Distributed algorithms for random graphs

K. Krzywdziński * and K. Rybarczyk †
Adam Mickiewicz University, Poznań, Poland

Abstract

In this article we study statistical properties of a commonly used network model – an Erdős–Rényi random graph $G(n,p)$. We are interested in the performance of distributed algorithms on large networks, which might be represented by $G(n,p)$. We concentrate on classical problems from the field of distributed algorithms such as: finding a maximal independent set, a vertex colouring, an approximation of a minimum dominating set, a maximal matching, an edge colouring and an approximation of a maximum matching. We propose new algorithms, which with probability close to one as $n \to \infty$ construct anticipated structures in $G(n,p)$ in a low number of rounds. Moreover, in some cases, we modify known algorithms to obtain better efficiency on $G(n,p)$.

1 Introduction

1.1 Background

In many distributed systems, such as for example internet network, ad hoc networks or sensor networks, there are many entities, which operate in the system. Their processors are active at any moment and may perform some local computations. Moreover the entities have ability to communicate with each other to achieve some goals. We study a theoretical model called LOCAL, which captures some properties of those distributed systems (see [29]). LOCAL is a massage passage model, i.e. each processor wishing to communicate with an other, sends a message via an available link. In the described model the network of possible links is represented by an undirected graph, whose vertices represent processors (computational units) and edges stand for communication channels (links). A vertex may communicate directly only with its neighbours in the graph. It is generally assumed that in the distributed systems the communication is the cost that dominates others more traditional ones, such as processing or storage capacity of a single vertex. Therefore in the LOCAL model the time complexity of an algorithm is expressed in the number of rounds in which processors send messages. We assume that the network is synchronized and in one

*supported by grant N N206 565740
†partially supported by the National Science Center grant DEC–2011/01/B/ST1/03943

round a vertex can send and receive messages from its neighbours as well as perform some local computations. Neither the amount of local computations nor the size of messages sent is restricted in any way.

Most known distributed algorithms either work on general graphs, i.e. study the worst cases, or concentrate on a particular family of graphs, such as for example: trees, bounded growth graphs, planar graphs or bounded degree graphs. In this work we analyse the performance of distributed algorithms on average graphs. To this end we use an Erdős–Rényi random graph $G(n,p)$ with independent edges. Given an integer n and $p = p(n) \in [0,1]$, $G(n,p)$ is a graph with the vertex set $V = \{1,\ldots,n\}$, in which each pair of vertices is added to the edge set independently with probability p. The aim of this paper is by means of $G(n,p)$ analyse statistical properties of large distributed networks. We concentrate on the case, where the set of processors is large and examine asymptotic properties of $G(n,p)$, when $p = p(n)$ is a function of n and the size of $G(n,p)$ is tending to infinity.

A motivation for our research is the fact that random graphs are a commonly used tool to depict and analyse statistical behaviour of large networks. Moreover the structure of random graphs is usually less complicated than this of general graphs in the worst case. Therefore we should expect that there exist algorithms, which perform better on random graphs then in general graphs. The investigation of properties of $G(n,p)$ may be regarded as a step in the study of statistical properties of performance of distributed algorithms on many types of large networks. To the best of our knowledge the only known distributed algorithm designed to work on $G(n,p)$ was introduced by Levy et al. in [24] and constructs a Hamilton cycle in $G(n,p)$.

We concentrate on several problems, which are considered important in the distributed computing. In order to present the problems we recall some notation and definitions. For any graph G we denote by $V(G)$ and $E(G)$ its vertex set and its edge set, respectively. For any graph G and $V' \subseteq V(G)$, $N_G(V')$ is the set of neighbours in G of vertices from V' and $G[V']$ is the subgraph of G induced on V'. A subset $V' \subseteq V(G)$ is an independent set in G if $G[V']$ has no edges. $V' \subseteq V(G)$ is a maximal independent set (a MIS) if there is no other independent set which contains V'. By a colouring (a vertex colouring) of G with k colours we mean a division of $V(G)$ into k disjoint independent sets (each set of the division gets a distinct colour). A matching in G is a subset of $E(G)$ such that no two edges of the matching share a vertex. Maximal matching is a matching, which is not contained in any other matching in G. Moreover a maximum matching in G is a matching of the maximum size in G. An edge colouring of G with k colours is a division of $E(G)$ into k matchings. A set $V' \subseteq V(G)$ is a dominating set of G if $N_G(V') \cup V' = V(G)$. Note that any MIS is a dominating set, however it is not necessarily a minimum dominating set (a dominating set with the minimum number of vertices). Given $C > 0$, a C–approximation of a maximum matching M (minimum dominating set D) is a matching (a dominating set) of size at least $C|M|$ (at most $C|D|$).

All results in the paper are asymptotic. We use the standard Landau notation consistently with [18]. All limits as well as asymptotic notation are taken

as $n \to \infty$, where n stands for the number of vertices of the considered graph. We use the phrase *with high probability* to say with probability tending to one as $n \to \infty$. All inequalities hold for n large enough. All logarithms are natural except this in the definition of $\log^* n$.

1.2 Our contribution and related results

We start with a general remark concerning all distributed algorithms on $G(n,p)$. For a connected graph G we denote by $d_G(v,w)$ the length of the shortest path connecting v and w in G and by $diam(G) = \max\{d_G(v,w) : v,w \in V(G)\}$ the diameter of G. For a disconnected graph G the diameter is the largest diameter of the components. By the definition of the LOCAL model, each problem in a graph G may be solved in $O(diam(G))$ rounds (if G is disconnected then the problem is solved in each component independently). Usually for $G(n,p)$ this heuristic with high probability gives better running time than most known distributed algorithms. Namely, by classical results on random graphs we have the following theorem.

Theorem 1.1. *Let $c > 0$ be a constant and $p = p(n)$ be such that $np > c$ then with high probability $diam(G(n,p)) = O\left(\frac{\ln n}{\ln np}\right)$.*

For proofs and tight estimates on the constants see for example [8, 10, 20, 31] and references therein. In particular Theorem 1.1 implies that if $np \geq n^\varepsilon$ for some constant $\varepsilon > 0$, then with high probability $G(n,p)$ has a constant diameter. We also recall that for $np = o(1)$ with high probability $G(n,p)$ is a forest with all components of size $O(\ln n)$, therefore with high probability known algorithms working on trees perform efficiently on such graphs. Thus we might concentrate on the case $c < np < n^\varepsilon$, for some constants $c, \varepsilon > 0$. However, for the sake of completeness, we state results for a wider range of parameters even if with high probability $diam(G(n,p))$ is smaller than the complexity of the proposed algorithm.

