Distributed Large Independent Sets in One Round On Bounded-independence Graphs*

Magnús M. Halldórsson and Christian Konrad

ICE-TCS, School of Computer Science, Reykjavik University, Reykjavik, Iceland {christiank,mmh}@ru.is

Abstract. We present a randomized one-round, single-bit messages, distributed algorithm for the maximum independent set problem in polynomially bounded-independence graphs with poly-logarithmic approximation factor. Bounded-independence graphs capture various models of wireless networks such as the unit disc graphs model or the quasi unit disc graphs model. For instance, on unit disc graphs, our achieved approximation ratio is $O((\frac{\log n}{\log \log n})^2)$.
A starting point of our work is an extension of Turán's bound for independent sets by Caro and Wei which states that every graph $G = (V, E)$ contains an independent set of size at least $\beta(G) := \sum_{v \in V} \frac{1}{\deg_G(v)+1}$, where $\deg_G(v)$ denotes the degree of v in G. Alon and Spencer's proof of the Caro-Wei bound in [1] suggests a randomized distributed one-round algorithm that outputs an independent set of expected size equal to $\beta(G)$, using messages of sizes $O(\log n)$, where n is the number of vertices of the input graph. To achieve our main result, we show that $\beta(G)$ gives poly-logarithmic approximation ratios for polynomially bounded-independence graphs. Then, for $O(1)$-claw free graphs (which include graphs of bounded-independence), we show that using a different algorithm, an independent set of expected size $\Theta(\beta(G))$ can be computed in one round using single bit messages, thus reducing the communication cost to an absolute minimum.
Last, in general graphs, $\beta(G)$ may only give an $\Omega(n)$-approximation. We show, however, that this is best possible for one-round algorithms: We show that each such distributed algorithm (possibly randomized) has an approximation ratio of $\Omega(n)$ on general graphs.

1 Introduction

Something For Almost Nothing. When designing approximation algorithms, the usual goal is to find desirable trade-offs between approximation guarantee and the resources required by the algorithm, such as computation time, memory consumption, the number of queries to the input, or, in the area of distributed computing, message size and the number of communication rounds. In past years, in various algorithmic disciplines, research has been carried out in order to determine the minimum amount of resources required to achieve non-trivial solutions. Often, it is asked how much effort it takes to obtain at least something from the given problem instance. Examples include property testing algorithms [18] that query a given instance only a few times in order to reason about whether the instance is close to having a certain property or it is far from having this property. In distributed computing, this phenomenon can be observed for example with regards to communication patterns and the total number of communication rounds. It has been shown that non-trivial computation is possible even when the communication pattern of nodes is restricted to beeps [4]. Moreover, research on so-called local algorithms [17, 12] that employ only a few communication rounds has been carried out and highly non-trivial results have been obtained (e.g. even some NP-hard problems can be solved in only a constant number of communication rounds [2]).

In this paper, we ask whether non-trivial computation is possible if we grant a distributed algorithm only a single communication round. Specifically, we ask whether reasonable approximations to the maximum independent set problem can be computed in this harsh setting.

* Supported by Icelandic Research Fund grants 120032011 and 152679-051.

Computational Model. We consider a network of computational units of unbounded computational power V modelled by a graph $G = (V, E)$. The graph G constitutes the input graph of the problem. We assume that vertices have unique IDs. Initially, besides its ID, every node $v \in V$ also knows its degree $\deg_G(v)$. Communication occurs in simultaneous communication rounds along the edges E of G. Then the runtime of a distributed algorithm is the total number of communication rounds. In this work, we mainly focus on algorithms that run in a single communication round. In the $\mathcal{LOCAL}$ model, algorithms may exchange messages of unbounded sizes. In the $\mathcal{CONGEST}$ model, message sizes are restricted to $\mathrm{O}(\log n)$, where n denotes the number of vertices of the input graph.

Independent Sets. An independent set I in a graph $G = (V, E)$ is a subset of non-adjacent vertices. An independent set I is maximal if it is inclusion-wise maximal, i.e., $I \cup \{v\}$ is not an independent set for any $v \in V \setminus I$. A maximum independent set is one of maximal size. The independence number of graph G is the size of a maximum indendent set in G and is denoted by $\alpha(G)$. Computing maximum independent sets is NP-hard on general graphs [10] and is even hard to approximate within factor $n^{1-\epsilon}$ for any $\epsilon > 0$ [21]. The independent set problem is one of the most studied problems in distributed computing, and we detail related work further below.

Our Main Result. Our main result concerns graphs of *polynomially bounded-independence*, a graph class that includes unit disc graphs and similar graph classes that are used for modelling wireless networks (for a precise definition see the next paragraph). We show that in the harsh setting of a single communication round, a poly-logarithmic approximation ratio can be achieved in polynomially bounded-independence graphs. Furthermore, we show that not only the number of communication rounds but also message sizes can be reduced to an absolute minimum, i.e., to single bit messages.

Bounded-independence Graphs. Bounded-independence graphs capture many intersection graphs of geometrical objects which in turn are used for modelling conflict graphs of wireless networks. Given a collection $X = \{X_1, \ldots, X_n\}$ of geometrical objects, the corresponding intersection graph is obtained by assigning X as the vertices of the graph, and an edge is introduced between two vertices X_i, X_j iff the objects X_i and X_j intersect. In the literature, conflict graphs of wireless networks are often modelled by unit disc graphs [7], the intersection graph of discs with equal radii, where the radius of the discs corresponds to the transmission range of the wireless transmitters. Unit disc graphs have many nice properties that allow for the design of efficient distributed algorithms, but the assumption of identical transmission radii for all wireless transmitters is often too restrictive. Consequently, the unit disc graphs model has been extended to more elaborate models such as quasi unit disc graphs [13] or general disc graphs. In a general disc graph, no restriction on the radii of the discs are imposed. Then, the parameter $\delta = r_{\max}/r_{\min}$ is introduced into the analysis of algorithms, where $r_{\max}$ and $r_{\min}$ denote the maximal and the minimal radius of a disc, respectively.

All graphs of the previously mentioned graph classes are of bounded-independence, a property that restricts the size of a maximum independent set within the set of nodes at a given maximal distance from any node. The r-neighborhood of a node v is the set of nodes at distance at most r from v (excluding v).

