Properties of Random Graphs via Boltzmann Samplers

Konstantinos Panagiotou[†] and Andreas Weißl

Institute of Theoretical Computer Science, ETH Zurich, Universitätsstr. 6, CH-8092 Zurich, Switzerland

This work is devoted to the understanding of properties of random graphs from graph classes with structural constraints. We propose a method that is based on the analysis of the behaviour of Boltzmann sampler algorithms, and may be used to obtain precise estimates for the maximum degree and maximum size of a biconnected block of a "typical" member of the class in question. We illustrate how our method works on several graph classes, namely dissections and triangulations of convex polygons, embedded trees, and block and cactus graphs.

Keywords: Random graphs with structural constraints, Generating functions, Sampling algorithms

1 Introduction

In this paper we introduce a new method, which we apply to prove that graphs on n nodes, drawn uniformly at random from a graph class with structural constraints, have specific properties with high probability (w.h.p., i.e., with probability tending to 1 as $n \to \infty$). In order to investigate properties of such random graphs, we have to cope with the difficulty of the dependence of the edges. Such graph classes, however, are important when modeling real-world scenarios, and developing and analysing (the average performance of) algorithms, which take inputs from those classes. Popular examples in this area are trees, series-parallel, outerplanar, and planar graphs, and models for the world-wide-web, which have been studied extensively over the last few years.

In the literature there are two common approaches, which can be used to prove that a random member of such a graph class has a specific property. Central to both is the *enumeration* of the objects in the class under consideration. The first approach is purely combinatorial. McDiarmid, Steger, and Welsh were the first to apply this method to obtain results about random planar graphs [MSW05], and later generalised their results to other classes of graphs [MSW06]. Although the approach is quite powerful, it has some disadvantages: one might have to adapt it already for slightly different graph classes, see e.g. [GMSW05], and sometimes it only yields rough lower and/or upper bounds (see below). Furthermore, it seems to be difficult to apply this method to estimate certain parameters, such as the expected number of edges.

The second approach is mainly based on analytic combinatorics and uses generating functions [FS05]. It has recently led to immense progress in the enumeration and the understanding of several properties (such as the distribution of the number of edges and the number of (bi-)connected components) of constrained graph classes, as for instance planar, series parallel, and outerplanar graphs [GN05, BGKN05]. However, it seems to be inherently difficult to investigate properties of those classes, which cannot be directly addressed with (a finite number of) parameters of the generating functions, such as the maximum node degree or the degree sequence.

In this context, there are two remarkable papers which provide solutions for special classes of graphs and parameters. Gao and Wormald show in [GW00] that the distribution of the maximum node degree in random triangulations of convex polygons with n nodes is essentially distributed as the maximum of n independent identically distributed random variables G_2, where G_2 is the sum of two independent geometric variables with mean 2. Moreover, a similar result is shown to hold for general random planar maps.

[†] This work was partially supported by the SNF, grant no. 200021-107880/1.

1365–8050 © 2007 Discrete Mathematics and Theoretical Computer Science (DMTCS), Nancy, France

In [Gou98] Gourdon presents a unified analytic framework dedicated to the estimation of the size of the largest component in random combinatorial structures. This framework may be used for instance to estimate the size of a maximum biconnected block in random graphs. Unfortunately, the framework does not cover all cases that might appear, as we demonstrate in the results section below.

Here we propose an alternative method, which is orthogonal to the approaches described above and can support us in understanding properties of random graphs. Our approach does not yield the precise limiting distribution of the parameters in question, but it has the big advantage that it provides us with very precise (and we believe tight) bounds for the *tail probabilities*, i.e., for the probability that a parameter deviates much from its expectation, even if these probabilities become very small. Such bounds are very important if we want to analyse algorithms from an average-case point of view.

Our approach is based on the analysis of the behaviour of *Boltzmann sampler* algorithms. The framework of Boltzmann samplers was developed by Duchon, Flajolet, Louchard and Schaeffer [DFLS04], and enabled the design of very efficient and but surprisingly simple algorithms – in practice and in theory – for sampling objects u.a.r. from several classes of combinatorial objects. As an additional advantage, these samplers can be derived systematically. Here, our central idea is to *analyse the execution* of a Boltzmann sampler for the graph classes in question. If we knew what properties the "typical" output of such an algorithm has, then a random member of the considered graph class would also have those properties. On the other hand, from a technical point of view, the analysis may become much easier if we reconceive, as just described.

Let us briefly compare the combinatorial, the generating function, and our new approach on a simple example, namely Cayley trees. With the combinatorial approach we easily obtain the exact number $t_n := n^{n-2}$ of labelled trees on n nodes. The method relies on a bijection between trees on n nodes and the encoding of a tree as a string – the well-known Prüfer code. Moon [Moo68] used this to obtain precise asymptotic bounds for the maximum node degree in a random tree. Although it works quite well for trees, the problem-tailored approach yields difficulties (and often weaker results), as soon as the complexity of the graph class in consideration increases. If we use generating functions, we obtain by singularity analysis an asymptotic estimate of the form $t_n \sim n! \left(\frac{e^n}{\sqrt{2\pi n^5}} + O(n^{-7/2}) \right)$. This approach has the advantage that it works for more complex graph classes as well, and that it might be used to obtain results about the expected number of edges, for instance (of course, for graph classes different from trees). On the other hand, as soon as we are interested in parameters like the maximum degree of the number of nodes of a given fixed degree, especially in more complex graph classes, it becomes much harder to introduce the right parameters in the generating functions and to analyse them. Contrary, our new method via Boltzmann samplers delivers the precise asymptotic behaviour of the maximum node degree, as well as the complete degree sequence for Cayley trees; moreover, it has the benefit that it generalises easily and mechanically to moderately complex graph classes, as we will demonstrate in this paper. Our method yields precise estimates – that is the *precise* asymptotic behaviour – for the maximum node degree and maximum size of a biconnected block of a random graph within the class under consideration. We are confident that the method is applicable to other parameters, and graph classes as well. This will be topic of further study.