For an overview of the distributed algorithms we refer the reader to handbooks such us for example [6] or [29]. Here we just mention some results concerning general graphs and relate them to our algorithms. The algorithms introduced in this article exploit statistical properties of $G(n,p)$ and with high probability have running time $o(diam(G(n,p)))$ for a wide range of parameters. Moreover they perform on an average graph far better than known algorithms for general graphs.

First we consider the problem of constructing a MIS and a vertex colouring. In general graphs in the LOCAL model a MIS may be found by a randomised algorithm in $O(\log n)$ time (see [1], [21] and [25]). Moreover it may be constructed deterministically using an algorithm of Panconesi and Srinivasan [28] in $O(2^{O(\sqrt{\ln n})})$ rounds or an algorithm of Barenboim and Elkin [3] in $O(a + a^\varepsilon \log n)$ rounds, for an arbitrarily small constant $\varepsilon > 0$. Here $a = a(G)$ is the arboricity of G, i.e. the minimum number of edge disjoint forests $F_1, \ldots, F_{a(G)}$, whose union covers the entire edge set of G. In known

distributed algorithms colouring verices with k colours, k is either a function of $\Delta(G)$ (the maximum vertex degree of an input graph G) or depends on $a(G)$. In the case $np = \Omega(1)$ with high probability $a(G) = \Theta(np)$ (see Lemma 2.3). Moreover $\Delta(G(n,p)) = \Theta(\ln n/\ln(\ln n/\ln(np)))$ for $np = o(\ln n)$ and $np = \Omega(1)$ and $\Delta(G(n,p)) = \Theta(np)$ for $np = \Omega(\ln n)$. The algorithms presented in [3] and [28] give colourings of an input graph with $O(\Delta)$ colours in $O(\Delta^{\varepsilon}\ln n)$ rounds (for an arbitrarily small constant $\varepsilon > 0$) and $O(2^{O(\sqrt{\ln n})})$ rounds, respectively. Moreover in the result from [4] Δ may be replaced by $a = a(G)$.

In Section 3 we present algorithms GnpMIS and GnpColour. GnpMIS with high probability finds a MIS in $G(n,p)$ in $O(\ln\ln n \cdot \ln(a(G(n,p))))$ rounds and GnpColour constructs a colouring with $O(a(G(n,p)))$ colours in $O(\ln\ln n)$ rounds. This combined with Theorem 1.1 shows that in the worst case in $G(n,p)$ with high probability a MIS may be constructed in $O(\min\{\ln\ln n \cdot \ln a, \ln n/\ln a\}) = O(\sqrt{\ln n/\ln\ln n})$ rounds and the colouring may be found in $O(\min\{\ln\ln n, \ln n/\ln a\}) = O(\ln\ln n)$ time.

We also show that GnpMIS with high probability resolves the problem of $(2+\varepsilon)$–approximation of a minimum dominating set and we argue that possibly the constructed MIS might be $(1+\varepsilon)$–approximation of a minimum dominating set. It should be stressed that in general graphs best algorithms are randomised and achieve a constant approximation only in the case of graphs with either bounded degree or bounded arboricity (see [19, 22, 23]).

The remaining algorithms resolve the edge colouring and the matching problems. In a general case a maximal matching problem may be resolved deterministically in polylogarithmic time [26] and using a randomised algorithm in $O(\sqrt{\ln n} + \ln\Delta)$ rounds [7]. The running times of the algorithms constructing edge colourings depend on the number of colours one want to use. For algorithms, which efficiently construct edge colouring with more than $2\Delta - 1$ colours we refer the reader to [5]. Here we propose an algorithm, which using a similar heuristic to this of Panconesi and Rizzi from [27] with high probability colours $G(n,p)$ with $2\Delta - 1$ colours. A modification allows to reduce the running time from $O(a + \ln n)$ in general graphs to $O(a + \ln\ln n)$ in $G(n,p)$. Moreover we show how to adapt the algorithm to construct a maximal matching in the same number of rounds.

Our last algorithm resolves the problem of $(1-\varepsilon)$–approximation of a maximum matching in $G(n,p)$. We show that in the case of random graphs the problem reduces to the problem for graphs with bounded arboricity and we propose an algorithm which bases on the one introduced by Czygrinow, Hańćkowiak and Szymańska in [12].

1.3 The article organisation

In Section 2 we gather some properties of random graphs, which will be used in the analysis of the proposed algorithms. In Section 3 we introduce algorithms, which find a maximal independent set and a vertex colouring. Moreover we give there a note on the dominating set problem, which in $G(n,p)$ is closely related

to the MIS problem. Section 4 is devoted to the construction of a matching and an edge colouring in $G(n,p)$.

2 Useful properties of $G(n,p)$

In this section we state some useful results concerning $G(n,p)$. For a proof of the following lemma see for example the proof of Theorem 5.4(i) in [18].

Lemma 2.1. *Let $0 < C < 1$ be a constant and $p = p(n) \in [0;1]$ be such that $np \leq C$. Then with probability at least $1 - n^{-1}$ all connected components in $G(n,p)$ are of size at most $4\ln n/(1-C)^2$.*

Remark 2.2. *In the statement of Lemma 2.1 values $1-n^{-1}$ and $4\ln n/(1-C)^2$ may be replaced by $1 - Cp$ and $4\ln p^{-1}/(1-C)^2$, respectively.*

Interesting bounds on the arboricity of $G(n,p)$ and the relationship of arboricity with tree packing in $G(n,p)$ may be found in [15]. Obviously with high probability $np \leq a(G(n,p))$. We state an upper bound on $a(G(n,p))$, which is not included in [15]. Constants in bounds on $a(G(n,p))$ might be improved, but we state them so for simplicity.

Lemma 2.3. *Let $C = C(n)$ and $p = p(n) \in [0;1]$ be such that $np \leq C$.*

(i) If $C \leq 1/2$ then $a(G(n,p)) \leq 2$ with probability at least $1 - 2n^{-1}$.

(ii) If $C > 1/2$ then $a(G(n,p)) \leq 8C$ with probability at least $1 - 8Cn^{-1}$.

Remark 2.4. *In the statement of Lemma 2.3 value $1 - 2n^{-1}$ may be replaced by $1 - 7p$ and $1 - 8Cn^{-1}$ by $1 - 28p$.*

Proof. (i) If $C \leq 1/2$ then by classical results (see for example the proof of Theorem 5.5 in [18]) with probability at least $1-2n^{-1}$ $G(n,p)$ has no components with more than one cycle. Thus $a(G(n,p)) \leq 2$.