Definition 1. *A graph $G = (V, E)$ is of* bounded-independence *if there is a bounding function $f(r)$ so that for each node $v \in V$, the size of a maximum independent set in the r-neighborhood of v is at most $f(r), \forall r \geq 1$. We say that G is of* polynomially bounded-independence *if $f(r)$ is a polynomial.*

It is easily verified that unit disc graphs are of bounded-independence with respect to a bounding function in $\mathrm{O}(r^2)$, and disc graphs are of bounded-independence with respect to a bounding function in $\mathrm{O}((r\delta)^2)$. Many important problems such as the maximal independent set problem, or the $(\Delta{+}1)$-coloring problem can be solved on bounded independence graphs by a distributed algorithm by Schneider and Wattenhofer that uses $\mathrm{O}(\log^* n)$ communication rounds [19] which underlines the usefulness of this graph class for distributed computation.

Turán's Bound and a One-round Algorithm. A starting point of our work is an extension of a celebrated theorem by Paul Turán. Turán showed that every graph $G = (V, E)$ contains an independent set of size at least n/d, where d is the average degree of G. This result has been extended by Caro [3] and Wei [20] who showed that there is an independent set of size at least

$$\beta(G) := \sum_{v \in V} \frac{1}{\deg_G(v) + 1} \ ,$$

where $\deg_G(v)$ denotes the degree of vertex v in G. An independent set of expected size $\beta(G)$ can be found by a simple linear time randomized algorithm that follows from an analysis of the Caro-Wei bound by Alon and Spencer in [1]. This algorithm works as follows: Every node v chooses a random real value between 0 and 1 and adds itself to the independent set I if none of its neighbors have chosen a larger real value than v. Then, the probability that a node v is added to the independent set is $\frac{1}{\deg_G(v)+1}$, and, hence, by linearity of expectation, $\mathbb{E}|I| = \sum_{v \in V} \frac{1}{\deg_G(v)+1} = \beta(G)$.

This algorithm can also be implemented distributively in a single communication round. Instead of choosing a random real value, every node chooses a random value from a large enough ordered set (e.g. $\{1, 2, \dots, n^3\}$ suffices) so that neighboring nodes choose different values with large enough probability. In order to be able to determine such a number, nodes require knowledge of n, i.e., the order of the input graph. Furthermore, communicating the chosen value to neighboring nodes requires messages of size $\mathrm{O}(\log n)$. This algorithm fulfills, hence, the restrictions of the $\mathcal{CONGEST}$ model. In the following, we will refer to this algorithm as Alon-Spencer-IS.

It is easy to see that in general graphs, an independent set of size $\beta(G)$ may be a factor $\Theta(n)$ smaller than the independence number $\alpha(G)$[1]. This raises the following questions:

1. Are there interesting graph classes for which $\beta(G)$ is a non-trivial approximation to the independence number $\alpha(G)$?
2. What are the minimum communication requirements for achieving the $\beta(G)$ bound?
3. Is there a one-round independent set algorithm with approximation factor $o(n)$ on general graphs?

Our Results in Detail. Concerning Question 1, we identify that in graphs of polynomially bounded-independence, an independent set of size $\beta(G)$ is a poly-logarithmic approximation to a maximum independent set. For instance on unit disc graphs, an independent set of size $\beta(G)$ is an $\mathrm{O}((\frac{\log n}{\log \log n})^2)$-approximation to a maximum independent set. Furthermore, we prove that our analysis is tight up to a constant factor on d-dimensional unit sphere graphs, for any constant integer d. We also show that on the more general class of k-claw free graphs[2], for $k \geq 3$, a similar result cannot be obtained. In Appendix A, we provide k-claw free graphs for which the Caro-Wei bound is not a poly-logarithmic approximation to the independence number of the graph.

[1] Consider, for instance, the graph $G = (C \cup I, E_1 \cup E_2)$ with $|C| = |I| = n/2$. The edges E_1 turn C into a clique. Furthermore, for every $u \in C$ and $v \in I$, the edge (u, v) is included in E_2. Then, the size of a maximum independent set is $n/2$ while $\beta(G) \leq \frac{3}{2}$.

[2] A graph is k-claw free, if it does not contain the complete bipartite $K_{1,k}$ as an induced subgraph.

With regards to Question 2, we show that for the more general class of O(1)-claw free graphs, the communication requirements can be reduced to an absolute minimum at the price of losing a constant factor. We present a different and even simpler one-round algorithm that computes an independent set of expected size $\Theta(\beta(G))$ using single bit messages, thus decreasing the message sizes from $O(\log n)$ to 1. This algorithm has the additional advantage that it does not require the knowledge of n in advance. The latter property and the low communication requirements allow this algorithm to be implemented in wireless and radio networks. Note that our main result, a poly-logarithmic approximation one-round single bits messages algorithm for the maximum independent set problem in polynomially bounded-independence graphs, follows from the previous two results.

Last, we answer Question 3 in the negative. We provide a lower bound that shows that any possibly randomized one-round algorithm with error probability at most $1/n$ has approximation ratio $\Omega(n)$.

Further Related Work. As already mentioned, independent sets are among the most studied problems in distributed computing. However, most works consider the maximal independent set problem while we consider the maximum independent set problem in this paper. It is known that computing a maximal independent set requires $\Omega(\sqrt{\log n})$ communication rounds [12] in general graphs, and even on a ring, $\Omega(\log^* n)$ rounds are necessary [15, 14]. Concerning approximations to the maximum independent set problem, a $(1+\epsilon)$-approximation can be computed in $O(\log^* n)$ rounds in planar graphs [5]. As in graphs of bounded-independence, a maximal independent set is a constant factor approximation to a maximum independent set, the previously mentioned $O(\log^* n)$ rounds algorithm of Schneider and Wattenhofer [19] gives a constant-factor approximation. Last, we note that the Caro-Wei bound and Turán bound have been previously used as quality guarantees for independent set approximation (e.g., [6]).

Notations. Throughout the paper, we use the following notations. Let $G = (V, E)$ be a graph. For a node $v \in V$, let $\Gamma_G(v)$ denotes the neighborhood of v and $\deg_G(v) = |\Gamma_G(v)|$ its degree. The *d-neighborhood of* v, denoted $\Gamma_G^d(v)$, is the set of nodes of distance at most d from v excluding v, while the set of nodes at distance exactly d from v is denoted by $\Gamma_G^{(d)}(v)$. Let $\Gamma_G^d[v] := \Gamma_G^d(v) \cup \{v\}$ (and $\Gamma_G[v] = \Gamma_G(v) \cup \{v\}$). For a subset of vertices $U \subseteq V$, the graph $G|_U$ is the subgraph of G induced by the vertices U.