Here we will study and illustrate how our method works on several graph classes, namely on block graphs $\mathcal{B}$, cactus graphs $\mathcal{C}$, dissections $\mathcal{D}$ and triangulations $\mathcal{T}$ of convex polygons, and embedded trees $\mathcal{E}$. All graphs we consider are simple, labelled, and connected. A *block graph* is a graph whose maximal biconnected blocks are cliques. A *cactus graph* is a graph in which each edge is contained in at most one cycle. A *dissection* is a biconnected outerplanar graph. All these graphs are well-known and well-studied combinatorial objects [FS05, DFLS04, FN99], which have plenty of applications in physics, computer science, and bioinformatics (see e.g. [BHM02, Fle99]).

Our results We first derive for every graph class $\mathcal{G} \in \{\mathcal{C}, \mathcal{B}, \mathcal{D}, \mathcal{E}, \mathcal{T}\}$ an efficient sampler based on the Boltzmann sampler framework, which was recently introduced by Duchon et al. in [DFLS04]. They will be crucial for our further analysis.

Theorem 1.1 *Let $n \in \mathbb{N}$, $0 < \varepsilon < 1$, and $\mathcal{G} \in \{\mathcal{C}, \mathcal{B}, \mathcal{D}, \mathcal{E}, \mathcal{T}\}$. There exists a randomised algorithm with expected running time $\mathcal{O}(n\varepsilon^{-1})$, which outputs random graphs of $\mathcal{G}$ with size in the interval $I_\varepsilon = \big(n(1-\varepsilon), n(1+\varepsilon)\big)$, such that the distribution of graphs from $\mathcal{G}$ is uniform on each size $k \in I_\varepsilon$. The*

Graph Class	Maximum Degree	Maximum biconnected Block
Cactus graphs $\mathcal{C}$	$m_{\mathcal{C}}(n) := 2\frac{\log n}{\log\log n}$	$s_{\mathcal{C}}(n) := \log_{C_1} n$
Block graphs $\mathcal{B}$	$m_{\mathcal{B}}(n) := \frac{\log n}{\log^{(3)} n}$	$s_{\mathcal{B}}(n) := \frac{\log n}{\log\log n}$
Embedded Trees $\mathcal{E}$	$m_{\mathcal{E}}(n) := \log_2 n$	n
Dissections $\mathcal{D}$	$m_{\mathcal{D}}(n) := \log_{C_2} n$	n
Triangulations $\mathcal{T}$	$m_{\mathcal{T}}(n) := \log_2 n$	n

Tab. 1: Maximum degree and maximum biconnected block for the considered graph classes. The constants $C_1 \approx 2.19$ and $C_2 \approx 2.38$ are analytically given.

variance of the running time is $\mathcal{O}(n^2\varepsilon^{-2})$.

Our main results on the structure of graphs drawn uniformly at random from $\mathcal{G}_n$ (i.e. graphs from $\mathcal{G}$, restricted to precisely n nodes) are summarised in the following theorem. Unless stated otherwise, logarithms are to the base e.

Theorem 1.2 *Let $\mathcal{G}$ be a graph class and denote by G_n a graph drawn u.a.r. from $\mathcal{G}_n$. Let $m_{\mathcal{G}}$ and $s_{\mathcal{G}}$ be as in Table 1. For every $\varepsilon > 0$, the following statements hold for almost all n.*

- *For $\mathcal{G} \in \{\mathcal{C}, \mathcal{B}, \mathcal{D}, \mathcal{E}, \mathcal{T}\}$ the maximum node degree of G_n is with probability $1 - n^{-\Omega(\varepsilon)}$ in the interval $\big((1-\varepsilon)m_{\mathcal{G}}(n), (1+\varepsilon)m_{\mathcal{G}}(n)\big)$.*
- *For $\mathcal{G} \in \{\mathcal{C}, \mathcal{B}\}$ the size of a largest biconnected block in G_n is with probability $1 - n^{-\Omega(\varepsilon)}$ in the interval $\big((1-\varepsilon)s_{\mathcal{G}}(n), (1+\varepsilon)s_{\mathcal{G}}(n)\big)$.*

Observe that in both cases we obtain the *precise asymptotic behaviour* of the parameters in question. It seems that as soon as we consider more complex objects than trees, some of the involved constants become quite unpredictable and it might be very difficult – if not impossible – to obtain them by a purely combinatorial approach. Note that the result about random triangulations was already known (it was first proved by Devroye, Flajolet, Hurtado, Noy, and Steiger in [DFH$^+$99], and then made more precise by the previously mentioned paper [GW00]), and we simply reprove it with our method. Moreover, the result about the maximum size of a biconnected block in random cactus graphs can be proved by the more general statements in [Gou98], but the similar statement for block graphs *cannot* (as the general case of the special combination of generating functions involved in the description of the class of block graphs is not covered there). Finally, note that block and cactus graphs are examples of graph classes, where the general lower bound for the maximum node degree of $(1+o(1))\frac{\log n}{\log\log n}$ for addable graph classes, given in [MSW06, Corollary 5.3], is not tight.

Roughly speaking, we proceed as follows to prove the above theorem: instead of investigating directly the properties of the graph class, we consider the execution of its corresponding Boltzmann sampler. We examine how the *shape* of a sampled object evolves over a run of the sampler and how this affects the related property. From this knowledge, we can eventually deduce the probability that a sampled output object has the property in interest.

As already mentioned, enumerating the objects in the graph class in question is important for understanding its structural properties. Here we exploit generating function techniques, which were only recently applied to obtain similar results for planar, series parallel, and outerplanar graphs [BGKN05, GN05], and are well described in the forthcoming book "Analytic Combinatorics" by Flajolet and Sedgewick [FS05].

Outline Due to space constraints, in this extended abstract we focus solely on the maximum degree of random block graphs, and demonstrate how we can determine it by analysing the execution of a Boltzmann sampler for this graph class. The proof is given in (almost) full detail. The proofs of the other results can be proved analogously by applying a similar proof pattern, and all the details will be given in the full version of the paper.

