

# Huge Pseudo-Random Graphs that Preserve Global Properties of Random Graphs

M.Sc. Thesis by Asaf Nussboim

Advisor: Prof. Shafi Goldwasser

# Acknowledgements

This study is a joint work with Prof. Shafi Goldwasser and Prof. Oded Goldreich. I'm very grateful to Shafi for warmly accepting me to Weizmann family, for introducing me to this fascinating field, and for her help during my research and writing periods. I am also very grateful to Oded for his help and for teaching me and sharing his invaluable knowledge on pseudo-randomness. Specifically, the testing algorithm  $T_2$  in page 72 is due to him. Next, I wish to thank Noa Agmon-Segal for her help in editing this work. Finally, I wish to thank the many excellent teachers I was lucky to have both in the Weizmann institute and in the Hebrew University, and especially Prof. Avi Wigderson and Prof. Nati Linial.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Randomness vs Pseudo-Randomness . . . . .	3
1.1.1	Pseudo-Random Strings . . . . .	4
1.1.2	Pseudo-Random Functions . . . . .	5
1.2	Global vs Local Properties of Huge Random Graphs . . . . .	6
1.2.1	Pseudo-Random Functions Can Provide Feasible Implementations of Huge Graphs . . . . .	6
1.2.2	Properties of Random Graphs . . . . .	7
1.2.3	Pseudo-Random Graphs May Defy Global Properties of Random Graphs . . . . .	10
1.3	Constructing Pseudo-Random Graphs Preserving Properties of Random Graphs . . . . .	11
1.3.1	Pseudo-Random Graphs Preserving the Clique Number, Independence Number and Chromatic Number of Random Graphs . . . . .	12
1.3.2	Pseudo-Random Graphs Preserving Arbitrary Sparse Monotone Properties . . . . .	13

1.3.3	Pseudo-Random Graphs Approximating the Connectivity Number and the Minimal and Maximal Degrees of Random Graphs . . . . .	14
1.3.4	Strengthening the Pseudo-Randomness of the Graphs . . . . .	15
1.4	Roadmap . . . . .	16
<b>2</b>	<b>Preliminaries</b>	<b>17</b>
2.1	Strings and Functions . . . . .	17
2.2	Computational Notions and Notations . . . . .	18
2.3	Graph Notation and Random Graphs . . . . .	19
2.4	Basic Probability Tools . . . . .	20
<b>3</b>	<b>Pseudo-Random Functions - Definitions and Basic Tools</b>	<b>21</b>
3.1	Pseudo-Random Functions - Discussion . . . . .	21
3.2	Pseudo-Random Functions - Definitions . . . . .	22
3.3	Mild Modifications Preserve Pseudo-Randomness . . . . .	24
3.3.1	Boolean Combinations of Efficiently Computable Ensembles . . . . .	25
3.3.2	Mild Modifications of Efficiently Computable Ensembles . . . . .	25
3.3.3	Mild Modifications Preserve Pseudo-Randomness . . . . .	26
<b>4</b>	<b>Pseudo-Random Graphs Preserving Properties of Random Graphs</b>	<b>30</b>
4.1	Graphs Notation . . . . .	30
4.2	Pseudo-Random Graphs - Discussion . . . . .	31
4.3	Pseudo-Random Graphs - Definitions . . . . .	32
4.4	Random Graphs' Properties . . . . .	33

4.5	Pseudo-Random Graphs Preserving the Clique Number, Independence Number and Chromatic Number of Random Graphs	35
4.6	Pseudo-Random Graphs Preserving Sparse Monotone Properties	59
4.6.1	Handling Hamiltonicity, Perfect Matching, and $V^c$ -Connectivity	59
4.6.2	Handling General ECS-Properties . . . . .	63
4.7	Pseudo-Random Graphs Approximating the Connectivity Number, and Minimal and Maximal Degrees of Random Graphs . .	65
4.7.1	Handling the Connectivity Number . . . . .	66
4.7.2	Handling the Minimal and Maximal Degrees . . . . .	74
4.8	Simultaneously Preserving the Random-Graphs Properties Previously Handled . . . . .	75
4.9	Strengthening the Pseudo-Randomness of the Constructed Graphs	83
<b>5</b>	<b>Conclusions</b>	<b>85</b>
<b>A</b>	<b>Proofs for some Random Graphs Properties</b>	<b>87</b>
A.1	Hamiltonicity of Random Graphs . . . . .	87
A.2	Minimal and Maximal Degrees of Random Graphs . . . . .	88
A.3	Matchings in Random Graphs . . . . .	89
A.4	High Connectivity of Random Graphs . . . . .	91