(ii) Now assume that $C > 1/2$. Let $t = \lfloor 2C+1 \rfloor$. Denote by $\bigcup_{i=1}^{t} G(n,1/(2n))$ a sum of t independent random graphs $G(n,1/(2n))$. Each edge appears in $\bigcup_{i=1}^{t} G(n,1/(2n))$ independently with probability $1-(1-(1/2n))^t = t/2n + O(1/n^2)$. Therefore, for n large enough, there exists a coupling of $G(n,p)$ and $\bigcup_{i=1}^{t} G(n,1/(2n))$, such that

$$G(n,p) \subseteq \bigcup_{i=1}^{t} G(n,1/(2n))$$

with probability 1. Thus, by the coupling and (i) we get that $G(n,p) \leq 2t$ with probability $1 - 2tn^{-1}$. □

Denote by $\mu(G)$ the size of a maximum matching in G. By [30] (see also [2]) we have the following.

Lemma 2.5. *Let $C > 0$ and $\varepsilon > 0$ be constants. If $np = C$, then with high probability*

$$\mu(G(n, C/n)) \geq (1 - \varepsilon/3)(1 - \beta(C))n$$

for $\beta(C) = \frac{a_1(C)+a_2(C)+a_1(C)a_2(C)}{2C}$, where $a_1 = a_1(C)$ is the smallest solution of the equation $x = Ce\exp(-Ce^{-x})$ and $Ce^{-a_2} = a_2$.

Denote by $\gamma(G)$ the size of the minimum dominating set in G. The following lemma presents a simple fact concerning the domination number $\gamma(G(n,p))$ of $G(n,p)$. Denote by $I_g(G)$ the independent set constructed by the greedy sequential algorithm in G (i.e. the algorithm, which adds one by one vertices to a constructed independent set if they are not neighbours of any vertex of the independent set constructed so far). Partially the following result is a corollary of the results presented in [16, 32]. We give here a short proof for completeness.

Lemma 2.6. *Let $p = p(n) = o(1)$ and $\gamma = \gamma(G(n,p))$ be the domination number of $G(n,p)$.*
(i) If $np \to \infty$ then with high probability

$$\gamma(G(n,p)) = (1 + o(1))\frac{\ln np}{p}$$

and $I_g(G(n,p))$ is a $(1+\varepsilon)$–approximation of a minimum dominating set in $G(n,p)$.
(ii) If $np = O(1)$ then with high probability $\gamma(G(n,p)) = \Theta(n)$ and $I_g(G(n,p))$ is a constant approximation of the minimum dominating set in $G(n,p)$.

Proof. We prove only (i). Note that any MIS is a dominating set. It is a folklore result that in this range of parameters with high probability $|I_g(G(n,p))| = (1 + o(1))\frac{\ln np}{p}$. We are left with showing that there are no asymptotically smaller dominating sets. Let $k = (\ln np - 3\ln\ln np)/p$ and X_k be the number of dominating sets of size k in $G(n,p)$ then

$$\mathbb{E}X_k = \binom{n}{k}\left[1 - (1-p)^k\right]^{n-k} \leq \exp\left(-(1 + o(1))k(\ln np)^2\right) = o(1),$$

which shows that there are no dominating sets of size at most k. □

3 Maximal independent set, vertex colouring and dominating set

The algorithms presented in this section use as a procedure an algorithm introduced by Elkin and Barbenboin in [3] (Algorithm 3 in [6]). We call it ArbColour(ε, a) as in [6] . Given $\varepsilon > 0$ and $a = a(G)$, ArbColour(ε, a) finds a legal colouring of a connected graph G with $\lfloor(2+\varepsilon)a\rfloor$ colours in $O(a\ln|V(G)|)$ time. Moreover ArbColour(ε, a) may be easily modified to give algorithm ArbMIS(ε, a), which using $O(a)$ additional rounds constructs a MIS basing

on the colouring constructed by ArbColour(ε, a). We denote the outputs of the algorithms ArbColour(1, 2) and ArbMIS(1, 2) on a graph G by ArbColour(G) and ArbMIS(G), respectively.

3.1 ID assignment

In the algorithms below, as an input we have a value r_v attributed to each vertex $v \in V$. Here we explain how assign r_v to each vertex. In many distributed networks, a natural assumption is that IDs of the vertices are subsequent numbers from the set $\{1, \ldots, n\}$ or are assigned in a uniform manner from a larger set of size polynomial with n. Moreover whether an edge appears or not does not depend on IDs of the ends of the edge. In this case we set r_v equal to ID of vertex v divided by the size of the set from which IDs are taken.

However in some settings the above assumptions might fail. Then we assume that each vertex v is assigned a number chosen independently of all other vertices, uniformly at random from $\{1, \ldots, N\}$ ($N \geq n$) with or without replacement. Note that in this case the presented algorithms become random. By r_v we denote the number chosen by v divided by N. In the proofs we consider the case in which r_v are random, because it is technically a little more complicated. However we stress that the proofs may be easily rewritten to the other case. In the case with random r_v we will need the following fact.

Fact 3.1. *Let $0 < \varepsilon < 1$, $Z := |\{v : r_v \in (a; b]\}|$, and let $a = a(n)$ and $b = b(n)$ be such that $0 < a < b < 1$ and $n(b-a) \to \infty$. Then $(1-\varepsilon)(b-a) \leq Z \leq (1+\varepsilon)(b-a)$ with probability at least $1 - 2e^{-\frac{\varepsilon^2 n(b-a)}{54}}$.*

Proof. In the interval $(aN; bN]$ there are at least $N(b-a) - 1$ and at most $N(b-a)+1$ natural numbers. Depending on whether we choose with replacement or not, Z has either the binomial distribution $\mathrm{Bin}\,(n, q)$ or the hypergeomeric distribution with parameters N, n and Nq, where

$$(b-a)(1-\varepsilon/3) \leq (b-a) - 1/N \leq q \leq (b-a) + (2/N) \leq (1+\varepsilon/3)(b-a)$$

and $\mathbb{E}|Z| \geq n(b-a)/2$. By Chernoff's inequality (see for example Theorems 2.1 and 2.10 in [18]) for any $0 < \varepsilon < 1$

$$\mathbb{P}\left(||Z| - \mathbb{E}|Z|| \geq \frac{\varepsilon}{3}\mathbb{E}|Z|\right) \leq 2e^{-\frac{\varepsilon^2 \mathbb{E}|Z|}{27}} \leq 2e^{-\frac{\varepsilon^2 n(b-a)}{54}}.$$