This paper is structured as follows. In Section 2 we describe briefly a few facts about block graphs and introduce necessary notation. Section 3 presents our sampling algorithm in detail, as it is crucial for our analysis. In Section 4 we demonstrate our method, where we prove tight bounds for the maximum degree of random block graphs.

2 Decomposition & Generating Functions

Before we proceed, let us introduce some necessary notation, which we are going to use in the remainder of the paper. Let $\mathcal{G}$ be a class of graphs and denote by $\mathcal{G}_n$ the set of all graphs in $\mathcal{G}$ on n nodes, and let $G(x) = \sum_n g_n \frac{x^n}{n!}$ denote its exponential generating function (egf), where $g_n := |\mathcal{G}_n|$. In the following, we will frequently use the *pointing* and *derivative* operators. Given $\mathcal{G}$, we define $\mathcal{G}^\bullet$ as the class of *pointed* (or *rooted*) graphs, where a node is distinguished from all other nodes. The *derivated* class $\mathcal{G}'_n$ is obtained by removing the label n from every object of $\mathcal{G}_n$, such that the obtained objects have size $n-1$, i.e., node n can be considered as a distinguished node that does not contribute to the size. We set $\mathcal{G}' := \bigcup_{n\geq 0} \mathcal{G}'_n$. On generating function level, the pointing operation means taking the derivative with respect to x, and multiplying by it, i.e. the egf $G^\bullet(x) = \sum_{n\geq 0} g^\bullet_n \frac{x^n}{n!}$ satisfies $G^\bullet(x) := xG'(x)$, where $g^\bullet_n := |\mathcal{G}^\bullet_n| = ng_n$. Similarly, the egf of $\mathcal{G}'$ satisfies $G'(x) = \sum_{n\geq 0} g'_n \frac{x^n}{n!} = \frac{\partial}{\partial x} G(x)$.

By applying a standard decomposition of a graph into 2-connected blocks (see [HP73, p. 10]), and by using singularity analysis (see [FS05]) we obtain the following lemma. Here we denote by $\mathcal{Z}$ the graph class consisting of one single labelled node. Furthermore, for two graph classes $\mathcal{X}$ and $\mathcal{Y}$, we denote by $\mathcal{X} \times \mathcal{Y}$ the *cartesian product* of $\mathcal{X}$ and $\mathcal{Y}$, followed by a relabelling step, so as to guarantee that all labels are distinct for an object $\gamma \in \mathcal{X} \times \mathcal{Y}$. Moreover, $\mathfrak{Set}(\mathcal{X})$ is the graph class such that each object in it is a *set* of graphs in $\mathcal{X}$. Finally, the class $\mathcal{X} \circ \mathcal{Y}$ (*substitution*) consists of all graphs that are obtained from graphs from $\mathcal{X}$, where each node is replaced by a $\mathcal{Y}$ graph. For a more detailed description we refer to [DFLS04].

Lemma 2.1 *Let $\mathcal{Q}$ be the class of all labelled cliques on at least two nodes. The class $\mathcal{B}^\bullet$ satisfies $\mathcal{B}^\bullet = \mathcal{Z} \times \mathfrak{Set}(\mathcal{Q}' \circ \mathcal{B}^\bullet)$. Furthermore, the egf for $\mathcal{B}^\bullet$ admits a singular expansion around its singularity ρ_B of the form $B^\bullet(x) = B^\bullet_0 + B^\bullet_1 X^{1/2} + o\big(X^{1/2}\big)$, where $X = 1 - \frac{x}{\rho_B}$, and $B^\bullet_0$, $B^\bullet_1$, and ρ_B are implicitly given constants.*

The proof is an application of singularity analysis, and we omit the details due to space constraints. Proofs of similar results can be found e.g. in [BGKN05].

3 Sampling

In this section we will demonstrate how we can obtain efficient sampling procedures for the graph classes in interest. The framework of Boltzmann samplers was introduced by Duchon et al. in [DFLS04], and was only recently extended and applied to obtain an efficient sampler for random planar graphs by Fusy [Fus05]. This framework will allow us to derive an algorithm whose running time is expected *linear* in the size of the output object for approximate-size sampling and expected *quadratic* for exact-size sampling.

Here we just present the basic ideas of this framework for the case of exponential generating functions; for further details see [DFLS04]. Let $\mathcal{G}$ denote the class of objects, in which we are interested, and let $G(x)$ denote its egf. In the Boltzmann model of parameter x, we assign to any object $\gamma \in \mathcal{G}$ of size $|\gamma|$ the probability

$$\Pr[\gamma] = \frac{1}{G(x)} \cdot \frac{x^{|\gamma|}}{|\gamma|!}, \tag{1}$$

if the above expression is well-defined. A Boltzmann sampler $\Gamma G(x)$ for class $\mathcal{G}$ is a process that produces objects from $\mathcal{G}$ according to the corresponding Boltzmann model. In [DFLS04] several general procedures are given, which translate construction rules like union, set, etc. into Boltzmann samplers. Fusy (see [Fus05]) extended these by further constructions, as for instance substitution. Our classes can be described within this set of rules, which implies that we can derive systematically a Boltzmann sampler for them.

The efficiency of a Boltzmann sampler for a class $\mathcal{G}$ depends highly on the type of the singular expansion of its generating function, see Theorems 6.3 and 7.3 in [DFLS04]. More precisely, in order to obtain an expected linear running time sampler, either the exponent α of the singular expansion of $G(x)$ has to be negative *or*, in the special case that $-\alpha = \frac{1}{2}$, we can use a so-called *singular ceiled rejection sampler*. This sampler simply discards objects during sampling, as soon as they become too large.