## Abstract

This Thesis considers the efficient construction of huge random looking objects of size exponential in  $n$ . These random looking objects are required 1) to maintain a variety of properties that the corresponding random objects do, and 2) to be generated, stored and accessed using polynomially bounded resources. Indeed, based on the assumption that one-way functions exist, [GGM] have efficiently constructed pseudo-random functions  $\mathcal{F}_n$  having an exponentially large domain. These pseudo-random functions are guaranteed to meet any 'local' property of random functions that can be efficiently tested by receiving the value  $\mathcal{F}_n(x_i)$  for  $poly(n)$  inputs  $x_i$ .

Our work focuses on the efficient construction of huge random looking objects which not only preserve local properties of the corresponding random objects, but also maintain '**global**' **properties** that cannot be locally tested.

As a working example, we consider random graphs  $\mathcal{G}_V^{Rnd}$  of order  $V = 2^n$ , which are uniformly distributed over all simple, labeled undirected graphs on  $V$  vertices. Such random graphs have been extensively studied in combinatorics [B1], and are known to exhibit remarkable structure. Indeed, for some value  $s(V) \approx 2 \log V$  (which is easy to compute), a random graph  $\mathcal{G}_V^{Rnd}$  maintains all following conditions with overwhelming probability:

1. Being connected, Hamiltonian and having a perfect matching.
2. Having clique number and independence number exactly  $s(V) \pm 1$ .

3. Having chromatic number  $\frac{V}{s(V)}(1 \pm \frac{1}{\sqrt{\log(V)}})$ .
4. Having maximal and minimal degree  $\frac{1}{2}V(1 \pm 2\sqrt{\frac{\log(V)}{V}})$ .
5. Having connectivity number  $\frac{1}{2}V(1 \pm 2\sqrt{\frac{\log(V)}{V}})$ .

This Thesis, under the assumption that one-way functions exist, efficiently constructs 'pseudo-random graphs'  $\mathcal{G}_V^{Psd}$  which simultaneously preserve conditions 1-3, and approximate conditions 4-5 listed above.

Our techniques are based on pseudo-random functions [GGM], s.t.  $\mathcal{G}_V^{Psd}$  is efficiently computable and preserves any **locally testable property** of random graphs (An example of a local property of random graphs is having  $\approx \frac{1}{2}\binom{V}{2}$  edges with overwhelming probability). We prove that properties 1-5 are **global properties** which are not guaranteed to hold for arbitrary pseudo-random graphs (Indeed, we can actually construct pseudo-random graphs where many edges must be modified in order to achieve properties 1-5). Instead, our construction involve additional steps that enforce the desired properties, but without harming the pseudo-randomness of the resulting graphs.

In addition, the pseudo-randomness of the graphs  $\mathcal{G}_V^{Psd}$  could be strengthened to achieve almost  $(n^k)$ -wise independence, in the sense that the distribution of any  $n^k$  edges in  $\mathcal{G}_V^{Psd}$  is statistically close to the uniform distribution.

# Chapter 1

## Introduction

### 1.1 Randomness vs Pseudo-Randomness

Randomness is a central computational resource. Indeed, for many natural computational problems the best-known probabilistic solution is significantly better than the best-known deterministic one. However, randomness does not come for free. Rather, it is a computational resource we wish to minimize. Furthermore, from the theoretical perspective, one is interested to understand what is the minimal amount of randomness required to perform a given computational task.

In this light, a theory of **computational pseudo-randomness** was pioneered by Shamir, Blum, Micali and Yao [S, BM, Y]. This theory considers pseudo-random objects, which can replace the corresponding random objects in any efficient computation.