Outline. First, in Section 2, we prove our main result that the Caro-Wei bound is a poly-logarithmic approximation to the independence number in polynomially bounded-independence graphs. An algorithm with single-bit messages achieving the Caro-Wei bound up to a constant factor for O(1)-claw free graphs is discussed in Section 3. Then, in Section 4, we show that on general graphs, any possibly randomized distributed one-round algorithm computes an independent set of size at most O(1), while the graph has an independence number of $\Omega(n)$. Last, in Section 5 we show that our analysis of Section 2 is tight for d-dimensional unit sphere graphs.

In Appendix A, we show that in O(1)-claw-free graphs, $\beta(G)$ generally is not a poly-logarithmic approximation to $\alpha(G)$. Last, in Appendix B, we argue that running our algorithm from Section 3 iteratively multiple times does not substantially improve the approximation ratio of the algorithm.

2 Poly-logarithmic Approximation On Bounded-independence Graphs

We show that in graphs of polynomially bounded-independence, an independent set of size $\beta(G)$ is a poly-logarithmic approximation of a maximum independent set.

We first show that in any graph $G = (V, E)$, for any node $v \in V$ and a large enough constant C, the sum of the inverted degrees in the $C\frac{\log n}{\log\log n}$-neighborhood of v is $\Omega(1)$ (Lemma 1). The size of an independent set in such a $C\frac{\log n}{\log\log n}$-neighborhood in a bounded-independence graph is at most $f(C\frac{\log n}{\log\log n})$, by definition. Hence, within the $C\frac{\log n}{\log\log n}$-neighborhood of any node $v \in V$, the ratio between the size of a maximum independent set and the Caro-Wei bound is $O(f(\frac{\log n}{\log\log n}))$. Then, by decomposing the input graph G into components of diameters at most $2C\frac{\log n}{\log\log n}$, we extend this result to hold for the entire graph (Theorem 1).

Lemma 1. *Let $G = (V, E)$ be an arbitrary graph with maximal degree Δ. Let $m = \min\{\Delta, C\frac{\log n}{\log\log n}\}$, for a large enough constant C. Then:*

$$\sum_{u \in \Gamma_G^m[v]} \frac{1}{\deg_G(u)} = \Omega(1).$$

Proof. Let $v \in V$ be any node, and let $d_0 = \deg_G(v)$. For abbreviation, let $s_j = |\Gamma_G^{(j)}(v)|$ for $j \geq 1$. We set $s_0 = 1$ and we clearly have $s_1 = d_0$. Furthermore, let $d_i = \frac{1}{s_i}\sum_{u \in \Gamma_G^{(i)}(v)} \deg_G(u)$ be the average degree of the nodes in $\Gamma_G^{(i)}(v)$. Then, the inverted degree sum of the nodes in the m-neighborhood can be bounded as follows:

$$\begin{aligned}\sum_{u \in \Gamma_G^m[v]} \frac{1}{\deg_G(u)} &= \frac{1}{d_0} + \sum_{j=1}^{m} \sum_{u \in \Gamma_G^{(j)}(v)} \frac{1}{\deg_G(u)} \geq \frac{1}{d_0} + \sum_{j=1}^{m} \sum_{u \in \Gamma_G^{(j)}(v)} \frac{1}{d_j} \\ &= \frac{1}{s_1} + \frac{s_1}{d_1} + \sum_{j=2}^{m} \frac{s_j}{d_j}, \end{aligned} \tag{1}$$

where the first inequality follows from the relationship between the harmonic mean and the arithmetic mean. For $i \geq 2$, consider a node $u \in \Gamma_G^{(i)}(v)$ of degree at least d_i. Then, $\Gamma_G(u) \subseteq \Gamma_G^{(i-1)}(v) \cup (\Gamma_G^{(i)}(v) \setminus \{u\}) \cup \Gamma_G^{(i+1)}(v)$. Hence, $\deg_G(u) \leq s_{i-1} + s_i - 1 + s_{i+1}$, and since $d_i \leq \deg_G(u)$, we also have $d_i \leq s_{i-1} + s_i + s_{i+1}$. Similarly, for d_1 we obtain the inequality $d_1 \leq s_1 + s_2$. Using this in Inequality 1, we obtain:

$$\sum_{u \in \Gamma_G^m[v]} \frac{1}{\deg_G(u)} \geq \frac{1}{s_1} + \frac{s_1}{d_1} + \sum_{j=2}^{m} \frac{s_j}{d_j} \geq \frac{1}{s_1} + \frac{s_1}{s_1 + s_2} + \sum_{j=2}^{m} \frac{s_j}{s_{j-1} + s_j + s_{j+1}}. \tag{2}$$

Suppose that the sequence $(s_i)_{1 \leq i \leq m}$ is not strictly increasing. Let j be the smallest index so that $s_j \leq s_{j-1}$. If $j = 2$, then the term $\frac{s_1}{s_1+s_2}$ of Inequality 2 can be bounded by $\frac{s_1}{s_1+s_2} \geq \frac{s_1}{s_1+s_1} = 1/2$, and thus, $\sum_{u \in \Gamma_G^m[v]} \frac{1}{\deg_G(u)} > \frac{1}{2} = \Omega(1)$. Suppose that $j > 2$. Then, since j is the smallest index, we have $s_{j-2} < s_{j-1}$. Therefore, the addend with index $j-1$ of the sum in the right side in Inequality 2 can be bounded as follows:

$$\frac{s_{j-1}}{s_{j-2} + s_{j-1} + s_j} > \frac{s_{j-1}}{3 \cdot s_{j-1}} = 1/3,$$

which implies $\sum_{u\in\Gamma_G^m[v]}\frac{1}{\deg_G(u)} > \frac{1}{3} = \Omega(1)$. Assume now that the sequence $(s_i)_i$ is strictly increasing. We bound the right side of Inequality 2 as follows:

$$\sum_{u\in\Gamma_G^m[v]}\frac{1}{\deg_G(u)} \geq \frac{1}{s_1}+\frac{s_1}{s_1+s_2}+\sum_{j=2}^{m}\frac{s_j}{s_{j-1}+s_j+s_{j+1}}$$
$$\geq \frac{1}{s_1}+\frac{s_1}{s_1+s_2}+\sum_{j=2}^{m}\frac{s_j}{2\cdot s_j+s_{j+1}}. \tag{3}$$

