2^{i-1}})| - |N(I')| \\
&\geqslant |V(G_{2^i})| - n' - |I'|2^{i-1} \qquad \text{as } I' \subseteq G_{2^{i-1}} \\
&\geqslant 0.1(n/d)2^i - (0.1)2^{i-1}(n/d) - 0.01(n/d)2^{i-1} \\
&\geqslant 0.04(n/d)2^i
\end{aligned}$$

So we are applying a single round of the Luby algorithm to the set S. We now analyze the behavior of that algorithm. For each vertex $v \in S$, let X_v denote that vertex v is marked; these are independent Bernoulli random variables with probability $\frac{1}{2d_v}$. Let $Y = \sum_{v \in S} X_v - \sum_{\substack{u,v \in S \\ \langle u,v \rangle \in E}} X_u X_v$. Clearly Y is a lower-bound on the number of vertices selected. Note that Y is a polynomial in the underlying independent variables X.

Let $Z = \sum_{v \in S} \frac{1}{d_v}$. Note that $d_S \leqslant 2^i$ and $Z \geqslant |S|/d_S \geqslant 0.04(n/d) \geqslant \Omega(\sqrt{n})$. We first may calculate the mean of Y:

$$\mathbf{E}[Y] \geqslant \sum_{v \in S} \frac{1}{2d_v} - \sum_{\substack{v \in S, u \in N(v) \cap S \\ d(u) > d(v)}} \frac{1}{4 d_v d_u} \geqslant \sum_{v \in S} \frac{1}{2d_v} = Z/2$$

Thus, if Y were equal to its mean value, then in round i we would select $0.04(n/d)$ vertices for the independent set, thus achieving the induction claim. So we need to show that Y is close to its mean value. To do so, we use the moment inequality of Schudy & Sviridenko [15]. To apply this inequality, we must calculate μ_0 and μ_1, which are almost the same as the maximum partial derivatives of the polynomial Y in terms of the underlying variables X_v. These are given by

$$\mu_1 = \max_{v \in S} \mathbf{E}[\sum_{u \in N(v) \cap S} X_u] \leqslant \sum_{u \in N(v) \cap S} \frac{1}{2d_u} \leqslant 2^i \frac{1}{2(2^{i-1})} \leqslant 1$$

Similarly, we have for μ_0:

$$\mu_0 = \sum_{u \in S} \frac{1}{2d_u} + \sum_{\substack{u,v \in S \\ \langle u,v \rangle \in E}} \frac{1}{4 d_u d_v} \leqslant Z/2 + \sum_{u \in S} \frac{1}{2d_u} \sum_{v \in N(v) \cap S} \frac{1}{2d_v} \leqslant Z$$

We may now apply the inequality of Schudy & Sviridenko, setting $\lambda = Z/4$:

$$\begin{aligned}
P(Y \leqslant Z/4) &\leqslant P(|Y - \mathbf{E}[Y]| \geqslant \lambda) \\
&\leqslant e^2 \max(e^{-\frac{\lambda^2}{\mu_0 \mu_1 L R}}, e^{-(\frac{\lambda}{\mu_1 L R})^{1/2}}) \\
&\leqslant e^2 \max(e^{-\frac{Z^2/16}{Z L R}}, e^{-(\frac{Z/4}{L R})^{1/2}}) \\
&\qquad \text{for constants } L, R \\
&\leqslant e^{-Z^{\Omega(1)}} \leqslant e^{-n^{\Omega(1)}} \leqslant n^{-\omega(1)}
\end{aligned}$$

Thus, with high probability, we have $Y \geqslant Z/4$. So in round i, we choose an independent set containing $Z/4 \geqslant 0.01(n/d)$ vertices as desired.

Proposition 2. *Suppose G has average degree d. Then with high probability the independent set created by this algorithm after $O(\log n)$ rounds is an MIS with average degree $O(d^2)$.*

Proof. First, note that when $i \geqslant \log_2 n$, then the Phase I of this algorithm does nothing, while Phase II executes a single round of Luby's algorithm on the residual graph. Thus, after $\log_2 n + O(\log n)$ rounds, this algorithm produces an MIS with high probability.

If $d \geqslant \sqrt{n}$, then this MIS trivially has average degree $\leqslant d^2$.

Suppose $d \leqslant \sqrt{n}$. We claim that there is some i with the property that $|V(G_{2^i})|/2^i \geqslant 0.1(n/d)$. For, consider setting $i = \lfloor \log_2 2d \rfloor$. Then by Proposition 1, $|V(G_{2^i})| \geqslant n/2$. So we have:

$$\begin{aligned} |V(G_{2^i})|/2^i &\geqslant \frac{n/2}{2^{\log_2 2d}} \\ &\geqslant \frac{n/2}{2d} \\ &\geqslant 0.25(n/d) \end{aligned}$$

Thus, let i be minimal such that $V(G_{2^i})/2^i \geqslant 0.1(n/d)$. By Lemma 2, the independent set after round i contains $0.01(n/d)$ vertices with high probability. Thus, the final MIS produced by this algorithm contains at least $0.01(n/d)$ vertices. As the full graph G contains nd edges, the final MIS must contain $\leqslant nd$ edges, and hence has average degree $\leqslant \frac{nd}{0.01n/d} \leqslant O(d^2)$ as desired. □

2.3 The greedy algorithm for SMIS

The greedy algorithm for SMIS is very simple. We label the vertices in order of increasing degree (breaking ties arbitrarily). Each vertex is added to the independent set I (initially, $I = \emptyset$), unless it was adjacent to an earlier vertex already selected.