The ceiled rejection sampler has the advantages that it is not necessary to tune it explicitly for a specific output size and that only a very small amount of constants has to be calculated (in our case only two, see below). In contrast, if we want to obtain an efficient sampler of the first type, we have to point our objects as many times as it is necessary to obtain a negative singular exponent (which increases the number of involved generating functions and hence the complexity of the resulting sampler) and tune the input parameters for every desired output size, see Theorem 6.3 in [DFLS04]. Furthermore, if we want to implement the sampler, the generating functions have to be evaluated very precise for values near their singularities, which means that we might have to calculate many terms of the singular expansions. Therefore we design a singular rejection sampler for the considered graph classes. Finally, it turns out that the singular rejection sampler is very well suited for our further analysis.

The singular rejection sampler repeatedly samples *pointed* block graphs according to (1), until an object with size in $((1-\varepsilon)n, (1+\varepsilon)n)$ is sampled. It maintains a global variable $nodes$, which counts the number nodes which were generated through recursive calls.

```
$\Gamma\nu B(n,\varepsilon)$:   repeat
                    $nodes \leftarrow 0$
                    $\gamma \leftarrow \Gamma B^\bullet(n,\varepsilon)$
                until $|\gamma| > (1-\varepsilon)n$
                label the nodes of $\gamma$ uniformly at random
                return $\gamma$ (and discard the root)
```

The sampler for rooted block graphs $\Gamma B^\bullet$ works as follows. Roughly speaking, due to Lemma 2.1, a rooted block graph is a set of biconnected rooted block graphs, merged at their roots, in which every node is again replaced by a rooted block graph. The sampler reverses this decomposition: it starts with a single node, chooses according to a carefully chosen probability distribution a random number of cliques adjacent to it, and then replaces each generated node by a randomly drawn rooted block graph.

```
$\Gamma B^\bullet(n,\varepsilon)$:   $\gamma \leftarrow$ a single root node $r$
                $nodes \leftarrow nodes + 1$
                $k \leftarrow \mathrm{Po}\,(Q'(B^\bullet(\rho_B)))$                // $\rho_B$ denotes the singularity of $\mathcal{B}^\bullet$
                for ($j = 1 \ldots k$)
                    $\gamma' \leftarrow \Gamma Q'_{<(1+\varepsilon)n}(B^\bullet(\rho_B))$
                    if ($\gamma'$ equals $\perp$) return $\perp$
                    $\gamma \leftarrow$ merge $\gamma$ and $\gamma'$ at their root nodes
                for (each node $v \neq r$ of $\gamma$)
                    $nodes \leftarrow nodes - 1$
                    $\gamma_v \leftarrow \Gamma B^\bullet(n,\varepsilon)$
                    if ($\gamma_v$ equals $\perp$) return $\perp$
                replace all nodes $v \neq r$ of $\gamma$ with $\gamma_v$
                return $\gamma$
```

Finally, the sampler $\Gamma Q'_{<m}$ for cliques choses the size of the objects it outputs according to its probability in the Boltzmann model. It returns a default empty object ($\perp$), if the size of the generated object would have been too large. Before we can state it formally, let us define the distribution it samples from.

Definition 3.1 *The clique distribution with parameter $x > 0$, denoted by* $\mathrm{Cl}(x)$*, is defined by*

$$\Pr[\mathrm{Cl}(x) = k] \;:=\; \begin{cases} \frac{x^k}{Q'(x)k!}, & k \geq 1 \\ 0, & \textit{otherwise,} \end{cases}$$

where $Q'(x) := e^x - 1$ is the egf of $\mathcal{Q}'$, where $\mathcal{Q}$ is the class of cliques of size ≥ 2.

With this, the sampler can be implemented as follows:

```
$\Gamma Q'_{<m}(x)$:   $k \leftarrow \mathrm{Cl}\,(x)$
                $nodes \leftarrow nodes + k$
                if($nodes \geq m$)
                    generate $k - (nodes - m)$ nodes          $(\star)$
                    return $\perp$
                else return a rooted clique on $k+1$ nodes
```

It can easily be checked that the sampler above generates cliques according to the Boltzmann distribution (1), if it does not early interrupt (returning $\perp$). Note that in the line marked by $(\star)$, the sampler returns $\perp$ *after* it has generated as many nodes as it is necessary to have generated precisely m nodes in the object sampled in the current execution of $\Gamma B^\bullet$. This is a detail convenient for the analysis which is useful in the proof of Theorem 4.2.

This completes the specification of the sampler for block graphs. By applying Lemma 2.1, Theorem 7.3 of [DFLS04], and Proposition 1 from [Fus05] we obtain straightforwardly the following lemma.

Lemma 3.2 *Let $0 < \varepsilon < 1$ and $n \in \mathbb{N}_0$. The following two statements hold.*

- *$\Gamma B^\bullet(n,\varepsilon)$ is a Boltzmann sampler for the class $\mathcal{B}^\bullet$ of rooted block graphs. For each $(1-\varepsilon)n < N < (1+\varepsilon)N$ it satisfies* $\Pr\left[|\Gamma B^\bullet(n,\varepsilon)| = N\right] = \Theta(n^{-3/2})$.
- *$\Gamma\nu B(n,\varepsilon)$ outputs a graph from $\mathcal{B}$ with size in $I_\varepsilon = (n(1-\varepsilon), n(1+\varepsilon)))$, such that the distribution is uniform for each size in I_ε. Furthermore, let T be the running time of* $\Gamma\nu B(n,\varepsilon)$*. Then* $\mathbb{E}[T] = \mathcal{O}(n\varepsilon^{-1})$ *and* $\mathrm{Var}[T] = \mathcal{O}(n^2\varepsilon^{-2})$.

Note that in a similar fashion we can design Boltzmann samplers for the graph classes $\mathcal{C}, \mathcal{E}, \mathcal{T}$ and $\mathcal{D}$. With the above lemma, Theorem 1.1 follows immediately. As a technical tool, we will later need the following observation, which says that all possible output sizes are almost equally probable to occur, even if we constrain the number of atoms the sampler can generate.