Let $J\subseteq\{2,\dots,m\}$ be the subset of indices so that for each $j\in J: \frac{s_j}{2\cdot s_j+s_{j+1}} \leq \frac{\log\log n}{\log n}$. This implies that for $j\in J$ we have $s_{j+1}\geq s_j\left(\frac{\log n}{\log\log n}-2\right)$. Since the sequence $(s_i)_i$ is strictly increasing, we can bound the size of the set J as follows:

$$\left(\frac{\log n}{\log\log n}-2\right)^{|J|}\leq n,$$

and therefore $|J| = \mathrm{O}(\frac{\log n}{\log\log n})$. We now set $m = C\frac{\log n}{\log\log n}$ for a large enough constant C so that there are $\Theta(\frac{\log n}{\log\log n})$ indices $i\notin J$ with $\frac{s_i}{2\cdot s_i+s_{i+1}}\geq\frac{\log\log n}{\log n}$. Then, the addends in the right side of Inequality 3 that correspond to those indices $i\notin J$ sum up to a constant which proves part 1 of the result.

We derive now a bound on m that depends on the maximal degree Δ. To this end, we depart from Inequality 3. Notice that the bound on Δ implies $s_j\leq s_{j-1}\Delta$. Therefore, for any j, the addend in Inequality 3 that corresponds to j is bounded as follows: $\frac{s_j}{2s_js_{j-1}}\geq\frac{s_j}{2s_j+\Delta s_j}=\frac{1}{2+\Delta}$. Setting $m=\Theta(\Delta)$ implies that the right side of Inequality 3 sums up to a constant. □

Theorem 1. *Let $G=(V,E)$ be a graph of polynomially bounded-independence with maximal degree Δ and with bounding function f. Then:*

$$\alpha(G)=\mathrm{O}\left(\beta(G)\cdot f(\min\{\Delta,\frac{\log n}{\log\log n}\})\right).$$

Proof. Let $m=\min\{\Delta, C\frac{\log n}{\log\log n}\}$ where C is the constant as in Lemma 1. Let S be a maximal $2m$-independent set in G, i.e., a maximal set of vertices of mutual distance at least $2m$. Let I^* denote a maximum independent set in G. Since S is maximal, every vertex in I^* is at a distance at most $2m$ from a vertex in S, and thus $|I^*|\leq|S|\cdot f(2m)$. Since S is $2m$-independent, the m-neighborhoods around nodes in S are disjoint. Thus, using Lemma 1, we have

$$\beta(G)=\sum_{v\in V}\frac{1}{\deg_G(v)}\geq\sum_{s\in S}\sum_{v\in\Gamma_G^m(s)}\frac{1}{\deg_G(v)}=\Omega(|S|).$$

Thus,

$$\alpha(G)\leq|S|\cdot f(2m)=\mathrm{O}(\beta(G)\cdot f(2m))=\mathrm{O}(\beta(G)f(m)),$$

since f is a polynomial function. □

3 Distributed Algorithm With Single Bit Messages

In the previous section, we showed that an independent set of size $\beta(G)$ is a poly-logarithmic approximation on graphs of polynomially bounded-independence. The ALON-SPENCER-IS algorithm computes an independent set of expected size $\beta(G)$, and thus we obtain a one-round poly-logarithmic approximation algorithm for the maximum independent set problem on graphs of polynomially bounded-independence with message sizes $\mathrm{O}(\log n)$. In this section, we improve on the message complexity of the previous algorithm. We propose an alternative algorithm that computes an independent set of expected size $\Theta(\beta(G))$ on $\mathrm{O}(1)$-claw free graphs using single bit messages. As bounded-independence graphs are $(f(1)+1)$-claw free and $f(1)$ is a constant, this algorithm also constitutes an improvement for bounded-independence graphs.

We will consider the one-round algorithm, Algorithm 1, which can be seen as a simplified version of the well-known distributed maximal independent set algorithm by Luby [16]. In each round of Luby's algorithm, nodes of a general graph $G = (V, E)$ are added to an initially empty independent set. One round consists of two phases: First, every node $v \in V$ pre-selects itself with probability $\Theta(\frac{1}{\deg_G(v)})$ as a candidate to join the independent set. Then, in the second phase, ties are broken among the pre-selected nodes so that nodes with larger degree are preferred. Finally, selected nodes and their neighbors are removed from G, and the round is completed. The algorithm terminates when G is empty. In our version of the algorithm, a simplified method for breaking ties is used. Instead of preferring nodes with larger degree, we only add a pre-selected node to the independent set if none of its neighbors have been pre-selected. This method of breaking ties has been previously used, e.g., in [8, 11, 9].

Algorithm 1 One-round independent set algorithm

Require: $G = (V, E)$ {Input graph}
$I \leftarrow \varnothing$ {the independent set to be computed}
$p_i \leftarrow \frac{1}{2\deg(v)}$
$T_v \leftarrow$coin(p_i) {Pre-selection step: If $T_v = \mathit{true}$ then v is a candidate to join the IS}
for all $v \in V$ with $T_v = \mathit{true}$ **do**
  if $\bigvee_{u \in \Gamma_G(v)} T_u = \mathit{false}$ **then** {Check whether a neighbor of v has been pre-selected}
    $I \leftarrow I \cup \{v\}$ {v is selected into the IS}
  end if
end for

We first derive a bound on the inverted degree sum of the neighborhood of an arbitrary node $v \in V$ in a k-claw free graph $G = (V, E)$.

Lemma 2. *Let $G = (V, E)$ be a k-claw free graph. Then for every $v \in V$,*

$$\sum_{u \in \Gamma_G(v)} \frac{1}{\deg_G(u)} \leq k - 1 \ .$$

Proof. Let v be a node and let $H_v = G|_{\Gamma_G(v)}$ be the subgraph induced by v's neighbors. Observe that for $u \in V(H)$, $deg_G(u) \geq deg_H(u) + 1$. Since G is k-claw free, $\alpha(H) \leq k - 1$. Thus, using the Caro-Wei bound, we get that

$$\sum_{u \in \Gamma_G(v)} \frac{1}{\deg_G(u)} \leq \sum_{u \in V(H)} \frac{1}{\deg_H(u) + 1} \leq \alpha(H) \leq k - 1 \ .$$

□

Theorem 2. *Algorithm 1 is a randomized distributed one-round algorithm using single bit messages that finds independent sets with expected $\Theta(\beta(G))$ size on graphs G with constant claw size. In particular, when G is polynomially bounded-independence, it achieves an expected approximation ratio* $\mathrm{O}(f(\min\{\Delta, \frac{\log n}{\log\log n}\}))$.