□

3.2 A note on the average degree np

In the following algorithms, one of the inputs is the value np. We should note here, that if $np = \ln n + c_n$ for some $c_n \to \infty$ (i.e. when with high probability $G(n,p)$ is connected), then with high probability the value np in $G(n,p)$ may be easily approximated by each vertex independently in $O(1)$ time. In this case each vertex needs just to calculate an average degree of several vertices in the

closest neighbourhood (i.e. at distance at most C from v, where C is a constant). The correctness of the procedure follows by the standard facts that in this range of p with high probability each vertex in $G(n,p)$ has many vertices in the closest neighbourhood and the degree of each vertex has the binomial distribution with parameters n and p. Therefore to prove that this procedure works it is enough to apply Chernoff's inequality to the sum of degrees of vertices at distance at most C from $v \in V$ and use the union bound over all $v \in V$. Exactness of the approximation depends on the choice of the value C.

3.3 MIS in $G(n,p)$

Given an integer n and $p \in (0,1)$. Let $K = \lceil 8 \ln(3np(e^{1/8}-1)+1) \rceil$,

$$c_k = \frac{e^{\frac{k-1}{8}}}{3np} \quad \text{for } k = 1,2,\ldots,K-1, \quad \text{and } c_K = 1 - \sum_{k=1}^{K-1} c_k.$$

Define values

$$s_0 = 0, \quad s_k = \sum_{i=1}^{k} c_i = \frac{1}{3np} \cdot \frac{e^{\frac{k}{8}}-1}{e^{\frac{1}{8}}-1} \quad \text{for } 1 \le k \le K-1, \quad \text{and } s_K = 1.$$

Algorithm 1. GnpMIS

---

input: G , np – an average degree in G, $\{r_v, v \in V(G)\}$.
output: an independent set I.

1. $I := \emptyset$.
2. For each v, if $s_{k-1} \le r_v < s_k$, then v adds itself to Z_k.
3. FOR k=1 TO K
   (a) $R_k := Z_k \setminus N_G(I)$;
   (b) Run ArbMIS(1,2) on $G[R_k]$;
   (c) $I := I \cup$ ArbMIS$(G[R_k])$.

Theorem 3.1. *Let $p = p(n) = o(1/\ln n)$. Then with high probability* GnpMIS *constructs a maximal independent set in $G(n,p)$ in $O\left(\ln(np) \cdot \ln\ln p^{-1}\right)$ rounds.*

The main idea of algorithm GnpMIS is to choose constants s_k so that with high probability for all k, $k = 1,\ldots,K$, $G[R_k]$ is a vary sparse random graph, namely with the average degree at most $1/2$. Then by classical results with high probability all connected components of $G[R_k]$ are of logarithmic size and have arboricity at most 2. ArbMIS(1,2) runs on each component of $G[R_k]$ independently, therefore the number of rounds necessary to construct ArbMIS(G[R$_k$]) is $O(\ln\ln p^{-1})$. The following lemma will be used on each $G[R_k]$.

Lemma 3.2. *Let $p = p(n) \in [0;1]$ and $n_0 = n_0(n) \to \infty$ be such that $n_0 p \leq 1/2$. Then with probability at least $1 - 8p$* ArbMIS(1,2) *constructs a maximal independent set in $G(n_0, p)$ in $O(\ln \ln p^{-1})$ rounds.*

Proof. We have $n_0 p \leq 1/2$ thus by Lemmas 2.1 and 2.3 with probability at least $1-p-7p$ all components of $G(n,p)$ are of size at most $16 \ln p^{-1}$ and $a(G(n_0,p)) \leq 2$. Therefore with probability at least $1 - 8p$ ArbMIS(1,2) constructs a MIS in each component in $O(2\ln(16 \ln p^{-1}))$ rounds. □

The following fact we will use to ensure that the set constructed in step 4(b) is not too small.

Fact 3.3. *Let $p = p(n) = o(1)$ and $n_0 = n_0(n) \to \infty$ be such that $\frac{e^{-\frac{1}{8}}}{4p} \leq n_0 \leq \frac{10}{24p}$ and let Y_0 be the number of isolated vertices in $G(n_0, p)$. Then there exists a constant $c_2 > 0$ such that with probability at least $1 - e^{-c_2/p}$*

$$\frac{n_0}{2} \leq Y_0 \leq n_0.$$

Proof. Let Y_1 be the number of edges in $G(n_0, p)$. Then Y_1 has the binomial distribution $\mathrm{Bin}\,(n_0, p)$ with the expected value $\mathbb{E}Y_1 = \binom{n_0}{2} p \leq \frac{5}{24} n_0$. Thus by Chernoff's inequality (see for example Theorem 2.1 in [18]) there exists a constant $c_2 > 0$ such that

$$\mathbb{P}\left(Y_1 > \frac{1}{4} n_0\right) \leq \mathbb{P}\left(Y_1 > \mathbb{E}Y_1 + \frac{n_0}{24}\right) \leq e^{-\frac{\frac{n_0^2}{24^2}}{2\mathbb{E}Y_1 + \frac{n_0}{36}}} \leq e^{-\frac{n_0}{256}} \leq e^{-\frac{c_2}{p}}.$$

Obviously $Y_0 \geq n_0 - 2Y_1$ thus the lemma follows. □

Finally recall that algorithm GnpMIS relies on the assignment of r_v. Let Z_k, $1 \leq k \leq K$, be defined as in GnpMIS. Recall that we present the proof for r_v which are assigned randomly. If in Fact 3.1 we substitute $\varepsilon = 1/8$ and use the union bound over all k, then we get that there exists a constant $c_1 > 0$ such that with probability at least $1 - 2Ke^{-c_1/p}$ for all $1 \leq k \leq K-1$

$$\frac{7}{24} e^{\frac{k-1}{8}} p^{-1} \leq |Z_k| \leq \frac{9}{24} e^{\frac{k-1}{8}} p^{-1} \quad \text{and} \quad |Z_K| \leq \frac{9}{24} e^{\frac{K-1}{8}} p^{-1}. \tag{1}$$

Proof of Theorem 3.1. Let Z_k and R_k be defined as in GnpMIS and let $X_k = |\mathrm{ArbMIS}(G[R_k])|$. Our aim is to prove that with probability $1 - o(1)$ for all $k = 1, \ldots, K$ in Step 3b ArbMIS(1,2) constructs a MIS in $G[R_k]$ in $O(\ln \ln p^{-1})$ rounds. Then the theorem follows.