In this section, we show that this greedy algorithm gives a good distributed algorithms, thus proving Part 2 of Theorem 2. The greedy algorithm is also a fast sequential algorithm requiring time $O(m)$, which is slightly faster than the algorithm of Theorem 2.1.

Theorem 6. *The greedy algorithm produces an MIS of degree at most $\frac{d^2}{4} + d$. (As we have seen in Theorem 1, this is within a factor of 2 of the lowest degree possible.)*

Proof. Order the vertices in order of increasing degree $d_1 \leqslant d_2 \leqslant \ldots \leqslant d_n$. Define the indicator variable x_v to be 1 if $v \in I$ and 0 otherwise, where I is the MIS produced. For any pair of vertices u and v with $d_u \geqslant d_v$, we also define the indicator y_{vu} to be 1 if $v \in I$ and there is an edge from v to u. (It may seem strange to include the variable y_{vv}, as we always have $y_{vv} = 0$ in the intended solution, but this will be crucial in our proof, which is based on LP relaxation.)

As the greedy algorithm selects v iff no earlier vertex was adjacent to it, we have $x_v = 1$ if and only if $y_{1v} = y_{2v} = \cdots = y_{v-1,v} = 0$. In particular, x_v satisfies the linear constraint $x_v \geqslant 1 - y_{1v} - y_{2v} - \cdots - y_{vv}$. The variables x, y also clearly satisfy the linear constraints $\forall v\colon 0 \leqslant x_v \leqslant 1$, $\forall v \leqslant u\colon 0 \leqslant y_{vu}$, and $\forall v\colon \sum_u y_{vu} \leqslant d_v x_v$ which we refer to as the *core constraints.* The final MIS contains $\sum x_v$ vertices and $\sum_v d_v x_v$ edges, and hence the average degree of the resulting MIS is $d_I = \sum_v d_v x_v / \sum_v x_v$.

We wish to find an upper bound on the ratio $R = \frac{\sum_v d_v x_v}{\sum_v x_v}$. The variables x, y satisfy many other linear and non-linear constraints, and in particular are forced to be integral. However, we will show that the core constraints are sufficient to bound R. The way we will prove this is to explicitly construct a solution x, y which satisfies the core constraints and maximizes R subject to them, and then show that the resulting x, y still satisfies $R \leqslant \frac{d^2}{4} + d$.

Let x, y be real vectors which maximizes R among all real vectors satisfying the core constraints, and among all such vectors, which minimize $\sum_{u>v} y_{vu}$ (the vertices have been sorted in order of degree, so $u > v$ here means that u comes after v in the ordering). Suppose $y_{vu} > 0$ for some $u > v$. If $x_u = 1$, then we simply decrement y_{vu} by ϵ. The constraint $x_u \geqslant 1 - y_{1u} - \dots - y_{uu}$ clearly remains satisfied as $x_u = 1$, and all other constraints are unaffected. The objective function is also unchanged. However, this reduces $\sum_{u>v} y_{vu}$, contradicting maximality of x, y.

Suppose $y_{vu} > 0$ for some $u > v$, and $x_u < 1$ strictly. Note that $y_{vu} \leqslant d_v x_v$, so we must have $x_v > 0$ strictly. For some sufficiently small ϵ, we change x, y as follows: $y'_{vu} = y_{vu} - \epsilon$, $y'_{vv} = y_{vv} + \frac{\epsilon}{d_v+1}$, $x'_v = x_v - \frac{\epsilon}{d_v+1}$, $x'_u = x_u + \frac{\epsilon}{d_u+1}$, and $y'_{uu} = y_{uu} + \frac{\epsilon d_u}{d_u+1}$. All other values remain unchanged. We claim that the constraints on x, y are still preserved. Furthermore, the numerator of R does not decrease and the denominator does not increase; hence $R' \geqslant R$. However, $\sum_{u>v} y'_{vu} < \sum_{u>v} y_{vu}$ strictly. This contradicts the maximality of x, y.

In summary, we can assume $y_{vu} = 0$ for all $u > v$. In this case, the core constraints on v become simply $1 - y_{vv} \leqslant x_v \leqslant 1$ and $y_{vv} \leqslant d_v x_v$.

It is a simple exercise to maximize R subject to these constraints (every vertex operates completely independently). The maximum is achieved by a solution which has the form, for some $t > 0$, of $x_v = \frac{1}{d_v+1}$ for $d_v \leqslant t$, and $x_v = 1$ for $d_v > t$. In this case, the objective function $R(x)$ satisfies

$$R \leqslant \frac{\sum_{d_v \leqslant t} \frac{d_v}{d_v+1} + \sum_{d_v > t} d_v}{\sum_{d_v \leqslant t} \frac{1}{d_v+1} + \sum_{d_v > t} 1}$$

Let S, B denote respectively the vertices of degree $\leqslant t, > t$. Then by concavity, we have

$$R \leqslant \frac{|S| \frac{d_S}{d_S+1} + |B| d_B}{\frac{|S|}{d_S+1} + |B|} \leqslant \frac{d(d_B - d_S) + d_B d_S (d - d_S)}{d_S(d - d_S) + (d_B - d_S)}$$