Proof. Let v be any node in G. Algorithm 1 adds v to the independent set if two independent events happen: v is pre-selected in Line 3 of Algorithm 1 while none of its neighbors are pre-selected. Then, by the linearity of expectation,

$$\begin{aligned}\mathbb{E}\,|I| &= \sum_{v\in V}\mathbb{P}\left[v\in I\right] = \sum_{v\in V}\mathbb{P}\left[v \text{ pre-selected }\right]\cdot\mathbb{P}\left[v\in I \mid v \text{ pre-selected }\right]\\ &= \sum_{v\in V}\frac{1}{\deg_G(v)}\cdot\prod_{u\in\Gamma_G(v)}(1-\frac{1}{\deg_G(u)}) = \sum_{v\in V}\frac{1}{\deg_G(v)}\cdot\Theta\left(e^{-\sum_{u\in\Gamma_G(v)}\frac{1}{\deg_G(u)}}\right)\\ &= \Theta(1)\cdot\beta(G)\ ,\end{aligned}$$

applying Lemma 2 in the last equality. If G is of bounded-independence with bounding function f, it is $(f(1)+1)$-claw free, which is a constant. Applying Theorem 1 we obtain the approximation result. □

Implementing Algorithm 1 in Beep Models and Wireless Networks. Algorithm 1 places minimal demands on the underlying model in which it is implemented. Initially, nodes only require the knowledge of their own degree (or of an estimate thereof), and, in particular, information about the network size is not needed. In many wireless networks, the degree of local congestion provides a good estimate for a node's degree, and congestion can often be inferred using carrier sensing techniques.

The communication structure of the algorithm naturally fits beep-like models and wireless networks. Pre-selected nodes send a signal to all their neighbors. Hence, models that only support radio broadcast rather than the transmission of individual messages to neighboring nodes are sufficient for implementing this step. With regards to the reception of signals from neighboring nodes, in Line 5 of the algorithm, nodes only have to be able to learn whether no neighboring node emitted a signal or whether at least one neighboring node emitted a signal. This type of information matches precisely what can be learned by a node in one round in the discrete beeping model as introduced in [4]. Also, in wireless networks, carrier sensing can yield information that is possibly weaker (a node that is within a short range did transmit) but sufficient for the operation of our algorithm.

4 Lower Bound for One-round Algorithms on General Graphs

In this section, we prove that no distributed one-round algorithm can compute an independent set whose size exceeds the Caro-Wei bound by more than a constant. In particular, every possibly randomized distributed one-round algorithm on general graphs has an approximation factor of $\Omega(n)$, where n is the number of vertices of the input graph.

Consider an arbitrary d-regular bipartite graph $H=(A,B,E)$ with $|A|+|B|=n'$. Let $G=(V,E)$ be the graph consisting of a $(d+1)$-clique and a copy of H which is disjoint from the $(d+1)$-clique. Let $n=|V|$, and hence $n'=n-d-1$. G is clearly d-regular. Furthermore, since H contains an independent set of size $n'/2$, the independence number of G is $\alpha(G) = \frac{n-d-1}{2}$. We assume that each node has a unique label chosen from $\mathcal{U} = \{1, \ldots, m\}$, where $m \geq n$. Let $\mathcal{L}$ denote the set of all possible labellings.

In order to prove our lower bound, we exploit the fact that all nodes in V have the same local *views*, i.e., in one round, all nodes can only learn the d labels of their adjacent nodes. As all nodes run the same algorithm, clearly in average over all possible labellings $\mathcal{L}$, the probabilities for all nodes to end up in I is equal. This fact is used in the following theorem:

Theorem 3. *Every possibly randomized one-round distributed algorithm for maximum independent set has an expected approximation factor of at least* $\frac{(n-\Delta-1)(\Delta+1)}{2n}$, *where* Δ *is the maximal degree of the input graph.*

Proof. Consider the d-regular graph $G = (V, E)$ as defined above. Then $\Delta = d$. Consider a possibly randomized one-round algorithm for maximum independent set. Then, as previously argued, for all $u, v \in V$, we have:

$$\mathbb{P}\left[u \in I\right] = \mathbb{P}\left[v \in I\right], \text{ and} \tag{4}$$

$$\mathbb{E}|I| = \sum_{u \in V} \mathbb{P}\left[u \in I\right], \tag{5}$$

where the probabilities are taken over all possible labellings $\mathcal{L}$ and the random coin flips of the algorithm. Let p be the probability that a node ends up in I. Let C denote the $(d+1)$-clique of G. Then, $p \cdot |C| = \mathbb{E}|I \cap C| \leq 1$, and hence, $p \leq \frac{1}{|C|} = \frac{1}{d+1}$. Therefore, $\mathbb{E}|I| \leq np = \frac{n}{d+1}$. Next, since $\alpha(G) = \frac{n-d-1}{2}$, the expected approximation ratio is at least $\frac{(n-d-1)(d+1)}{2n}$. □

Remark. The graph G of the previous construction is disconnected. This can be circumvented by removing arbitrary edges u_1v_1, u_2v_2, where u_1v_1 is contained in the $(d+1)$-clique and u_2v_2 is outside the $(d+1)$-clique, and reinserting edges u_1u_2 and v_1v_2. The resulting graph is connected and equally suits for proving the same lower bound.

5 Lower Bound for d-dimensional Unit Sphere Graphs

In this section, we show that the statement of Theorem 1, i.e., $\alpha(G) = \mathrm{O}\left(\beta(G) f(\min\{\Delta, \frac{\log n}{\log\log n}\})\right)$ for any graph $G = (V, E)$ of polynomially bounded-independence with bounding function f, is tight for d-dimensional unit sphere graphs. As a consequence, the analysis of Algorithm 1 is also tight.

In Appendix B, we investigate on the performance of running multiple rounds of Algorithm 1. We show that a super-constant number of iterations is necessary in order to improve on the one-round bound performance by more than a constant factor.

A d-dimensional *unit sphere graph* $G = (S, E)$ is the intersection graph of d-dimensional unit spheres $S = \{s_1, \ldots, s_n\}$ (all spheres have the same radius): Each sphere s_i constitutes a vertex in G and two spheres are adjacent iff they intersect. For $d = 1$, a unit sphere graph is a unit interval graph, and for $d = 2$, a unit sphere graph is a unit disc graph.