We might think of the execution of GnpMIS as of the random process during which we construct (or equivalently discover) an instance of $G(n,p)$. For all $k = 0 \ldots, K-1$, in the $(k+1)$–st step of the process we first discover edges between Z_{k+1} and I constructed in k first steps (Step 3a of GnpMIS). This gives R_{k+1}. Then we discover edges between vertices of R_{k+1} and finally we run ArbMIS(1,2) on $G[R_{k+1}]$ (Step 3b of GnpMIS). Note that edges in $G[R_{k+1}]$

are independent of edges discovered until step k. Therefore in the $(k+1)$–st step of the process $G[R_{k+1}]$ may be considered a random graph on $|R_k|$ vertices with independent edges and with edge probability p. Thus if $|R_k|p \le 1/2$ we may apply Lemma 3.2.

Define a random variable $A_k = X_1 + X_2 + \ldots + X_k$ for $k = 1, \ldots, K-1$, where $X_i = |\textsc{ArbMIS}(G[R_i])|$, for $i = 1, \ldots, K$. Note that the value of A_k depends only on the first k steps of the above described process. Let $\mathcal{A}$ be the event that (1) holds. For each $k = 1, \ldots, K-1$ let $\mathcal{A}_k$ be the event that

$$\frac{k}{8p} \le A_k \le \frac{k}{2p} \text{ and } \textsc{ArbMIS} \text{ constructs a MIS in } G[R_k] \text{ in } O(\ln\ln p^{-1}) \text{ time.}$$

Finally let $\mathcal{A}_K$ be event that $\textsc{ArbMIS}(1,2)$ finds a MIS in $G[R_K]$ in $O(\ln\ln p^{-1})$ rounds.

We will prove that there exists $c > 0$ such that for all $k = 0, 1, \ldots, K-1$

$$\mathbb{P}\left(\mathcal{A}_{k+1} \middle| \bigcap_{i=1}^{k} \mathcal{A}_i \cap \mathcal{A}\right) = 1 - 2e^{-cp^{-1}} - 8p. \tag{2}$$

Here we set $\bigcap_{i=1}^{0} \mathcal{A}_i \cap \mathcal{A} = \mathcal{A}$.

Assume that $\mathcal{A}$ holds. Let $n_0 = |R_1| = |Z_1|$. Then $7/(24p) \le n_0 \le 9/(24p)$. Each edge in $G[R_1]$ appears independently with probability p thus by Fact 3.3 with probability at least $1 - e^{-c_2 p^{-1}}$ we have $n_0/2 \le |X_1| \le n_0$. Moreover by Lemma 3.2 $\textsc{ArbMIS}(1,2)$ constructs a MIS in $G[R_1]$ in $O(\ln\ln p^{-1})$ rounds with probability at least $1 - 8p$. Therefore

$$\mathbb{P}\left(\mathcal{A}_1 | \mathcal{A}\right) \ge 1 - e^{-c_2 p^{-1}} - 8p. \tag{3}$$

Assume now that $\bigcap_{i=1}^{k} \mathcal{A}_i \cap \mathcal{A}$ holds for some $1 \le k \le K-2$. Note that, whether $\bigcap_{i=1}^{k} \mathcal{A}_i \cap \mathcal{A}$ occurs, depends only on the first k steps of the above defied process. Let B_k be such that

$$A_k = \frac{k}{8p} + \frac{B_k}{p} \text{ for } 1 \le k \le K-2.$$

First consider the case $0 \le B_k \le \frac{1}{8}$. Given values B_k and $|Z_{k+1}|$ (such that (1) holds), the random variable $|R_{k+1}|$ has the binomial distribution $\mathrm{Bin}\left(|Z_{k+1}|, (1-p)^{A_k}\right)$ with

$$\frac{7}{24}\frac{e^{-B_k(1+o(1))}}{p} \le \mathbb{E}|R_{k+1}| \le \frac{9}{24}\frac{e^{-B_k}}{p}.$$

Thus by Chernoff's inequality for some $c_3 > 0$ with probability at least $1 - e^{-c_3 p^{-1}}$

$$\frac{1}{4}\frac{e^{-B_k}}{p} \le |R_{k+1}| \le \frac{10}{24}\frac{e^{-B_k}}{p}. \tag{4}$$

Recall that $G[R_{k+1}]$ is a random graph with $|R_{k+1}|$ edges and edge probability p. Under condition (4) by Lemma 3.2 with probability at least $1-8p$ ArbMIS(1,2) finds a MIS in $G[R_{k+1}]$ in $O(\ln\ln p^{-1})$ rounds. Moreover $0 \le B_k \le \frac{1}{8}$ thus under condition that (4) holds by Fact 3.3 with probability at least $1 - e^{-c_2 p^{-1}}$

$$\frac{e^{-B_k}}{8p} \le X_{k+1} \le \frac{10}{24}\frac{e^{-B_k}}{p} \le \frac{1}{2p}.$$

Therefore if $\bigcap_{i=1}^{k} \mathcal{A}_i \cap \mathcal{A}$ and $0 \le B_k \le \frac{1}{8}$ hold them with probability at least $1 - e^{-c_2 p^{-1}} - e^{-c_3 p^{-1}} - 8p$

$$\frac{k+1}{8p} \le \frac{k}{8p} + \frac{B_k}{p} + \frac{e^{-B_k}}{8p} \le A_{k+1} \le \frac{k+1}{2p}$$

and ArbMIS(1,2) finds a MIS in $G[R_{k+1}]$ in $O(\ln\ln p^{-1})$ rounds.

Now let $1/8 \le B_k \le 7k/8$. Then

$$\mathbb{E}|R_{k+1}| \le \frac{10}{24p} e^{-\frac{1}{8}}.$$

Thus by Chernoff's inequality there exists $c_4 > 0$ such that with probability at least $1 - e^{-c_4 p^{-1}}$ $|R_{k+1}| \le 1/2p$. Thus

$$\frac{k+1}{8p} \le \frac{k}{8p} + \frac{B_k}{p} \le A_{k+1} \le A_k + |R_{k+1}| \le \frac{k+1}{2p}.$$

So after applying Lemma 3.2 to $G[R_{k+1}]$ we get that if $\bigcap_{i=1}^{k} \mathcal{A}_i \cap \mathcal{A}$ and $1/8 \le B_k \le 7k/8$ hold then $\mathcal{A}_{k+1}$ occur with probability at least $1 - e^{-c_4 p^{-1}} - 8p$.