Lemma 3.3 *Let $0 < \varepsilon \leq \frac{1}{2}$. Whenever n is sufficiently large the following statement holds. For every $s \in \mathbb{N}$ and $t \in \mathbb{N}_0$ define the quantity*

$$p_{s,t} := \Pr\left[|\Gamma\nu B(n,\varepsilon)| = s \textit{ and } \Gamma\nu B(n,\varepsilon) \textit{ created precisely } t+s \textit{ nodes}\right]. \qquad (2)$$

For all $s = \alpha n$, where $\alpha \in (1-\varepsilon, 1+\varepsilon)$ it holds $p_{s,t} \sim \alpha^{-3/2} \cdot p_{n,t}$ (where "$\sim$" is with respect to $n \to \infty$).

4 The maximum degree of a random block graph

In this chapter we are going to demonstrate our proof method by estimating the maximum node degree in a random block graph. Following the same proof pattern, one can show the corresponding results of Theorem 1.2 for the other graph classes, as well as the statements about the size of maximum blocks. Before we proceed, let us first make a technical definition. We say that a random variable X is *sumClique*-distributed with parameters λ and μ, $X \sim \mathrm{SCl}(\lambda,\mu)$, if it is the sum of $\mathrm{Po}(\lambda)$ independent $\mathrm{Cl}(\mu)$ distributed random variables. A crucial ingredient in our proof for the magnitude of the maximum degree in a random block graph will be the following lemma, which is a statement about the tail of a sumClique distributed random variable.

Lemma 4.1 *Let $\lambda, \mu > 0$ and $0 < \varepsilon < \frac{1}{9}$ be constants. There is an $n_0 = n_0(\varepsilon) \geq 0$ such that whenever $n > n_0$ it holds*

$$\Pr\left[\mathrm{SCl}(\lambda,\mu) \geq (1-\varepsilon)\frac{\log n}{\log^{(3)}(n)}\right] \gg n^{-1+\frac{\varepsilon}{2}} \textit{ and } \Pr\left[\mathrm{SCl}(\lambda,\mu) \geq (1+\varepsilon)\frac{\log n}{\log^{(3)}(n)}\right] \ll n^{-1-\frac{\varepsilon}{2}}.$$

The proof can be found at the end of the section. Observe that this lemma in fact implies that the maximum of n independent $\mathrm{SCl}(\lambda,\mu)$ variables is concentrated around $\frac{\log n}{\log^{(3)}(n)}$ (independent of the actual values of λ and μ). With these preparations, we are ready to prove one of our main results.

Theorem 4.2 *Let $\varepsilon > 0$. For almost all n, the maximum node degree in a random block graph is with probability at least $1 - n^{-\varepsilon/40}$ in the interval $\mathcal{I}_{\varepsilon,n} := ((1-\varepsilon)d(n), (1+\varepsilon)d(n))$, where $d(n) := \frac{\log n}{\log^{(3)} n}$.*

Proof:

The main idea in the following proof is to consider the process of object construction during a run of the singular Boltzmann sampler $\Gamma\nu B$ for block graphs, as it is described in Section 3. Recall that the output of $\Gamma\nu B$ is a random block graph of varying size from a well-defined distribution (Lemma 3.3), and here we want to make a statement about random block graphs B_n of a *given size* n. As described below, it turns out to be very convenient to study the properties of the *generated shapes* of $\Gamma\nu B$, instead of studying properties of B_n directly. We shall see that these translate to properties of B_n in a straightforward way.

Before we proceed, let us slightly modify the sampling algorithm defined in Section 3. This is done for solely technical reasons, and will become clear later. The idea behind the singular rejection sampler $\Gamma\nu B$ is to repeatedly sample and reject rooted block graphs, until $\Gamma B^\bullet$ outputs an object of the desired size. $\Gamma B^\bullet$ proceeds as follows: for every generated node v, it calculates a random value p_v according to a Poisson law with parameter $\lambda := Q'(B^\bullet(\rho_B))$, and then calculates p_v independent random values according to a clique law with parameter $\mu := B^\bullet(\rho_B)$. Then it generates rooted cliques with sizes given by the random values, joins them at their roots, and proceeds in an identical way for all newly created nodes, until the process dies out. The important modification in the sampling procedure below is that we let the sampler $\Gamma\nu B$ make *precisely* $c(n) := \lceil n^{1+\varepsilon/100} \rceil$ random choices, abort after that immediately its execution, and output the first object of size in $\mathcal{I}_{\varepsilon,n} := (\frac{n}{2}, \frac{3n}{2})$, if there is any, that was generated during a call of $\Gamma B^\bullet$:

```
$\widetilde{\Gamma}\nu B(n)$:   for($i = 1 \ldots \lceil n^{1+\varepsilon/100} \rceil$)
                   $p_i \leftarrow$ random value according to $\mathrm{Po}\,(\lambda)$
                   $(C_1^{(i)}, \ldots, C_{p_i}^{(i)}) \leftarrow p_i$ independent random values according to $\mathrm{Cl}\,(\mu)$
                   $d_i :\leftarrow \sum_{1 \le j \le p_i} C_j^{(i)}$        $(\star\star)$
               run the sampler $\Gamma\nu B(n, \frac{1}{2})$ with the above random values, and do not
                   stop its execution if it generated an object with size in $(\frac{n}{2}, \frac{3n}{2})$, and
                   abort it as soon as all $\lceil n^{1+\varepsilon/100} \rceil$ random values were exhausted
               if(above execution of $\Gamma\nu B(n, \frac{1}{2})$ generated an object with size in $(\frac{n}{2}, \frac{3n}{2})$)
                   return the first such object
               else return $\perp$
```

The values d_i calculated in the line marked with $(\star\star)$ are needed later in the analysis. Before we proceed, let us make some important observations. $\widetilde{\Gamma}\nu B$ behaves very similar to the sampler $\Gamma\nu B$ described in Section 3. The only difference is that if it generates an object of size in $\mathcal{I}$, it does not output it directly, but continues its execution until it has generated at least $\lceil n^{1+\varepsilon/100} \rceil$ nodes. On the other hand, if it does not succeed in sampling an object of the desired size, it returns a default object $\perp$. Moreover, observe that if the sampler $\Gamma B^\bullet$, called by the algorithm above, had built a graph on s nodes, then it would have used *exactly* s *consecutive* values $p_i, \ldots, p_{i+s}$, and the corresponding random clique-distributed values to do so.