Routine calculus shows that this achieves its maximum value at $d_B = \infty$ and $d_S = d/2$, yielding $R \leqslant d^2/4 + d$ as claimed. □

This greedy algorithm can be converted, with only a little loss, to a parallel algorithm as shown in Algorithm 2.

```
Let $\phi > 1$ be a fixed parameter. Initialize $I = \emptyset$.
for $i = 0, \dots, \lceil \log_\phi n \rceil$ do
    Using any MIS algorithm, extend $I$ to an MIS of the graph $G_{\phi^i}$.
Return the final MIS $I$.
```

Algorithm 2: Greedy Distributed Approximation Algorithm for SMIS.

This is basically the greedy algorithm, except we are quantizing the degrees to multiples of some parameter ϕ. This does not immediately lead to a distributed algorithm, because we are assuming the existence of a "subroutine" implementing an MIS algorithm; this would require knowledge of the network size n. However, it is fairly easy to convert Algorithm 2 to work in the distributed setting. In parallel, we can run rounds of Luby's algorithm on the graph $G_{\phi^i} - G_{\phi^{i-1}}$, which are producing independent sets I_i. Whenever we add v to I_i, we can remove all of the vertices neighboring v from I_j for $j > i$. Thus, after $\log n$ rounds, I_1 converges to an MIS of G_{ϕ^1}; after $2 \log n$ rounds, I_2

converges to an MIS to $G_{\phi^2} - N(I_1)$; and so forth. After $O(\log_\phi^2 n)$ rounds, we have an MIS of the full graph.

For any constant $\epsilon > 0$, we can choose ϕ to be a sufficiently small constant so that this algorithm requires $O(\log^2 n)$ rounds and returns an MIS of average degree at most $(1+\epsilon)(d^2/4 + d)$.

The following observation shows that the above analysis of the greedy algorithm is essentially tight.

Observation 3 *For all real numbers $\alpha > 0$, there are graphs of average degree $\leqslant \alpha$, and for which the greedy algorithm produces an MIS of average degree at least $\alpha^2/4 + \alpha - 1$.*

Proof. Define the following graph G which contains three groups of vertices A, B, C. We have $|A| = a$, $|B| = b - 1$, and $|C| = ab$. Each A-vertex connects to b C-vertices. Each C-vertex connects to all $b - 1$ B-vertices.

The graph G contains $a + ab + b - 1$ vertices and ab^2 edges, and has an average degree of $d = \frac{2ab^2}{a+ab+b-1}$.

Now, the vertices in A, C have degree b, while the vertices in B have degree ab. Suppose that the greedy algorithm selects the A-vertices (they are tied with the C-vertices). It then selects all B-vertices, and hence, the resulting MIS has $a + b - 1$ vertices and ab^2 edges. Now set $b = \left\lfloor \frac{(a+1)\alpha + \sqrt{(a+1)^2\alpha^2 + 8(a-1)a\alpha}}{4a} \right\rfloor$. As a tends to infinity, we have $d \leqslant \alpha$, while the degree of the resulting MIS approaches a value that is at least $\alpha^2/4 + \alpha - 1$. □

2.4 NP-Completeness of the Decision Version of SMIS

We show NP-completeness of the decision version of SMIS (cf. Sec. 1.1) by reducing a variant of the 3-SAT problem to the SMIS problem and vice versa.

A Boolean formula in conjunctive normal form is called a (k, s)*-formula* if every clause contains exactly k distinct variables and every variable occurs in at most s clauses. A (k, s)-formula is called a $(k, =s)$-formula if every variable occurs in exactly s clauses. Let (k, s)-SAT (resp., $(k, =s)$-SAT) denote the satisfiability problem restricted to (k, s)-formula (resp., $(k, =s)$-formula). Kratochvil et al. [8] proved that the $(k, =s)$-SAT problem is NP-complete for every $k \geqslant 3$ and $s \geqslant 4$. We now show how to reduce an instance of the $(3, =4)$-SAT problem to an instance of SMIS.

In the $(3, =4)$-SAT problem, the input is a set of clauses, each of them with 3 variables and each variable occurs in exactly 4 clauses and the aim is to find a satisfying truth assignment to the whole $(3, =4)$-formula. To form a satisfying truth assignment we must pick one literal from each clause and give it the value TRUE. But our choices must be consistent, namely, if we choose a variable x in one clause, we cannot choose the negation of x in another. Any consistent choice of literals, one from each clause, specifies a truth assignment. Recall that for the SMIS problem, we are given a graph G and we want to know whether G contains MIS with average degree at most that of the graph. We relate the above two problems as follows.