Now assume that $\bigcap_{i=1}^{K-1} \mathcal{A}_i \cap \mathcal{A}$ occurs. Then by Chernoff's inequality and Lemma 3.2 there exists $c_5 > 0$ such that with probability at least $1-e^{-c_5 p^{-1}}-8p$. $|R_K| \le 1/2p$ and ArbMIS(1,2) finds a MIS in $G[R_K]$ in $O(\ln\ln p^{-1})$ time.

Finally if we take $0 < c < \min\{c_1, c_2, c_3, c_4, c_5\}$, then in all the above–stated cases we have (2). Therefore by (1) and the chain rule

$$\mathbb{P}\left(\bigcap_{i=1}^{K} \mathcal{A}_i \cap \mathcal{A}\right) \ge 1 - 4Ke^{-cp^{-1}} - 8Kp = 1 - o(1),$$

which finises the proof of Theorem 3.1. □

3.4 Vertex colouring in $G(n,p)$

Recall that ArbColour(ε, a) introduced by Elkin and Barbenboin [3] finds a colouring of a connected graph G with $\lfloor(2+\varepsilon)a(G)\rfloor$ colours in $O(a(G)\ln|V(G)|)$ time.

Algorithm 2. GnpColour

input: G , np – an average degree in G, $\{r_v, v \in V(G)\}$.
output: a vertex colouring of G.

1. $K := \lceil 3np \rceil$
2. For each v, if $\frac{k-1}{3np} \leq r_v < \frac{k}{3np}$, then v adds itself to the set Z_k, $k = 1, \ldots, K$.
3. Run ArbColour(1,2) on each $G[Z_k]$, using palette $\{6(k-1), \ldots, 6k\}$.

Theorem 3.2. *Let $p = p(n) = o(n^{-1/2})$. Algorithm* GnpColour *with high probability finds a colouring of $G(n,p)$ with $18np = O(a(G(n,p)))$ colours in $O(\ln \ln p^{-1})$ rounds.*

Proof. By Fact 3.1 and the union bound there exists a constant $c > 0$ such that with probability at least $1 - Ke^{-c/p}$ we have $|Z_k| \leq 1/2p$ for all $k = 1, \ldots, K$. Moreover, given sets Z_k, $k = 1, \ldots, K$, $G[Z_k]$ are independent random graphs with edge probability p. Therefore by the union bound, Lemmas 2.1 and 2.3 with probability at least $1 - 12Kp$ for all $k = 1, \ldots, K$ ArbColour(2,1) constructs a legal colouring of $G[Z_k]$ with 6 colours in $O(\ln \ln p^{-1})$ rounds. Thus all those colourings together give a legal colouring of $G(n,p)$ with probability $1 - O(np^2 + npe^{-cp}) = 1 - o(1)$. □

3.5 A note on the dominating set problem

Let $np \to \infty$ but $p = o(1)$. We recall that any MIS is as well a dominating set. By classical results on random graphs [9, 14], if $p = o(1)$ and $np \to \infty$ then with high probability the size of the maximum independent set in $G(n,p)$ is $(1 + o(1))2 \ln np/p$. This combined with Lemma 2.6 and Theorem 3.1 gives the following result.

Theorem 3.3. *Let $\varepsilon > 0$ be a constant and $p = p(n)$ be such that $np \to \infty$ and $p = o(1)$. Then Algorithm* GnpMIS *with high probability constructs a $(2 + \varepsilon)$–approximation of a minimum dominating set in $G(n,p)$.*

We should mention that there is no known sequential polynomial time algorithm, which with high probability constructs an independent set of size $(1 + \delta) \ln np/p$ for some constant $\delta > 0$ (see for example discussion concerning colouring in [13]). Therefore we may conjecture that in fact GnpMIS constructs a $(1 + \varepsilon)$–approximation of a minimum dominating set in $G(n,p)$.

4 Matchings and edge colouring

In this section we introduce an algorithm for edge colouring and two algorithms, which construct matchings in $G(n,p)$. In all of them we use a choice of a random subgraphs of $G(n,p)$, therefore the algorithms presented in this section are random.

4.1 Maximal matching and edge colouring

First we present algorithm GnpEdgeColour, which resolves the problem of edge colouring in $G(n,p)$. GnpEdgeColour may easily be rewritten to give algorithm GnpMaximalMatching constructing a maximal matching. GnpEdgeColour relies on the same heuristic as the edge colouring algorithm in [27] (see also Algorithm 14 in [6]). In the presented algorithm first we divide $G(n,p)$ into $\lceil np \rceil + 1$ random graphs with components of size $O(\ln n)$ and arboricity 2. Then on each of those graphs we run algorithm ForestDecomposition(a,ε) developed by Barenboim and Elkin in [3] (see also Algorithm 2 in [6]), which partition a connected graph of arboricity a into $\lfloor (2+\varepsilon)a \rfloor$ oriented forests with roots in $O(a \ln |V(G)|)$ rounds. Using this decomposition we construct a vertex colouring of each forest separately using classical algorithm of Cole and Vishkin (see [11, 17]), called here TreeColour, which in $O(\log^* n)$ rounds constructs a 3–colouring of any rooted tree. This colourings combined with orientations of edges given by the forest decomposition allow to divide the edge set into $O(np)$ sets consisting of disjoint stars and colour the stars in parallel.

Algorithm 3. GnpEdgeColour

---

input: graph G, average degree np in G
output: a colouring of edges $\varphi : E(G) \to \{1, \ldots, 2\Delta - 1\}$

1. $E_i = \emptyset$ for all $i = 1, \ldots, \lceil 2np \rceil$
2. $l_v := \emptyset$ for all $v \in V(G)$
3. For each $e \in E(G)$, e adds itself to E_i with probability $1/2np$, independently for all $i = 1, \ldots, \lceil 2np \rceil$.
4. $G_i := (V(G), E_i \setminus \bigcup_{j=1}^{i-1} E_j)$, $G_0 := (V(G), E(G) \setminus \bigcup_{j=1}^{\lceil 2np \rceil} E_j)$.
5. Run in parallel ForestDecomposition(2,1) on each component of G_i, $i = 0, \ldots, \lceil 2np \rceil$ to get forest decompositions $(F_{i,0}, \ldots, F_{i,5})$
6. Run in parallel TreeColour on each component of $F_{i,j}$ with the palette $\{18i + 6j + 1, 18i + 6j + 2, 18i + 6j + 3\}$, $i = 0, \ldots, \lceil 2np \rceil$, $j = 0, \ldots, 5$
7. FOR $i = 0$ TO $\lceil 2np \rceil$, FOR $j = 0$ TO 5, FOR $k = 1$ TO 3
   If v has colour $18i + 6j + k$ then
   (a) for each child w of v in $F_{i,j}$ one by one
      - v sets $\varphi(\{v, w\})$: = available colour from $\{1, \ldots, 2\Delta - 1\} \setminus (l_v \cup l_w)$
      - $l_v := l_v \cup \{\varphi(v, w)\}$