### 1.1.1 Pseudo-Random Strings

Since random algorithms may be considered as deterministic algorithms given a random binary string as an auxiliary input, pseudo-random strings are of utmost importance and were the first pseudo-random objects considered in [BM, Y].

Pseudo-Random strings are described by an ensemble of distributions  $\{\mathcal{D}_\ell\}_{\ell \in \mathbb{N}}$  where each  $\mathcal{D}_\ell$  ranges over  $I_\ell$ , the set of all binary strings of length  $\ell$ . By saying that this ensemble is pseudo-random, we mean that no efficient algorithm can distinguish samples taken from  $\mathcal{D}_\ell$  from samples uniformly taken from  $I_\ell$ . Throughout the introduction efficient algorithms are probabilistic polynomial time algorithms in the length of their inputs. Such computational indistinguishability implies that efficient algorithms behave essentially the same when fed with either random or pseudo-random strings.

To replace random strings by pseudo-random ones we wish to efficiently construct pseudo-random strings using limited randomness. Hence, **pseudo-random generators** were introduced. A pseudo-random generator is an efficient deterministic algorithm  $G$  that 'stretches' short inputs of length  $n$  into longer strings of length  $\ell(n)$ , such that for inputs uniformly distributed over  $I_n$ , the distribution of the output strings is pseudo-random. Such pseudo-random generators were shown to exist if and only if one-way functions exist [HILL, BM, Y, L].

### 1.1.2 Pseudo-Random Functions

In several applications, algorithms are assumed to have oracle access to a function in the universe  $F_n = F_n^{\ell_1, \ell_2} = \{f | f : \{0, 1\}^{\ell_1(n)} \rightarrow \{0, 1\}^{\ell_2(n)}\}$ , where  $\ell_1, \ell_2$  are polynomially bounded. This means that an algorithm  $A$  can query a function  $f$  on inputs  $x$  of its choice, and receive the value  $f(x)$  in unit time. For some applications, **random functions** uniformly taken from  $F_n$  are required. However, as the cardinality of  $F_n$  is  $2^{\ell_2(n)2^{\ell_1(n)}}$ , the number of random bits required to specify a function  $f \in F_n$  is  $\ell_2(n)2^{\ell_1(n)}$  which is usually much more than can be afforded.

Thus, an efficient method of utilizing "random looking" functions from  $F_n$  is required. In this light, Goldreich, Goldwasser and Micali [GGM] have introduced pseudo-random functions which can be picked, stored and evaluated efficiently and may faithfully replace truly random functions in a variety of cryptographic applications.

Like pseudo-random strings, pseudo-random functions are described by an ensemble of distributions  $\{\mathcal{F}_n\}_{n \in \mathbb{N}}$ , where each  $\mathcal{F}_n$  is taken over  $F_n$ . Here pseudo-randomness means that no probabilistic algorithm  $D$  running in polynomial time in  $n$  can distinguish a function sampled from  $\mathcal{F}_n$  from a function uniformly taken from  $F_n$ . The algorithm  $D$  may query a function on inputs of its choice but only polynomially many queries are allowed. Explicit constructions of pseudo-random functions were provided in [GGM], assuming that one-way functions exist.

Subsequently, more pseudo-random objects were presented, including pseudo-random permutations by Luby and Rackoff [LR], and pseudo-random permutations having a prescribed cyclic structure by Naor and Reingold [NR].

## 1.2 Global vs Local Properties of Huge Random Graphs

From now on, we assume that one-way functions exist and consequently pseudo-random functions exist.

### 1.2.1 Pseudo-Random Functions Can Provide Feasible Implementations of Huge Graphs

We take particular interest in Boolean functions over  $O(n)$ -bit strings as they naturally represent graphs of order  $V = 2^n$ . Consider the set  $G_V$  of all simple, labeled, undirected graphs on vertices  $\{0, 1, \dots, V - 1\}$ , or the corresponding set  $G'_V$  for directed graphs. These graphs have a (canonical) representation using functions in  $F_n = \{f | f : \{0, 1\}^{2n} \rightarrow \{0, 1\}\}$  as follows.