SMIS Graph Construction Given a $(3, =4)$-formula $F = A_1 \wedge A_2 \wedge \cdots \wedge A_k$, we construct the following graph G_F, which we simply denote as G if F is clear from the context. For each clause, say $A_i = (x \vee y \vee z)$, in F, we construct the following corresponding *clause component*. Construct a triangle with vertices labeled x, y, and z, and for each *triangle vertex*, add an additional neighbor labeled the negation of the corresponding vertex, and for each of these negations, add another

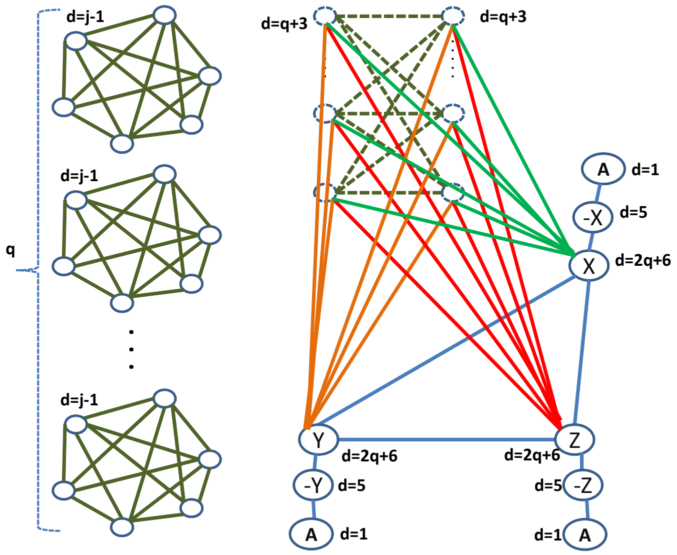

Fig. 1. Graph construction corresponding to clause $(x \vee y \vee z)$.

neighbor (of degree 1). Construct a single copy of $K_{q,q}$ and q copies of K_j, and connect each of the triangle vertices with all vertices in $K_{q,q}$ (see Fig. 1).

Finally, we connect each labeled vertex in each clause component with its negation vertex in all other clause components, and hence, any MIS in G cannot contain vertices labeled both x and $-x$ for all variables x in F. We will see in the proof of Theorem 7 how to obtain a mapping between truth assignments of the literals in F and maximal independent sets on G_F.

Clearly, this construction takes polynomial time and each of the triangle vertices has degree $2q + 6$, whereas each of the negation vertices has degree 5, and each neighbor of these negation vertices has degree 1.

Lemma 3. *Let F be a $(3, {=}4)$-formula and consider the corresponding graph G with average degree d. Consider any MIS S in G. Then $d_S \leqslant d$ iff S contains one triangle vertex from every clause component of G.*

Proof. The average degree of one component in G is $(qj(j-1) + 2q(q+3) + 3(2q+6) + 18)/(qj + 2q + 3 + 6)$. The average degree of an MIS (in one component) containing one of the triangle vertices and the three degree 1 vertices is $(q(j-1) + (2q+6) + 3)/(q+1+3)$. On the other hand, the average degree of any MIS that does not contain one of the triangle vertices is at least $(q(j-1) + q(q+3) + 3)/(q+q+3)$. Therefore, the ratio of the average degree of any MIS in one component that does not contain one of the triangle vertices to the average degree of this component tends to $(j+2)/4$ as q tends to infinity.

Now consider the whole graph G. Clearly, the average degree of G is the average degree of each component scaled appropriately by the number of clauses in F, denoted by k. Now, assume that i of the k clauses are not satisfiable (hence $k - i$ clauses are satisfiable). Then, the average degree of any MIS in G is at least

$$\frac{(k-i)(q(j-1) + (2q+6) + 3) + i(q(j-1) + q(q+3) + 3)}{(k-i)(q+1+3) + i(q+q+3)}.$$

Therefore, the ratio of the average degree of any MIS in G to the average degree of G tends to $i(j+2)/(2(k+i))$ as q tends to infinity. That is, by choosing j large enough with respect to k, the above ratio will be greater than 1.

Theorem 7. *The following decision problem is NP-complete: Given a graph G, is there an MIS in G with average degree at most that of the graph?*

Proof. We proceed by giving reductions from $(3,=4)$-SAT to SMIS and vice versa.
$(3,=4)$-SAT → SMIS: We need to show that if a $(3,=4)$-formula F has a satisfying assignment, then there exists an MIS S in G with $d_S \leqslant d$.

Assume that the $(3,=4)$-formula F has a satisfying assignment. For each clause in F, we pick any literal whose value under a satisfying assignment is **TRUE** (there must be at least one such literal), and add the corresponding vertex in the component corresponding to that clause to S. We then add, from each component, all degree one vertices and one vertex from each complete graph K_q to S. Clearly, S yields an independent set that is maximal. Moreover, since S contains one of the triangle vertices in each component, Lemma 3 tells us that the average degree of S is at most that of G.
SMIS → $(3,=4)$-SAT: We want to show that if there exists an MIS S in G with an average degree of at most that of the graph G, then the corresponding $(3,=4)$-formula F has a satisfying assignment.

By Lemma 3, if S has $d_S \leqslant d$, then S includes, for each clause component of G, one of the triangle vertices, all degree one vertices, and one vertex from each complete graph K_q.

To obtain a satisfying assignment of F, we assign x a value of **TRUE** if S contains a vertex labeled x, and a value of **FALSE** if S contains a vertex labeled $-x$ (if S contains neither, we assign an arbitrary truth value to x). Clearly this yields get a truth assignment that satisfies all clauses in F.