Let $d > 0$ be some fixed dimension. We will denote our hard instance graph with $H_k = (V_H, E_H)$ where k is a parameter which we define later. We start our construction of H_k with a *grid graph* $G_k = (V_G, E_G)$ that is parametrized by an integer $k \geq 1$. The vertex set of G_k is defined as $V_G = \{v_x \,|\, x \in \{0, 1, \ldots, k-1\}^d\}$. Let v_x, v_y with $x, y \in \{0, \ldots, k-1\}^d$ be two vertices of V_G. Then v_x and v_y are adjacent iff $|x - y| = 1$, where $|x| = \sum_{1 \leq i \leq d} |x_i|$.

The hard instance graph H_k is obtained from G_k as follows: For every vertex $v_x \in V_G$, a clique C_x of size $s(|x|)$ is introduced in H_k, where $s(i) = d^i k^{di} \log^i n$. Suppose that v_x and v_y are adjacent nodes in G_k. Then all nodes of C_x are connected to all nodes of C_y in H_k, or, in other words, $C_x \cup C_y$ also forms a clique in H_k.

First, notice that the graph H_k is in fact a d-dimensional unit sphere graph. Each vertex $v \in C_x \subseteq V_H$ with $x \in \{0, \ldots, k-1\}^d$ corresponds to a sphere centered at position x with radius $1/2$ (for convenience, in this construction we suppose that all spheres have the radius $1/2$ instead of 1). An example is provided in Figure 1.

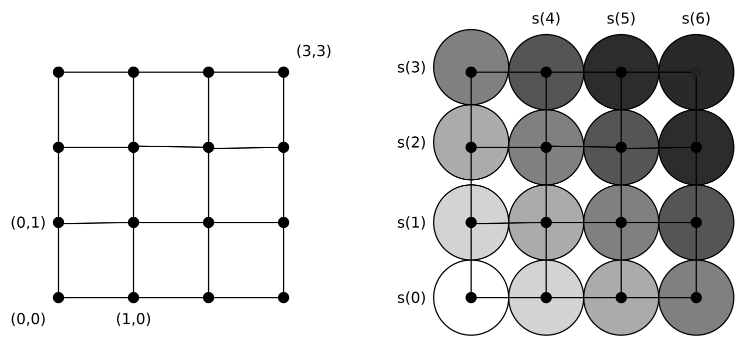

Fig. 1. Illustration of the two dimensional case: On the left, the grid graph G_4 is illustrated. On the right, the hard instance unit disc graph H_4 is shown. H_4 is obtained from G_4 by replacing each node at position (i, j) with a clique of size $s(i + j)$.

We state now that H_k is of bounded-independence with respect to the bounding function $f(r) = (2r + 1)^d$

Lemma 3. *The d-dimensional unit sphere graph H_k is of bounded independence with respect to the bounding function $f(r) = (2r + 1)^d$.*

Proof. The size of an independent set in the k-neighborhood of a node $v \in C_x \subseteq V_H$ for some $x \in \{0, \ldots, k-1\}^d$ is the same as the size of an independent set of the node $v_x \in V_G$ in the corresponding grid graph. Therefore, the r-neighborhood of an arbitrary node $v_x \in V_G$ with $x \in \{0, \ldots, k-1\}^d$ is a subset of the nodes with indices $j \in \{x_1 - r, \ldots, x_1 + r\} \times \cdots \times \{x_d - r, \ldots, x_d + r\}$. Therefore, $|\{x_1 - r, \ldots, x_1 + r\} \times \cdots \times \{x_d - r, \ldots, x_d + r\}| = (2r + 1)^d$ is an upper bound on the size of an independent set in the r-neighborhood of v. □

Next, we identify the correct value for k so that graph H_k has $\mathrm{O}(n)$ vertices, and we show that $\beta(H_k) = \mathrm{O}(1)$.

Lemma 4. *Consider graph $H_k = (V_H, E_H)$, and let $k = \mathrm{O}(\frac{\log n}{d^2 \log \log n})$. Then: $|V_H| = \mathrm{O}(n)$ and $\beta(H_k) = \mathrm{O}(1)$.*

Proof. Denote by n_i the number of cliques at distance i from the clique with index $(0, \ldots, 0)$. Furthermore, denote by $V_i := \{v \in C_x : |x| = i\}$ the set of nodes at distance i from the clique with index $(0, \ldots, 0)$.

First, note that by construction of H_k we have $n_i \leq n_{i+1}d$. This allows us to establish a relation between $|V_i|$ and $|V_{i+1}|$:

$$|V_i| = n_i \cdot s(i) \leq n_{i+1}d \cdot (d^i k^{di} \log^i n) \leq n_{i+1}(d^{i+1}k^{di} \log^i n) = \frac{|V_{i+1}|}{k^d \log n}.$$

Then, as $|V_H| = \sum_{i\in\{0,\dots,d(k-1)\}} |V_i|$ and by the previous inequality, we obtain: $|V_H| = \mathrm{O}(|V_{d(k-1)}|)$. Then, setting $k = \Theta(\frac{\log n}{d^2 \log\log n})$ proves the first part of the lemma:

$$|V_H| = \mathrm{O}\left(|V_{d(k-1)}|\right) = \mathrm{O}\left(d^{kd}k^{d^2k} \log^{kd} n\right) = \mathrm{O}(n).$$

Next, in order to prove that $\beta(H_k) = \mathrm{O}(1)$, notice that $|V_i| \leq n_i s(i)$. Moreover, the nodes of V_i have a degree of at least $s(i+1)$, the size of a clique at distance $i+1$. Each node of the clique $C_{(k-1,\dots,k-1)}$ clearly has a degree of at least $s(d(k-1))$. Thus, we have:

$$\sum_{v\in V_H} \frac{1}{\deg_{H_k}(v)} = \left(\sum_{i\in\{0,\dots,d(k-1)-1\}} n_i \cdot \frac{s(i)}{s(i+1)}\right) + \frac{n_{d(k-1)}s_{d(k-1)}}{s_{d(k-1)}} \leq$$
$$\left(\sum_{i\in\{0,\dots,d(k-1)-1\}} k^d \cdot \frac{1}{dk^d \log n}\right) + 1 = \frac{k-1}{\log n} + 1 = \mathrm{O}(1),$$

where we used the rough estimate $n_i \leq k^d$. □

Finally, we obtain the main theorem of this section on the performance of Algorithm 1.