   (b) v sends $\varphi(v, w)$ to each child w
   (c) v sends l_v to all neighbours in G
   (d) all children w of v in parallel set $l_w := l_w \cup \{\varphi(v, w)\}$ and send l_w to all neighbours

Theorem 4.1. *Let $p = p(n) \in (0,1)$ be such that $np = o(1)$. Then with high probability algorithm* GnpEdgeColour *finds in $G(n,p)$ an edge colouring with $2\Delta - 1$ colours in $O(\ln\ln n + np)$ synchronous rounds.*

Proof. Let $G'_0 = G_0$ and $G'_i = (V(G), E_i)$, for $i = 1, \ldots, \lceil 2np \rceil$. Note that each G'_i is a random graph $G(n, p_i)$, where $p_i \leq 1/2n$, however $G'_0, \ldots, G'_{\lceil 2np \rceil}$ are not independent. By Lemmas 2.1 and 2.3 each G'_i with probability at least $1 - 3n^{-1}$ is a graph with components of size at most $16 \ln n$ and arboricity at most 2. Thus by the union bound the probability that all G'_i, $i = 0, \ldots, \lceil 2np \rceil$, have all components of size at most $16 \ln n$ and arboricity at most 2 is $1 - O(p)$.

Therefore with high probability ForestDecomposition(2,1) performs properly on each G_i (since $G_i \subseteq G'_i$). Moreover, in any oriented forest $F_{i,j}$, for any two vertices v and v' of the same colour prescribed by TreeColour and any children w and w' of v and v', respectively, edges $\{v, w\}$ and $\{v', w'\}$ are disjoint. Therefore in Step 7a all intersecting edges get distinct colours. Therefore the constructed edge colouring is proper. Moreover each edge intersects at most $2\Delta - 2$ other edges, thus there are enough colours to use in Step 7a.

Finally with high probability Step 5 takes $O(\ln\ln n)$ rounds (since with high probability all G_i have components of logarithmic size), Step 6 takes $O(\log^* n)$ time and each iteration of Step 7 takes $O(1)$ rounds. Therefore all together this gives $O(\ln\ln n + np)$ rounds. □

It is easy to modify GnpEdgeColour to get algorithm GnpMaximalMatching. In GnpMaximalMatching instead of φ the output is a set M. We only need to rewrite steps 2 and 7 in the following way.

Algorithm 4. GnpMaximalMatching

2'. $l_v := 0$ for all $v \in V(G)$

7'. FOR $i = 0$ TO $\lceil 2np \rceil$, FOR $j = 0$ TO 5, FOR $k = 1$ TO 3
If v has colour $18i + 6j + k$, $l_v = 0$ and there is a child w of v in F_j with $l_w = 0$ then

(a) $M := M \cup \{v, w\}$

(b) $l_v := 1$, $l_w := 1$

(c) v and w send l_v and l_w to their neighbours

The proof of the following theorem is analogous to the proof of Theorem 4.1.

Theorem 4.2. *Let $p = p(n) \in (0,1)$ be such that $np = o(1)$. Then with high probability algorithm* GnpMaximalMatching *finds a maximal matching in $G(n,p)$ in $O(\ln\ln n + np)$ synchronous rounds.*

4.2 $(1+\varepsilon)$–approximation of a maximum matching

Recall that by $\mu(G)$ we denote the size of a maximum matching in G. In [12] Czygrinow et al. proposed an algorithm, which finds a matching of size at least $(1-\varepsilon)\mu(G)$ in $O(\log^* n)$ rounds in any graph with bounded arboricity. We call this algorithm ArbMatching. Lemma 2.3 implies the following theorem.

Theorem 4.3. *Let $p = p(n) \in (0,1)$ and $C > 0$ be a constant. If $np \leq C$ then with high probability* ArbMatching *finds in $G(n,p)$ a matching of size at least $(1-\varepsilon)\mu(G(n,p))$ in $O(\log^* n)$ rounds.*

Corollary 4.4. *For every $\varepsilon > 0$ there exists $C = C(\varepsilon)$ such that with high probability* ArbMatching *constructs in $G(n, C/n)$ a matching of size at least $(1-\varepsilon)n/2$ in $O(\log^* n)$ rounds.*

Proof. Let $\varepsilon > 0$ and $\beta(C)$ be defined as in Lemma 2.5. Note that $\beta(C)$ tend to $1/2$ as $C \to \infty$. Thus there exists $C = C(\varepsilon)$ such that with high probability $\mu(G(n, C/n)) \geq (1 - 2\varepsilon/3)n/2$. Thus by Theorem 4.3 with high probability ArbMatching finds in $G(n, C/n)$ a matching of size at least $(1-\varepsilon)n/2$ in $O(\log^* n)$ rounds. □

The following algorithm finds a $(1-\varepsilon)$–approximation of a maximum matching in $G(n,p)$ with $np \to \infty$.

Algorithm 4. GnpMatching

---

input: G, np – an average degree of G, $\varepsilon > 0$ – a constant
output: a matching M

1. Get G' by independently leaving each edge with probability $C(\varepsilon)/(np)$, where $C(\varepsilon)$ is as in the proof of Corollary 4.4
2. Run ArbMatching on G'.

Theorem 4.5. *Let $p = p(n) \in (0,1)$ be such that $np \to \infty$ and let $\varepsilon > 0$ be a constant. Then with high probability algorithm* GnpMatching *finds in $G(n,p)$ a matching of size at least $(1-\varepsilon)n/2$ in $O(\log^* n)$ synchronous rounds.*

Note that $(1-\varepsilon)n/2$ is with high probability a $(1-\varepsilon)$–approximation of a maximum matching in $G(n,p)$ with $np \to \infty$.

Proof. Each edge appears in G' independently with probability $C(\varepsilon)/n$. Therefore G' is in fact $G(n, C(\varepsilon)/n)$ and the theorem follows by Corollary 4.4. □

5 Conclusions

We have presented several algorithms, which solve classical problems of the distributed computing theory for $G(n,p)$. We have shown that statistical properties of random graphs allow to develop efficient algorithms, which construct a maximal independent set, a $(2+\varepsilon)$–approximation of a minimum dominating set, a maximal matching, a $(1-\varepsilon)$–approximation of a maximum matching, a vertex colouring and an edge colouring.