2.5 Hardness of Approximating SMIS

Theorem 8. *For every $\epsilon > 0$, the SMIS problem is hard to approximate within a factor of $\frac{1}{8}n^{\frac{1}{2}-\epsilon}$ in polynomial time, unless NP=ZPP.*

Proof. Given a graph H with $p = \sqrt{n}$ vertices[4], we construct a graph G with n vertices as follows. We first construct a complete graph K_{p^2-p}, and then attach H to K_{p^2-p} by connecting each vertex in H with $p^{1+\epsilon} - p^{\epsilon}$ vertices in K_{p^2-p} such that each vertex in K_{p^2-p} has p^{ϵ} neighbors in H. Assume that the size of the maximum independent set of H is at least $p^{1-\epsilon}$, where $\epsilon = O(1/\sqrt{\log\log p})$ (see for example the graph in [3]). It is known that the maximum independent set problem on graphs with p vertices cannot be approximated within $p^{1-\epsilon}$ in polynomial time (unless NP = ZPP) [3]. This implies that there is no approximation algorithm that guarantees an approximated solution for the maximum independent set problem on H of more than p^{ϵ} vertices (since the size of the maximum independent set is at most p). We now convert the graph H to a p-regular graph by adding an appropriate number of self loops to each vertex in H. We observe that the above inapproximability result on the maximum independent set problem on H still holds for the resulting p-regular graph H. Thus, each vertex of the graph H has degree $p^{1+\epsilon} - p^{\epsilon} + p$, and each vertex in K_{p^2-p} has degree $p^2 - p - 1 + p^{\epsilon}$. Clearly, the optimal solution for SMIS on G includes the

[4] To simplify the proof, we will ignore rounding issues; the errors they introduce dimnish asymptotically as n grows.

```
Mark each vertex of degree $> 2d$ independently with prob. $\frac{\ln t}{t}$ where $t = \frac{2d \ln d}{\ln \ln d}$.
Mark every vertex of degree $\leqslant 2d$.
If any vertex $v$ is not marked, and none of the neighbors of $v$ are marked, then mark $v$.
Let $M$ denote the set of marked vertices at this point. $M$ forms a dominating set of $G$, but is not necessarily
  minimal. Using any algorithm, select a minimal dominating set $M' \subseteq M$.
Check if $d_{M'} \leqslant t$. If so, return $M'$. Otherwise, return FAIL.
```

Algorithm 3: Approximation Algorithm for SMDS.

maximum independent set on H, and has average degree $p^{1+\epsilon} - p^{\epsilon} + p$ (the degree of each vertex in H). On the other hand, the average degree of the approximated polynomial time solution of SMIS is at least $(p^{\epsilon}(p^{1+\epsilon} - p^{\epsilon} + p) + (p^2 - p - 1 + p^{\epsilon}))/(p^{\epsilon} + 1)$. Thus the approximated SMIS is at least $(p^2 - p + p^{\epsilon} - 1)/((p^{\epsilon} + 1)(p^{1+\epsilon} - p^{\epsilon} + p))$ times the optimal. Note that, $p^{\epsilon} - 1 \geqslant 0$, $p^2 - p \geqslant p^2/2$, and $(p^{\epsilon} + 1)(p^{1+\epsilon} - p^{\epsilon} + p) \leqslant 4p^{1+2\epsilon}$ hold. Therefore, the approximated SMIS is at least $p^{1-2\epsilon}/8$ times the optimal, which proves the theorem.

3 Sparse Minimal Dominating Set (SMDS)

For arbitrary graphs, we turn our attention to designing algorithms for finding approximate solutions to SMDS. Since any MIS in a given graph G is also an MDS in G, all algorithms designed for SMIS also return an SMDS in G of the same average degree. Thus, we have the same bounds (and distributed algorithms) corresponding to those in Section 2. However, for SMDS, better bounds are possible. Given a graph with average degree d, we will present a polynomial-time algorithm that finds an MDS of average degree $O(\frac{d \log d}{\log \log d})$. We will also construct a family of graphs G for which every MDS has average degree $\Omega(\frac{d \log d}{\log \log d})$.

Theorem 4. *Any graph G with average degree d has a minimal dominating set with average degree at most $O(\frac{d \log d}{\log \log d})$. Furthermore, there is a sequential deterministic algorithm to find such an MDS in time $O(m)$.*

Proof. For a target degree t, and any set of vertices V_0, we define $\Psi_t(V_0) = \sum_{v \in V_0} (d_v - t)$. Our goal is to find an MDS X with $\Psi_t(X) \leqslant 0$, for some $t = O(\frac{d \log d}{\log \log d})$.

Let $x = 2d$ and divide the vertices into three classes: A, the set of vertices of degree $\leqslant x$; B, the set of vertices of degree $> x$, which have at least one neighbor in A; and C, the set of vertices of degree $> x$, all of whose neighbors are in B or C. Let $D = B \cup C$. Mark each vertex in D with probability $p = \frac{\ln t}{t}$. Next, define the set $Y \subseteq D$ consisting of all marked vertices in D and vertices in C with no marked neighbors. Clearly Y dominates C, and $A \cup Y$ dominates G. We finish by producing an MDS $X \subseteq A \cup Y$; we may write $X = A' \cup Y'$ where $A' \subseteq A$ and $Y' \subseteq Y$.