First, denote  $2n$ -bit strings by  $u||v$  where  $u$  is the  $n$ -bit prefix and  $v$  is the  $n$ -bit suffix. Next, identify each integer  $i \in \{0, 1, \dots, V - 1\}$  with its binary representation. Now, an undirected graph  $g \in G_V$  is represented by the function  $f \in F_n$  where for  $u < v$  an edge  $\{u, v\}$  appears in  $g$  iff  $f(u||v) = 1$ . Similarly, a directed graph  $g' \in G'_V$  is given by  $f \in F_n$ , where an edge  $u \rightarrow v, u \neq v$  is in  $g'$  iff  $f(u||v) = 1$ . In general, to represent graphs of order  $V$  where  $V$  is not a perfect power of 2, we simply set  $n = n(V) = \lceil \log(V) \rceil$ , so  $n$  is the minimal integer such that  $V \leq 2^n$ .

Accordingly, when we say that an algorithm utilizes a specific huge graph, we will mean that the algorithm has an oracle access to a Boolean function representing this graph.

Note that the distribution of the graphs represented by a function uni-

formly taken from  $F_n$ , is identical to the uniform distribution over  $G_V$  (or over  $G'_V$ ). Said differently, a random Boolean function represents a random undirected graph (or a random directed graph resp.). Therefore when an application requires random graphs we can alternatively provide it with oracle access to a random function. Next, we may replace the random functions by pseudo-random ones, thus obtaining pseudo-random graphs. We note that although the graphs considered are exponentially large in  $n$ , they can be manipulated efficiently. Namely, they can be generated, stored and queried consuming time, memory and randomness polynomially bounded in  $n$ .

We remark, that one can come up with a few other mathematical objects, such as  $2^n$  by  $2^n$  Boolean matrices (or matrices of similar dimensions over finite fields), which are naturally represented by Boolean functions over  $O(n)$ -bit strings. Again, the representation could be defined such that the uniform distribution over the huge objects is induced by the uniform distribution over Boolean functions, and again pseudo-random functions could replace the truly random ones and provide efficient utilization of the huge objects concerned. However, in the subsequent discussion we focus our attention on pseudo-random graphs.

## 1.2.2 Properties of Random Graphs

Henceforth, random graphs refer to the uniform distributions over  $G_V$ , the set of all simple, labeled, undirected graphs on vertices  $\{0, 1, \dots, V - 1\}$ .

Now, suppose that the performance of an application utilizing random graphs is known to rely on the fact that a random graph on  $V$  vertices is typically connected and typically has maximal clique size  $\approx 2 \log V$ . This

means that to reliably replace random graphs with pseudo-random ones, the pseudo-random graphs must exhibit the same properties as random graphs do.

To discuss properties of random graphs (which are properties of distributions), we must first consider properties of single graphs, called graph properties. A graph property is a function assigning a real number to each graph. For instance, the graph property  $X_{Clique}$  assigns to each graph  $g$  the size of its maximal clique, whereas the graph property  $X_{Conn}$  assigns to  $g$  the value 1 if  $g$  is connected and the value 0 otherwise.

We next consider the distribution of graph properties over random graphs. These distributions have been extensively studied in combinatorics [B1], and it turns out that for almost any "natural" graph property  $X$ , the distribution of  $X$  is tightly concentrated around some typical value. For instance, for some value  $s(V) \approx 2 \log V$  (which we can easily compute for each  $V$ ) it is known that a random graph  $\mathcal{G}$  on  $V$  vertices obtains all following conditions with probability  $1 - 2^{-\Omega(n)}$ , where  $n = n(V) = \lceil \log(V) \rceil$ ,

- Having clique number  $X_{Clique}(\mathcal{G}) = s(V) \pm 1 = s(V)(1 \pm \frac{1}{s(V)})$ ,
- Having independence number  $X_{Indp}(\mathcal{G}) = s(V) \pm 1 = s(V)(1 \pm \frac{1}{s(V)})$ .
- Having chromatic number  $X_{Color}(\mathcal{G}) = \frac{V}{s(V)}(1 \pm \frac{1}{\sqrt{\log(V)}})$ .
- Being connected - Having  $X_{Conn}(\mathcal{G}) = 1(1 \pm 0)$ .
- Being Hamiltonian - Having  $X_{Ham}(\mathcal{G}) = 1(1 \pm 0)$ .
- Having a perfect matching - Having  $X_{Match}(\mathcal{G}) = 1(1 \pm 0)$ .