In order to prove the theorem, we will proceed in two steps. Let

$$\tilde{\mathcal{B}} := \left\{ \gamma \in \mathcal{B} \;\mid \text{the maximum degree of } \gamma \text{ is in } I_{\epsilon,|\gamma|} \right\}.$$

First, we are going to show

$$\Pr\left[\widetilde{\Gamma}\nu B(n) \in \tilde{\mathcal{B}}\right] = 1 - n^{-\varepsilon/20}, \tag{3}$$

i.e., with high probability, the sampler $\widetilde{\Gamma}\nu B(n)$ will output an object, which is not $\perp$, and which has the property that its maximum degree lies in the desired interval. Then, in the second part of the proof, we will show that the above statement indeed proves the theorem.

First we show (3). Observe that the sampler $\Gamma\nu B$, when called by $\widetilde{\Gamma}\nu B$, samples firstly the *shape* of a graph – the labels are distributed on the nodes at the end of the process. For a labelled graph G define $s(G)$ to be its shape, i.e., the corresponding unlabelled graph, and for a shape S, let $\ell(S)$ be the set of labelled graphs with shape S, and let $|S|$ denote the number of nodes in S. Furthermore, set $s(\tilde{\mathcal{B}}) := \bigcup_{G \in \tilde{\mathcal{B}}} s(G)$. Now, for a shape S with $|S| \in \mathcal{I}$, let $\Pr[S]$ be the probability that S is the first shape which has size in $\mathcal{I}_{\varepsilon,n}$ generated by the repeated execution of $\Gamma B^\bullet$, and let $\Pr[G \mid S]$ be the probability that the sampler outputs the labelled graph G, given that the shape S had been generated. Observe that $\Pr[G \mid S] = \frac{\mathrm{aut}(S)}{|S|!}$, if $S = s(G)$, and 0 otherwise, where $\mathrm{aut}(S)$ denotes the number of automorphisms of S. With this, we get

$$\Pr\left[\Gamma\nu B(n) \in \tilde{\mathcal{B}}\right] = \sum_{S \in s(\tilde{\mathcal{B}})} \Pr[S] \cdot \sum_{G \in \ell(S)} \Pr[G \mid S] = \sum_{S \in s(\tilde{\mathcal{B}})} \Pr[S] \cdot |\ell(S)| \cdot \frac{\mathrm{aut}(S)}{|S|!} = \sum_{S \in s(\tilde{\mathcal{B}})} \Pr[S], \tag{4}$$

where the last step follows from the fact $|\ell(S)| = \frac{|S|!}{\mathrm{aut}(S)}$. Hence, in order to show (3), it will suffice to prove that the shape generated by $\Gamma\nu B$, during its execution in $\widetilde{\Gamma}\nu B$, has with high probability the property that all its nodes have maximum degree in the desired interval.

Let us now focus on the evolution of the node degrees during the sampling process. The sampler repeatedly calls $\Gamma B^{\bullet}(n, \frac{1}{2})$, which starts with a single node v. This algorithm calculates a random value p_v according to a Poisson law with parameter λ, and then calculates p_v random values $C_1^{(v)}, \ldots, C_{p_v}^{(v)}$ according to a clique law with parameter μ. This determines the *size* and *structure* of the neighbourhood of v. Then it creates p_v cliques with sizes $C_1^{(v)} + 1, \ldots, C_{p_v}^{(v)} + 1$, and joins them together at their roots (and v). Finally, for each of the newly created nodes, the same procedure is repeated independently until the process dies out, or it is interrupted, because too many nodes were generated. Thus, the degree of a node is given by the outcome of a *sumClique*-distributed random variable with parameters λ and μ, plus the size of the clique, in which this node is contained, minus one. All in all, a *lower bound* for the degree of a node v generated during the sampling process is its *sumClique*-value $\mathrm{scl}\,(v)$, and an *upper bound* is the size of the clique it is contained in plus $\mathrm{scl}\,(v)$.

Let $\varepsilon' := \frac{\varepsilon}{2}$. With the above discussion in mind, consider the execution of $\widetilde{\Gamma}\nu B(n)$, which generates the random values p_i, $C_j^{(i)}$ and d_i (in the line marked with $(\star\star)$ in the exposition of the algorithm), and let us define the following four events:

(A) $\widetilde{\Gamma}\nu B(n) \neq \perp$.

(B) Every sequence of $\frac{n}{2}$ consecutive values d_i contains a value larger than $(1-\varepsilon')d(n)$, i.e.,

$$\mathsf{B} := \left\{ (d_1, \ldots, d_{\lceil n^{1+\varepsilon/100} \rceil}) \;\middle|\; \forall 1 \le i \le \lceil n^{1+\varepsilon/100} \rceil - \frac{n}{2} : \exists i \le j \le i + \frac{n}{2} : d_j \ge (1-\varepsilon')d(n) \right\}.$$

(C) There is no $1 \le i \le \lceil n^{1+\varepsilon/100} \rceil$ such that $d_i \ge (1+\varepsilon')d(n)$.

(D) For all $1 \le i \le \lceil n^{1+\varepsilon/100} \rceil$ and $1 \le j \le p_i$ we have $C_j^{(i)} \le 5\frac{\log n}{\log\log n}$.