We first examine $\Psi_t(Y')$. Any vertex of G with degree $\leqslant t$ contributes at most 0 to $\Psi_t(Y')$. Therefore, suppose v has degree $> t$. If $v \in B$, it is selected for Y with probability at most $\frac{\ln t}{t}$. If $v \in C$, all its neighbors are marked with probability $\frac{\ln t}{t}$, so it is selected for Y with probability at most $\frac{\ln t}{t} + (1 - \frac{\ln t}{t})^t \leqslant 2\frac{\ln t}{t}$. Hence the expected contribution of such vertex to $\Psi_t(Y')$ is at most $2\frac{\ln t}{t}(d_v - t) \leqslant 2d_v\frac{\ln t}{t}$. Summing over all such vertices, we have $E[\Psi_t(Y')] \leqslant 2|D|d_D\frac{\ln t}{t}$,

Now, some of the vertices in A are dominated by B-vertices of Y'. Let A_0 be the set of vertices *not dominated* by Y'. These vertices can only be dominated by vertices of A', so we must have

$|A'|(d_{A'}+1) \geqslant |A_0|$. Thus, we have

$$\Psi_t(A') = \sum_{v \in A'} d_v - t = |A'|(d_{A'} - t) \leqslant |A_0| \frac{d_{A'} - t}{d_{A'} + 1} \leqslant |A_0| \frac{x-t}{x+1}$$

Consider the expected size of A_0. A vertex $v \in A$ lies in A_0 if none of its neighbors are marked (this is not a necessary condition), and vertices are marked independently with probability p. Hence $E[|A_0|] \geqslant \sum_{v \in A}(1-p)^{d_v} \geqslant |A|(1-p)^{d_A}$.

Putting all this together, we have that the final MDS $X = A' \cup Y'$ satisfies $E[\Psi_t(X)] \leqslant 2p|D|d_D + |A|\frac{x-t}{x+1}(1-p)^{d_A}$. For d sufficiently large, p approaches zero, so $(1-p)^{d_A} \leqslant e^{-2pd_A}$.

We know that $|A| + |D| = n$, and $|A|d_A + |D|d_D = nd$. Eliminating $|A|, |D|$ we have

$$E[\Psi_t(X)] \leqslant n\Big(\frac{2d_D(d-d_A)\ln t}{t(d_D - d_A)} - \frac{(d_D - d)(t-2d)t^{-2\frac{d_A}{t}}}{(2d+1)(d_D - d_A)}\Big)$$

Routine calculus shows that, for t sufficiently large, this achieves its maximum value at $d_D \to \infty$ and $d_A = \frac{t \ln\left(\frac{t-2d}{2d+1}\right)}{\ln t}$, yielding

$$\mathbf{E}[\Psi_t(X)] \leqslant \frac{2d \ln t}{t} - 2\ln\left(\frac{t-2d}{2d+1}\right) - 1.$$

For $t = \frac{3d \ln d}{\ln \ln d}$, the RHS approaches $-\infty$ as $d \to \infty$. This implies that $\mathbf{E}[\Psi_t(X)] \leqslant 0$, so there is a positive probability of selecting an MDS of average degree $\leqslant t$.

This is summarized as Algorithm 3.

This process can be derandomized using the method of conditional expectations. The underlying random variables here are the marking vectors Z for the vertices of D. Given a marking vector Z, we can define the pessimistic estimator

$$\Psi'(Z) = \sum_{v \in D} Z_v(d_v - t) + \sum_{v \in C}\Big[\bigwedge_{w \in N(v) \cup \{v\}} Z_w = 0\Big](d_v - t) + \sum_{v \in A}\Big[\bigwedge_{w \in N(v)} Z_w = 0\Big]\frac{x-t}{x+1}$$

We have already shown that each vector $v \in D$ is marked independently with probability p, then $\mathbf{E}[\Psi'] \leqslant 0$ and furthermore that if $\Psi' \leqslant 0$ then we are guaranteed that the final MDS has degree $O(\frac{d \log d}{\log \log d})$. For any vertex $v \in D$, it is not hard to compute the change in $\mathbf{E}[\Psi']$ when we set $Z_v = 0$ or $Z_1 = 1$ deterministically; this only depends on the neighbors of v, so it requires time $O(d_v)$. Thus, we can determine in time $O(m)$ a marking vector Z which guarantees $\Psi' \leqslant 0$. This can be easily extended to an MDS in time $O(m)$.

We next prove Theorem 5, which shows that this bound $O(\frac{d \log d}{\log \log d})$ is optimal up to constant factors.

Theorem 5. *For any real number $\alpha > 0$, there are graphs with average degree $\leqslant \alpha$, but for which any MDS has an average degree of $\Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$.*

Proof. We will construct a graph of average degree $d = O(\alpha)$, all of whose MDS's have degree $\Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$. To simplify the proof, we will ignore rounding issues. As all the quantities tend to infinity with α, such rounding issues are negligible for α sufficiently large.

Define $k = \log_2(\alpha \ln \alpha / \ln \ln \alpha)$. We define a random process which constructs a graph with three types of vertices, which we denote A, B, C (these play the same role as in the proof of Theorem 4).

The vertices in A, B are organized into clusters of related vertices. For class A, there are $l = (\frac{\ln \alpha}{\ln \ln \alpha})^2$ clusters of size α. For class B, there are r clusters of size $\frac{\alpha \ln \alpha}{r \ln \ln \alpha}$, for some $r = \Theta(k)$ (the constant will be specified later).