Theorem 4. *Consider graph $H_k = (V_H, E_H)$, and let $k = \mathrm{O}(\frac{\log n}{d^2 \log\log n})$. Then, Algorithm 1 computes an $\Omega((\frac{\log n}{d^2 \log\log n})^d)$ approximation to the maximum independent set problem on H_k.*

Proof. Lemma 4 yields that the graph H has $\mathrm{O}(n)$ vertices, and the inverted degree sum of H is $\mathrm{O}(1)$. As in Algorithm 1 the probability that a node ends up in the independent set is bounded from above by its inverted degree, Algorithm 1 computes an independent set of expected size $\mathrm{O}(1)$. Since the graph H contains an independent set of size $\Omega((\frac{\log n}{d^2 \log\log n})^d)$, the theorem follows. □

References

1. Noga Alon and Joel H Spencer. *The probabilistic method*. John Wiley & Sons, 2004.
2. Leonid Barenboim. On the locality of some NP-complete problems. In *Proc.of International Colloquium on Automata, Languages, and Programming, ICALP'12 (2)*, pages 403–415, 2012.
3. Y. Caro. New results on the independence number. Technical report, Tel Aviv University, 1979.
4. Alejandro Cornejo and Fabian Kuhn. Deploying wireless networks with beeps. In *Proceedings of the 24th International Conference on Distributed Computing*, DISC'10, pages 148–162, Berlin, Heidelberg, 2010. Springer-Verlag.
5. Andrzej Czygrinow, Michal Hańćkowiak, and Wojciech Wawrzyniak. Fast distributed approximations in planar graphs. In *Proceedings of the 22Nd International Symposium on Distributed Computing*, DISC '08, pages 78–92, Berlin, Heidelberg, 2008. Springer-Verlag.
6. Bjarni V. Halldórsson, Magnús M. Halldórsson, Elena Losievskaja, and Mario Szegedy. Streaming algorithms for independent sets. In Samson Abramsky, Cyril Gavoille, Claude Kirchner, Friedhelm Meyer auf der Heide, and Paul G. Spirakis, editors, *ICALP (1)*, volume 6198 of *Lecture Notes in Computer Science*, pages 641–652. Springer, 2010.
7. Magnús M. Halldórsson. Wireless scheduling with power control. *ACM Trans. Algorithms*, 9(1):7:1–7:20, December 2012.
8. Magnús M. Halldórsson and Christian Konrad. Distributed algorithms for coloring interval graphs. In *Proceedings of the 28th International Conference on Distributed Computing*, DISC'14, 2014.
9. Magnús M. Halldórsson and Pradipta Mitra. Nearly optimal bounds for distributed wireless scheduling in the SINR model. In *Proceedings of the 38th International Conference on Automata, Languages and Programming - Volume Part II*, ICALP'11, pages 625–636, Berlin, Heidelberg, 2011. Springer-Verlag.
10. R. M. Karp. Reducibility Among Combinatorial Problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.
11. Thomas Kesselheim and Berthold Vöcking. Distributed contention resolution in wireless networks. In *Proceedings of the 24th International Conference on Distributed Computing*, DISC'10, pages 163–178, Berlin, Heidelberg, 2010. Springer-Verlag.
12. Fabian Kuhn, Thomas Moscibroda, and Roger Wattenhofer. What cannot be computed locally! In *Proceedings of the Twenty-third Annual ACM Symposium on Principles of Distributed Computing*, PODC '04, pages 300–309, New York, NY, USA, 2004. ACM.
13. Fabian Kuhn, Rogert Wattenhofer, and Aaron Zollinger. Ad-hoc networks beyond unit disk graphs. In *Proceedings of the 2003 Joint Workshop on Foundations of Mobile Computing*, DIALM-POMC '03, pages 69–78, New York, NY, USA, 2003. ACM.
14. Juhana Laurinharju and Jukka Suomela. Brief announcement: Linial's lower bound made easy. In *Proceedings of the 2014 ACM Symposium on Principles of Distributed Computing*, PODC '14, pages 377–378, New York, NY, USA, 2014. ACM.
15. Nathan Linial. Locality in distributed graph algorithms. *SIAM J. Comput.*, 21(1):193–201, February 1992.
16. M Luby. A simple parallel algorithm for the maximal independent set problem. In *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing*, STOC '85, pages 1–10, New York, NY, USA, 1985. ACM.
17. Moni Naor and Larry Stockmeyer. What can be computed locally? *SIAM J. Comput.*, 24(6):1259–1277, December 1995.
18. Ronitt Rubinfeld. Sublinear time algorithms. In *Proceedings of the International Congress of Mathematicians*, 2006.
19. Johannes Schneider and Roger Wattenhofer. An Optimal Maximal Independent Set Algorithm for Bounded-Independence Graphs. *Distributed Computing*, 22, March 2010.
20. V.K. Wei. A lower bound on the stability number of a simple graph. Technical report, Bell Laboratories, 1981.
21. David Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. In *Proceedings of the Thirty-eighth Annual ACM Symposium on Theory of Computing*, STOC '06, pages 681–690, New York, NY, USA, 2006. ACM.

A The Caro-Wei Bound in Claw-Free-Graphs

For $d \geq 2$, we construct now a $(d+1)$-claw free graph G on $\mathrm{O}(n)$ vertices that contains an independent set of size $2^{\sqrt{\log(n+d)}-\log d}$ while $\beta(G) \leq \sqrt{\log(n-d)}+1$. This construction shows that, generally, in $(d+1)$-claw free graphs, $\beta(G)$ is not a poly-logarithmic approximation to the size of a maximum independent set.

To this end, let T denote a complete d-ary rooted tree of depth $\lfloor\sqrt{\frac{\log n}{\log d}}\rfloor$. Then, let G be the graph obtained from T by replacing every node in T at level i by a clique of size $c_i = \lfloor\frac{n}{d^{(i+1)i/2}}\rfloor$. Two cliques C_u, C_v in G that correspond to two adjacent nodes $u, v \in T$ are entirely interconnected in G, or, in other words, $G|_{C_u \cup C_v}$ is a clique. If u and v are not adjacent, then there is no edge between C_u and C_v. The root of T has been replaced by a clique of size n. Let L_0 denote this clique. Even though G is not a tree, we will refer to the nodes at distance i from one of the nodes of L_0 as the nodes at level i, and we denote the set of those nodes by L_i. The construction is so that level i contains $|L_i| = d^i \cdot \lfloor\frac{n}{d^{(i+1)i/2}}\rfloor \leq \lfloor\frac{n}{d^{(i-1)i/2}}\rfloor$ nodes. Therefore, graph G contains $\mathrm{O}(n)$ nodes in total.