We believe that this article shows some general tools for designing distributed algorithms for $G(n,p)$. Moreover it gives rise to many other questions. For example, it would be interesting to find distributed algorithms for other random graph models, especially those, which might depict the structure of some networks. Moreover one could ask about lower bounds on the number of rounds needed to solve problems on $G(n,p)$ in the distributed models. In addition the LOCAL model considered here do not have any constraints on the size of the messages nor on the amount of local computations. Taking into account some additional limitations should inspire some new interesting questions.

References

[1] N. Alon, L. Babai, and A. A. Itai. A fast and simple randomized parallel algorithm for the maximal independent set problem. *Journal of Algorithms,*, 7(4):567–583, 1986.

[2] J. Aronson, A. Frieze, and B. Pittel. Maximum matchings in sparse random graphs: Karp-Sipser revisited. *Random Structures and Algorithms*, 12:111–177, 1998.

[3] L. Barenboim and M. Elkin. Sublogarithmic distributed MIS algorithm for sparse graphs using Nash-Williams decomposition,. *Distributed Computing*, 22(special issue of PODC'08):363–379, 2010.

[4] L. Barenboim and M. Elkin. Distributed deterministic vertex coloring in polylogarithmic time. *Journal of ACM*, 58(5):23, 2011.

[5] L. Barenboim and M. Elkin. Distributed deterministic edge coloring using bounded neighborhood independence. *Distributed Computing*, 26(5-6):273–287, 2013.

[6] L. Barenboim and M. Elkin. *Distributed Graph Coloring: Fundamentals and Recent Developments*. Synthesis Lectures on Distributed Computing Theory. Morgan §Claypool, 2013.

[7] L. Barenboim, M. Elkin, S. Pettie, and J. Schneider. The locality of distributed symmetry breaking. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pages 321–330, Oct 2012.

[8] B. Bollobás. The diameter of random graphs. *Transactions of the American Mathematical Society*, 267(1):285–288, 1981.

[9] B. Bollobás and P. Erdős. Cliques in random graphs. *Math. Proc. Camb. Phil. Soc.*, 1976:419–427, 1976.

[10] F. Chung and L. Lu. The diameter of sparse random graphs. *Adv. in Appl. Math.*, 26(4):257–279, 2001.

[11] R. Cole and U. Vishkin. Deterministic coin tossing with applications to optimal parallel list ranking. *Information and Control*, 70:32–53, 1986.

[12] A. Czygrinow, M Hańćkowiak, and E. Szymańska. Fast distributed alogrithm for the maximum matching problem in bounded arboricity graphs. *Algorithms and Computation: 20th International Symp. ISAAC 2009, LNCS*, 5878:668–678, 2009.

[13] A. Frieze and C. McDiarmid. Algorithmic theory of random graphs. *Random Structures and Algorithms*, 10(1-2):5–42, 1997.

[14] A. M. Frieze. On the independence number of random graphs. *Discrete Mathematics*, 81:171–175, 1990.

[15] P. Gao, X. Pérez-Giménez, and C. M. Sato. Arboricity and spanning-tree packing in random graphs with an application to load balancing. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '14, pages 317–326. SIAM, 2014.

[16] R. Glebov, A. Liebenau, and T. Szab. On the concentration of the domination number of the random graph. arXiv:1209.3115.

[17] A. Goldberg, S. Plotkin, and G. Shannon. Parallel symetry–breaking in sparse graphs. *SIAM Journal on Discrete Mathematics*, 1:434–446, 1988.

[18] S. Janson, T. Łuczak, and A. Ruciński. *Random Graphs*. Wiley, 2001.

[19] L. Jia, R. Rajaraman, and T. Suel. An efficient distributed algorithm for constructing small dominating sets. *Distrib. Comput.*, 15(4):193–205, December 2002.

[20] V. Klee and D. Larman. Diameters of random graphs. *Canad. J. Math.*, 33:618–640, 1981.

[21] K. Kothapalli, C. Scheideler, M. Onus, and Schindelhauer C. Distributed coloring in $o(\sqrt{\log n})$ bit rounds. *Proc. of the 20th International Parallel and Distributed Processing Symposium*, 2006.

[22] F. Kuhn and R. Wattenhofer. Constant-time distributed dominating set approximation. In *In Proc. of the 22 nd ACM Symposium on the Principles of Distributed Computing (PODC*, pages 25–32, 2003.

[23] C. Lenzen and R. Wattenhofer. Minimum dominating set approximation in graphs of bounded arboricity. In NancyA. Lynch and AlexanderA. Shvartsman, editors, *Distributed Computing*, volume 6343 of *Lecture Notes in Computer Science*, pages 510–524. Springer Berlin Heidelberg, 2010.

[24] E. Levy, G. Louchard, and J. Petit. A distributed algorithm to find hamiltonian cycles in G(n,p) random graphs. *In Proc. Workshop on Combinatorial and Algorithmic Aspects of Networking (CAAN) 2004, Springer Lecture Notes in Computer Science*, 3405:63–74, 2005.

[25] M. Luby. A simple parallel algorithm for the maximal independent set problem. *SIAM Journal on Computing*, 15:1036–1053, 1986.

[26] M. Michal Hańćkowiak, M. Karoński, and A. Alessandro Panconesi. On the distributed complexity of computing maximal matchings. *SIAM Journal on Discrete Mathematics*, 15:41–57, 2001.

[27] A. Panconesi and R. Rizzi. Some simple distributed algorithms for sparse networks. *Distributed Computing*, 14(2):97–100, 2001.

[28] A. Panconesi and A. Srinivasan. On the complexity of distributed network decomposition. *Journal of Algorithms*, 20:356–374, 1996.

[29] D. Peleg. *Distributed computing: a locality-sensitive approach.* Society for Industrial and Applied Mathematics,, Philadelphia, PA, USA, 2000.

[30] B. Pittel. On tree census and the giant component in sparse random graphs. *Random Structures and Algorithms*, 1:311–342, 1990.

[31] O. Riordan and N. Wormald. The diameter of sparse random graphs. *Combinatorics, Probability and Computing*, 19(5–6):835–926, 2010.

[32] B. Wieland and Godbole A. P. On the domination number of a random graph. *the Electronic Journal of Combinatorics*, 8:R37, 2001.