There are $2^k - 1$ vertices in class C. These are not organized into clusters but are considered individually. We index these vertices by the non-zero k-dimensional binary vectors over the finite field $GF(2)$. That is, C corresponds to $C = GF(2)^k - \mathbf{0}$.

We add the following edges to the graph (some of these edges are deterministic, some are random):

1. From each A-vertex to the other vertices in the same A-cluster.
2. From each B-vertex to all the other B-vertices, even those outside its cluster.
3. For each B-cluster b, we choose a random non-zero binary vector v_b in $GF(2)^k$. For each vertex in C, indexed by vector w, we construct an edge from all the B-vertices in the cluster b to the vertex w iff $v_b \bullet w = 1$. The dot product here is taken over the field $GF(2)$.
4. For each A-cluster a, we select $\frac{r \ln \ln \alpha}{\ln \alpha}$ of the B-clusters uniformly at random, with replacement.

We add an edge from every vertex in the A-cluster a to every vertex in the selected B-clusters. This graph has degree $O(\alpha)$. The following lemmas characterize the behavior of this graph and its minimal dominating sets:

Lemma 4. *Any MDS of G contains at most one vertex from each B-cluster. If the MDS contains i such B-vertices, then it contains at least $2^{k-i} - 1$ C-vertices.*

Proof. Let $v_1, \dots, v_i$ be the binary vectors associated with the selected B-vertices. Then the set of k-dimensionsal binary vectors which are perpendicular to $v_1, \dots, v_i$ has dimension at least $k - i$. So there are at least $2^{k-i} - 1$ non-zero vectors perpendicular to $v_1, \dots, v_i$. The corresponding C-vectors have no edges to the selected B-vertices, and no edges to any other type of vertex. In order for them to be dominated, they must themselves be part of the MDS. □

Lemma 5. *It is possible to select the parameter $r = \Theta(k)$ such that, with high probability, all C vertices have degree $\Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$.*

Proof. Let us fix a particular C-vertex, associated to binary vector $w \in GF(2)^k$. This vector w is perpendicular to any randomly selected vector $v \neq \mathbf{0}$ with probability $1/2 - 2^{-k}$. Hence the expected number of B-clusters connected to it is $r(1/2 - 2^{-k})$. By Chernoff's bound, the probability that it connects to fewer than $r/4$ clusters is $\exp(-\Omega(r))$.

For r a sufficiently large constant multiple of k, this probability is much less than 2^{-k}. By the union bound, this implies that there is a negligible probability that any C vertex connects to fewer than $r/4$ of the B-clusters. So with high probability, every C-vertex connects to $\Omega(r)$ B-clusters. As every B-cluster has $\frac{\alpha \ln \alpha}{r \ln \ln \alpha}$ vertices, this implies that the C-vertices have degree $\Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$. □

Lemma 6. *With high probability, the graph G satisfies the following property: For all sets X of B-clusters where X contains at least $(3/4)k$ distinct B-clusters, all but $O(\log \alpha)$ of the A-clusters are connected to some vertex in X.*

Proof. Suppose we fix a set X which contains $i \geqslant 3/4k$ distinct B-clusters. Any given A-cluster connects to $\alpha/(2^k/r)$ B-clusters chosen uniformly a random, so the probability that this A-cluster is disjoint to X is at most $(1 - i/r)^{\alpha/(2^k/r)} \leqslant \exp(-\alpha i/2^k)$. Hence the expected number of such A-clusters is at most $l \exp(-\alpha i/2^k)$. For $i \geqslant 3/4k$, this is $o(\log \alpha)$. Hence by Chernoff's bound, the probability that the number of disconnected A-clusters exceeds $\phi \ln \alpha$ is at most $\exp(-\phi' \ln \alpha)$, where ϕ' increases with ϕ.

The total number of such sets X is at most $2^r = \exp(O(k))$. So, by the union-bound, the probability that any such X has this event occuring is at most $\exp(O(k) - \phi' \ln \alpha)$. For ϕ a sufficiently large constant, this probability is negligible. □

Lemma 7. *Suppose the graph G has all the properties of Lemmas 4, 5, 6. Then every MDS of G has degree $\Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$.*

Proof. Suppose we have an MDS of G which contains i distinct B-clusters. There are two cases. First, suppose $i \leqslant 3/4k$. In this case, the MDS contains at least $2^{k-i} \geqslant 2^{k/4}$ C-clusters. The MDS contains at most one vertex from each of the A-clusters. Hence, the average degree of the A, C vertices in this MDS is $\Omega(\frac{l\alpha + 2^{k/4} 2^k}{l + 2^{k/4}}) = \Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$. As any B-vertex has also degree $\Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$, this implies that the total average degree of the MDS is $\Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$.

Next, suppose $i \geqslant 3/4k$. The total number of $B + C$ vertices is at least $i + 2^{k-i} = \Omega(k)$. All but $O(\log \alpha)$ of the A-clusters are already dominated by B vertices; these are the only A-clusters which can join the MDS. As a vertex in A cluster connects to only the other vertices in that same cluster, the total number of A-vertices in the MDS is at most $O(\log \alpha)$. Hence the degree of the MDS is at least $\Omega(\frac{\alpha \log \alpha + k 2^k}{\log \alpha + k}) = \Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$. This completes the proof of Theorem 5. □

With high probability, every MDS of G has degree $\Omega(\frac{\alpha \log \alpha}{\log \log \alpha})$.