The motivation behind the above events is that if they occured simultaneously, then the shape generated by $\widetilde{\Gamma}\nu B$ would have the property that the maximum node degree is in $\mathcal{I}_{\varepsilon,n}$. In fact, suppose that $\widetilde{\Gamma}\nu B(n) \neq \perp$, i.e. (A) holds. Then there is an index i_0 and a number $s \in (\frac{n}{2}, \frac{3n}{2})$, such that the sampler $\Gamma B^{\bullet}$, given the random choices $p_{i_0}, \ldots, p_{i_0+s}$ and the corresponding values $(C_1^{(i_0)}, \ldots, C_{p_{i_0}}^{(i_0)}), \ldots, (C_1^{(i_0+s)}, \ldots, C_{p_{i_0+s}}^{(i_0+s)})$, outputs a shape of size s. Now, suppose that additionally B, C and D occur. Then, due to B, there is an index $i_0 \le i' \le i_0 + s$, such that $d_{i'} = \sum_{j=1}^{p_{i'}} C_j^{(i')} \ge (1-\varepsilon')d(n)$, i.e. $\widetilde{\Gamma}\nu B(n)$ has a node of at least that degree. Furthermore, due to C and D, there is for sufficiently large n no node with degree larger than

$$5\frac{\log n}{\log\log n} + (1+\varepsilon')d(n) = 5\frac{\log n}{\log\log n} + (1+\varepsilon')\frac{\log n}{\log^{(3)} n} < (1+\varepsilon)d(n).$$

Hence, with (4) we have $\Pr[\widetilde{\Gamma}\nu B(n) \in \tilde{\mathcal{B}}] \ge \Pr[\mathsf{A} \text{ and } \mathsf{B} \text{ and } \mathsf{C} \text{ and } \mathsf{D}]$. In order to show (3), we have to prove that the probabilities for the complementary events $\overline{\mathsf{A}}$, $\overline{\mathsf{B}}$, $\overline{\mathsf{C}}$, and $\overline{\mathsf{D}}$ are all at most $n^{-\varepsilon/10}$. These proofs merely consist of routine calculations, and exploit Lemma 4.1 – we omit them due to space limitations.

To complete the proof, we now show how (3) implies the theorem. In fact, let B_n be a random block graph on n nodes. Let $\delta := n^{-\varepsilon/20}$, and choose a sufficiently large n such that

$$\Pr\left[\widetilde{\Gamma}\nu B(n) \in \tilde{\mathcal{B}}\right] > 1 - \delta \quad \text{and} \quad \sum_{n/2 < s < 3n/2} \Pr\left[|\widetilde{\Gamma}\nu B(n)| = s\right] > 1 - \delta.$$

Such an n always exists, due to (3) and Lemma 3.2. But then we have

$$\Pr\left[\widetilde{\Gamma}\nu B(n) \in \tilde{\mathcal{B}}\right] \; < \; \delta + \sum_{n/2 < s < 3n/2} \Pr\left[\widetilde{\Gamma}\nu B(n) \in \tilde{\mathcal{B}} \;\middle|\; |\widetilde{\Gamma}\nu B(n)| = s\right] \Pr\left[|\widetilde{\Gamma}\nu B(n)| = s\right],$$

which yields with our assumptions on δ

$$1-2\delta \ < \sum_{n/2<s<3n/2} \Pr\left[\mathsf{B}_s \in \mathcal{B}'\right] \Pr\left[|\widetilde{\Gamma}\nu B(n)| = s\right]. \tag{5}$$

Let $\alpha \in (\frac{1}{2}, \frac{3}{2})$. Before we proceed, we are going to show that for all $s = \alpha n$ the probabilities $\Pr\left[|\widetilde{\Gamma}\nu B(n)| = s\right]$ differ by at most a multiplicative factor. In fact, denote as in Lemma 3.3 with $p_{s,t}$ the probability that $\Gamma\nu B$ returns an object of size s, and having generated precisely $t+s$ nodes during its execution. Then, for all s as above

$$\Pr\left[|\widetilde{\Gamma}\nu B(n)| = s\right] = \sum_{t \le \lceil n^{1+\varepsilon/100} \rceil} p_{s,t} \overset{\text{(Lemma 3.3)}}{\sim} \alpha^{-3/2} \cdot \sum_{t \le \lceil n^{1+\varepsilon/100} \rceil} p_{n,t},$$

and the claim follows, as the last sum is independent of s. But this yields with (5) that, say, for more than $(1-30\sqrt{\delta})n$ numbers $s_1, \dots$ in the interval $(\frac{n}{2}, \frac{3n}{2})$, we have $\Pr[\mathsf{B}_{s_i} \in \tilde{\mathcal{B}}] \ge 1-\sqrt{\delta}$, as otherwise the sum on the right-hand side of (5) would have been smaller than $1-2\delta$. Hence, for every $\delta > 0$, in every interval of the form $(\frac{n}{2}, \frac{3n}{2})$, when n is sufficiently large, we have at most $30\sqrt{\delta}n$ numbers with $\Pr[\mathsf{B}_{s_i} \in \tilde{\mathcal{B}}] \le 1-\sqrt{\delta}$. This completes the proof. □ **Proof Proof of Lemma**

4.1: In the following we shall omit $\lceil . \rceil$ to keep the calculations concise, but it can easily be verified that our statements are also true in the general case. It is well-known that a sum of Poisson variables is distributed as a single Poisson variable with the sum of their parameters. Furthermore note that that there is a constant c_μ such that $\mathrm{Po}\,(\mu = x) \le \Pr\left[\mathrm{Cl}\,(\mu) = x\right] \le c_\mu \mathrm{Po}\,(\mu = x)$.Then, and for all $t, s \in \mathbb{N}$, if $C_1, \dots, C_t$ are independent clique-distributed variables with parameter μ, we obtain

$$\Pr\left[\mathrm{Po}\,(t\mu) = s\right] \ \le \ \Pr\left[\sum_{i=1}^{t} C_i - 1 = s\right] \ \le \ c_\mu^t \cdot \Pr\left[\mathrm{Po}\,(t\mu) = s\right]. \tag{6}$$

Let us abbreviate $b(n) := \frac{\log n}{\log^{(3)} n}$ and $q(n) := \frac{\log n}{\log\log n}$. With (6) we obtain

$$\Pr\left[\mathrm{SCl}\,(\lambda,\mu) \ge (1-\varepsilon)b(n)\right] \quad \ge \quad \Pr\left[\mathrm{Po}\,(\lambda) = \frac{\varepsilon}{9}q(n)\right] \Pr\left[\mathrm{Po}\left(\frac{\varepsilon\mu}{9}q(n)\right) \ge (1-\varepsilon)b(n)\right].$$