Theorem 5. *For any $d \geq 2$, there exists a $(d+1)$-claw free graph on $\mathrm{O}(n)$ vertices so that:*

$$\frac{\alpha(G)}{\beta(G)} \geq \frac{2^{\sqrt{\log(n+d)}-\log d}}{\sqrt{\log(n-d)}+1}.$$

Proof. Consider graph G of the previous construction. The nodes at the last level $L_{\lfloor\sqrt{\frac{\log n}{\log d}}\rfloor}$ contain an independence set of size at least $2^{\sqrt{\log(n+d)}-\log d}$, as the tree T contains $d^{\lfloor\sqrt{\frac{\log n}{\log d}}\rfloor} \geq 2^{\sqrt{\log(n+d)}-\log d}$ nodes on this level, which are also the leaves of T. Furthermore, $\beta(G)$ evaluates to:

$$\beta(G) = \sum_{v \in G} \frac{1}{\deg_G(v)+1} \leq |L_0| \cdot \frac{1}{|L_0|} + \left(\sum_{i=1}^{\lfloor\sqrt{\frac{\log n}{\log d}}\rfloor} |L_i| \frac{1}{c_{i-1}} \right)$$

$$\leq 1 + \sum_{i=1}^{\lfloor\sqrt{\frac{\log n}{\log d}}\rfloor} \lfloor\frac{n}{d^{(i-1)i/2}}\rfloor \cdot \frac{1}{\lfloor\frac{n}{d^{i(i-1)/2}}\rfloor} \leq \sqrt{\log(n-d)}+1,$$

which implies the result. □

B Lower Bound for Multiple Rounds

The construction of the hard instance graph H_k from Section 5 allows us to prove a lower bound on the performance of running multiple iterations of Algorithm 1. We consider the algorithm as depicted in Algorithm 2 (We denote Algorithm 1 by One-round-IS):

Algorithm 2 Multiple rounds independent set algorithm

Require: $G = (V, E)$ {Input graph}, r {number of rounds}
$V' \leftarrow V$ {active nodes}, $I \leftarrow \varnothing$ {the independent set to be computed}
for $i = 1 \ldots r$ **do**
  $I \leftarrow I \cup$ One-round-IS$(G|_{V'})$ {Run Algorithm 1}
  $V' \leftarrow V' \setminus (I \cup \Gamma_G(I))$
end for
return I

We will prove now that when running Algorithm 2 on the graph H_k of the previous subsection for r rounds, the algorithm computes an independent set of size at most r^d with probability $(1 - \mathrm{O}(\frac{1}{d \log n}))^i$. As the graph H_k contains an independent set of size $\Omega(k^d)$, this proves that the approximation ratio of Algorithm 2 is $\Omega\left((\frac{k}{r})^d\right)$.

Consider the situation of Algorithm 2 at the end of the ith iteration of the for-loop. We will prove that at this moment, all cliques C_x with $|x| \leq k^d - 2i \in V'$ are still contained in V' with high probability. In the following, denote by V'_i the set V' of Algorithm 2 after the ith iteration of the for-loop, and denote by I_i the set I after the ith iteration of the for-loop. Due to space restrictions, the proof of the following lemma can be found in the appendix.

Lemma 5. *With probability at least* $(1 - \mathrm{O}(\frac{1}{d \log n}))^i = \Omega(e^{\frac{i}{d \log n}})$*:*

$$\bigcup_{x \in \{0,\ldots,k-1\}^d \text{ with } |x| \leq k^d - 2i} C_x \subseteq V'_i.$$

Proof. We prove this lemma by induction on i. Consider the first iteration of the algorithm. Let C_x be a clique such that $|x| < k^d$. Then, the probability that a node of C_x is chosen into the independent set is $\mathrm{O}(\frac{|C_x|}{s(|x|+1)}) = \mathrm{O}(\frac{s(|x|)}{s(|x|+1)}) = \mathrm{O}(\frac{1}{dk^d \log n})$. Therefore, and by the union bound, the probability that any of the cliques C_x with $|x| < k^d$ is chosen into the independent set is $\mathrm{O}(\frac{1}{d \log n})$ since there are only $k^d - 1$ such cliques. Hence, with probability at least $1 - \mathrm{O}(\frac{1}{d \log n})$, any clique C_x with $|x| \leq k^d - 2$ is such that $C_x \in V'_1$.

Consider now iteration $i + 1$. By the induction hypothesis, with probability $(1 - \mathrm{O}(\frac{1}{d \log n}))^i$, every clique C_x with $|x| \leq k^d - 2i$ is included in V'_i. Then, by the same argument as before, the probability that a node of any of the cliques C_x with $|x| < k^d - 2i$ is chosen into the independent set in iteration $i + 1$ is at most $\mathrm{O}(\frac{1}{d \log n})$. Therefore, with probability $(1 - \mathrm{O}(\frac{1}{d \log n}))^{i+1}$ all cliques C_x with $|x| \leq k^d - 2(i + 1)$ are included in V'_{i+1}. This proves the lemma. ☐

Theorem 6. *Let* $H = H_k$ *for the value* $k = \frac{\log n}{d^2 \log \log n}$*. Then, Algorithm 2 computes an* $\Omega((\frac{\log n}{rd^2 \log \log n})^d)$ *approximation to the maximum independent set problem on* H *in* r *rounds with probability* $\Omega(e^{\frac{r}{d \log n}})$*.*

Proof. By Lemma 5, with probability $\Omega(e^{\frac{r}{d \log n}})$, all cliques C_x with $|x| \leq k^d - 2r$ are included V'_r, and hence, among the eliminated nodes, any independent set is of size $\mathrm{O}(r^d)$. As the graph H contains an independent set of size $\Omega((\frac{\log n}{\log \log n})^d)$, the result follows. ☐

Remark on Alon-Spencer-IS. Both the one-round and the multi-round lower bound result apply in the same way for Alon-Spencer-IS. The only property of Algorithm 1 that is needed in the analyses of both lower bound results is the fact that the probability that a node v is selected into the independent set is $\Theta(\frac{1}{\deg_G(v)})$. As the same property holds for ALON-SPENCER-IS, these results immediately carry over.