4 Sparse Minimal Vertex Cover (SMVC)

Observation 2. *For any real number $\alpha > 2$, there are graphs for which the average degree is at most α, but for which the average degree of any MVC is arbitrarily large.*

Proof. The following example shows that the ratio of the average degree of any MVC in the underlying graph to that of the graph itself can become arbitrarily large.

Consider the graph G that contains a single copy H of a complete graph K_p such that each vertex of H is connected to q neighbors, each of them of degree 1. Then we have $d = \frac{p-1+2q}{1+q}$.

On the other hand, any VC in G contains at least $p-1$ vertices from H. In particular, the minimum-average-degree MVC in G contains exactly $p-1$ vertices from H and the q neighbors of the remaining vertex of H, so has average degree at least $\frac{(p-1)(p-1+q)+q}{p-1+q}$. Now, let $p = \lfloor (\alpha - 2) q \rfloor$ and let $p, q \to \infty$. The resulting graphs have $d \leqslant \alpha$, while the MVC has its degree approach ∞.

(Note that if we allow G to contain isolated vertices, then this theorem becomes a triviality: we can simply add arbitarily many isolated vertices to a graph G.)

5 Conclusion

We have initiated the study – graph-theoretic, algorithmic, randomized, and distributed – of the balanced versions of some fundamental graph-theoretic structures. As discussed in Section 1, the study of balanced structures can be useful in providing fault-tolerant, load-balanced MISs and MDSs. We have developed reasonably-close upper and lower bounds for many of these problems. Furthermore, for the SMIS problem, we have presented fast (local) distributed algorithms that achieves an approximation close to the best possible in general; a key problem that is left open is whether one can do the same for the SMDS problem (this has been partially addressed in a subsequent work [10]). We view our results also as a step toward understanding the complexity of local computation of these structures whose optimality itself cannot be verified locally.

Acknowledgements. We would like to thank the anonymous referees for their suggestions that helped to improve the paper.

References

1. N. Alon, L. Babai, and A. Itai. A fast and simple randomized parallel algorithm for the maximal independent set problem. *J. Algorithms*, 7(4):567–583, 1986.
2. N. Alon and J.H. Spencer. *Probabilistic Method.* Wiley, 2008.
3. Lars Engebretsen and Jonas Holmerin. Towards optimal lower bounds for clique and chromatic number. *Theor. Comput. Sci.*, 1-3(299):537–584, 2003.
4. Jeremy T Fineman, Calvin Newport, Micah Sherr, and Tonghe Wang. Fair maximal independent sets. In *Parallel and Distributed Processing Symposium, 2014 IEEE 28th International*, pages 712–721. IEEE, 2014.
5. Pierre Fraigniaud, Amos Korman, and David Peleg. Local distributed decision. In *FOCS*, pages 708–717, 2011.
6. M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.
7. Richard M. Karp and Avi Wigderson. A fast parallel algorithm for the maximal independent set problem. *J. ACM*, 32(4):762–773, 1985.
8. J. Kratochvil, P. Savicky, and Z. Tuza. One more occurrence of variables makes satisfiability jump from trivial to NP-complete. *SICOMP*, pages 203—210, 1993.
9. F. Kuhn, T. Moscibroda, and R. Wattenhofer. Local computation: Lower and upper bounds. *CoRR*, abs/1011.5470, 2010.
10. Shay Kutten, Danupon Nanongkai, Gopal Pandurangan, and Peter Robinson. Distributed symmetry breaking in hypergraphs. In *Distributed Computing - 28th International Symposium, DISC 2014, Austin, TX, USA, October 12-15, 2014. Proceedings*, pages 469–483, 2014.
11. M. Luby. A simple parallel algorithm for the maximal independent set problem. *SIAM J. Comput.*, 15(4):1036–1053, 1986.
12. T. Moscibroda. Clustering. *Book Chapter in Algorithms for Sensor and Ad Hoc Networks*, pages 37–60, 2007.
13. D. Peleg. *Distributed Computing: A Locality-Sensitive Approach.* SIAM, 2000.
14. R. Rajaraman. Topology control and routing in ad hoc networks: a survey. *SIGACT News*, 33(2):60–73, 2002.
15. Warren Schudy and Maxim Sviridenko. Concentration and moment inequalities for polynomials of independent random variables. In *Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012, Kyoto, Japan, January 17-19, 2012*, pages 437–446, 2012.
16. D. Sivakumar. Algorithmic derandomization via complexity theory. In *STOC*, pages 619–626, 2002.
17. J. Spencer. Turan's theorem for k-graphs. *Discrete Mathematics*, 2(2):183–186, 1972.
18. J. Suomela. Survey of local algorithms. *ACM Computing Surveys*, 45(2):24, 2013.
19. P. Turan. On an extremal problem in graph theory. *Mat. Fiz. Lapok*, 48:436–452, 1941.
20. H. Zhang and H. Shen. Balancing energy consumption to maximize network lifetime in data-gathering sensor networks. *IEEE TPDS*, 20(10):1526–1539, 2009.