We estimate the terms on the right hand side of the above inequality one by one. The first probability is easily seen to be $n^{-\varepsilon/9+o(1)}$, if n is sufficiently large. In the sequel we will show that for all $\alpha, \beta > 0$

$$\Pr\left[\mathrm{Po}\,(\alpha q(n)) \ge \beta b(n)\right] = n^{-\beta+o(1)}, \tag{7}$$

which immediately completes the proof of the first statement of the lemma. Indeed, with the definition of the Poisson distribution and the estimate $x! = (1+o(1))\left(\frac{x}{e}\right)^x \sqrt{2\pi x}$ we obtain

$$\begin{aligned} \Pr\left[\mathrm{Po}\,(\alpha q(n)) \ge \beta b(n)\right] = \sum_{t \ge \beta b(n)} e^{-\alpha q(n)} \cdot \frac{(\alpha q(n))^t}{t!} &= (1+o(1)) \cdot e^{-\alpha q(n)} \cdot \frac{(\alpha q(n))^{\beta b(n)}}{(\beta b(n))!} \\ &= n^{o(1)} \cdot \left(\frac{e\alpha}{\beta} \cdot \frac{\log^{(3)} n}{\log\log n}\right)^{\beta \frac{\log n}{\log^{(3)} n}} = n^{-\beta+o(1)}. \end{aligned}$$

Now we show the second statement of the lemma. Recall (6) and note that for all $t \ge \lambda_1 \ge \lambda_2 \ge 1$ we have $\Pr\left[\mathrm{Po}\,(\lambda_1) \ge t\right] \ge \Pr\left[\mathrm{Po}\,(\lambda_2) \ge t\right]$. We obtain with $q(n) = \frac{\log n}{\log\log n}$

$$\begin{aligned} \Pr\left[\mathrm{SCl}\,(\lambda,\mu) \ge (1+\varepsilon)b(n)\right] &= \sum_{t \ge 0} \Pr\left[\mathrm{Po}\,(\lambda) = t\right] \cdot \Pr\left[\sum_{i=1}^{t} \mathrm{Cl}\,(\mu) - 1 \ge (1+\varepsilon)b(n)\right] \\ &\le \sum_{t=0}^{2q(n)} \Pr\left[\mathrm{Po}\,(\lambda) = t\right] \cdot \Pr\left[\mathrm{Po}\,(t\mu) \ge (1+\varepsilon)b(n)\right] \cdot c_\mu^t + \Pr\left[\mathrm{Po}\,(\lambda) \ge 2q(n)\right] \\ &\le n^{o(1)} \cdot \Pr\left[\mathrm{Po}\,(2q(n)) \ge (1+\varepsilon)b(n)\right] + n^{-2+o(1)} \\ &\overset{(7)}{\le} n^{-1-\varepsilon+o(1)} + n^{-2+o(1)} \ll n^{-1-\frac{\varepsilon}{2}}. \end{aligned}$$

□

References

[BGKN05] M. Bodirsky, O. Giménez, M. Kang, and M. Noy. On the number of series parallel and outerplanar graphs. In *EuroComb '05*, volume AE of *DMTCS Proceedings*, pages 383–388, 2005.

[BHM02] H.-J. Bandelt, K. T. Huber, and V. Moulton. Quasi-median graphs from sets of partitions. *Disc. Appl. Math.*, 122(1-3):23–35, 2002.

[DFH$^+$99] L. Devroye, P. Flajolet, F. Hurtado, M. Noy, and W. Steiger. Properties of random triangulations and trees. *Discrete Comput. Geom.*, 22(1):105–117, 1999.

[DFLS04] P. Duchon, P. Flajolet, G. Louchard, and G. Schaeffer. Boltzmann samplers for the random generation of combinatorial structures. *Combinatorics, Probability and Computing*, 13(4-5):577–625, 2004.

[Fle99] L. Fleischer. Building chain and cactus representations of all minimum cuts from Hao-Orlin in the same asymptotic run time. *Journal of Algorithms*, 33(1):51–72, 1999.

[FN99] P. Flajolet and M. Noy. Analytic combinatorics of non-crossing configurations. *Discrete Math.*, 204(1-3):203–229, 1999.

[FS05] P. Flajolet and R. Sedgewick. Analytic combinatorics. Book in preparation, October, 2005.

[Fus05] E. Fusy. Quadratic exact-size and linear approximate-size random generation of planar graphs. In *2005 International Conference on Analysis of Algorithms*, volume AD of *DMTCS Proceedings*, pages 125–138, 2005.

[GMSW05] S. Gerke, C. McDiarmid, A. Steger, and A. Weißl. Random planar graphs with n nodes and a fixed number of edges. *Proceedings of the 16th ACM-SIAM Symposium on Discrete Algorithms*, pages 999–1007, 2005.

[GN05] O. Giménez and M. Noy. The number of planar graphs and properties of random planar graphs. In *2005 International Conference on Analysis of Algorithms*, volume AD of *DMTCS Proceedings*, pages 147–156. xx, 2005.

[Gou98] X. Gourdon. Largest component in random combinatorial structures. *Discrete Math.*, 180(1-3):185–209, 1998.

[GW00] Z. Gao and N. C. Wormald. The distribution of the maximum vertex degree in random planar maps. *J. Comb. Theory Ser. A*, 89(2):201–230, 2000.

[HP73] F. Harary and E. M. Palmer. *Graphical enumeration*. Academic Press, New York, 1973.

[Moo68] J. W. Moon. On the maximum degree in a random tree. *Mich. Math. J.*, 15:429–432, 1968.

[MSW05] C. McDiarmid, A. Steger, and D. Welsh. Random planar graphs. *Journal of Combinatorial Theory, Series B*, 93:187 – 205, 2005.

[MSW06] C. McDiarmid, A. Steger, and D. Welsh. Random graphs from planar and other addable classes. *Topics in Discrete Mathematics*, pages 231–246, 2006.