- Having maximal degree  $X_{MaxDeg}(\mathcal{G}) = \frac{1}{2}V(1 \pm 2\sqrt{\frac{\log V}{V}})$ .
- Having minimal degree  $X_{MinDeg}(\mathcal{G}) = \frac{1}{2}V(1 \pm 2\sqrt{\frac{\log V}{V}})$ .
- Having connectivity number  $X_{ConnNum}(\mathcal{G}) = \frac{1}{2}V(1 \pm 2\sqrt{\frac{\log V}{V}})$ .

This means that for any of these graph properties  $X$ , it holds for a random graph  $\mathcal{G}$  on  $V$  vertices that  $\Pr[X(\mathcal{G}) \neq \mu_X(V)(1 \pm \delta_X(V))]$  is negligibly small in  $n$ <sup>1</sup>. Here  $\mu_X(V)$  is the typical value of  $X(\mathcal{G})$  for random graphs on  $V$  vertices and  $\delta_X(V)$  tends to 0.

Inspired by this, we define **distributional graph properties** which can either hold or not hold for arbitrary ensembles of distributions  $\mathcal{G}' = \{\mathcal{G}'_V\}_{V \in \mathbb{N}}$  where each distribution  $\mathcal{G}'_V$  is taken over the set of graphs  $G_V$ .

Consider some graph property  $X$ , and let  $\mu, \delta : \mathbb{N} \rightarrow \mathbb{R}$ , where  $\delta(V) \xrightarrow{V \rightarrow \infty} 0$ . The distributional graph property  $P = (X, \mu, \delta)$  is said to hold for the ensemble of distributions  $\mathcal{G}'$ , if  $\Pr[X(\mathcal{G}'_V) \neq \mu(V) \cdot (1 \pm \delta(V))]$  is negligibly small in  $n$ .

For instance, by the previous discussion, the distributional graph properties  $P_{Clique} = (X_{Clique}, s(V), \frac{1}{s(V)})$ , and  $P_{Conn} = (X_{Conn}, 1, 0)$  hold for random graphs. Such properties are called **random graphs properties**.

Given this formalism, **our goal is to obtain pseudo-random graphs preserving all random graphs properties mentioned earlier**. Consequently, whenever the performance of the application utilizing the graphs is known to rely on these properties, we can reliably use our pseudo-random graphs instead.

---

<sup>1</sup>Namely smaller than  $n^{-c}$  for any fixed  $c$  and sufficiently large  $n$ .

### 1.2.3 Pseudo-Random Graphs May Defy Global Properties of Random Graphs

Pseudo-randomness itself implies that random and pseudo-random graphs exhibit similar 'local' properties, namely, properties which can be efficiently tested. However, pseudo-random graphs may fail to retain 'global' properties of random graphs.

For instance, all (but exponentially few) graphs are connected, but it is easy to construct pseudo-random graphs that are always disconnected. Such bold difference does not contradict the pseudo-randomness of these graphs, since any efficient distinguisher is allowed only polynomially many queries while the graph is exponentially large. Thus, such a distinguisher  $D$  cannot decide even simple polynomial time properties in the size of the entire graph such as connectivity, as running even a simple connectivity algorithm takes too much time.

Unfortunately, connectivity is far from being a rare example:

- Consider the chromatic number  $X_{Color}$ , and the independence number  $X_{Indp}$  of undirected graphs. Random graphs on  $V$  vertices rarely have  $X_{Indp} \geq 3 \log V$  or  $X_{Color} \leq \frac{V}{3 \log V}$ . However, we can construct pseudo-random graphs with, say, guaranteed  $X_{Indp} > V^{0.9}$  and  $X_{Color} < V^{0.1}$ .
- Extending our previous connectivity example, random graphs on  $V$  vertices are highly connected - They can rarely be disconnected by omitting less than, say,  $\frac{V}{3}$  vertices. On the other hand, we can construct pseudo-random graphs that are highly disconnected - They can never become connected by adding less than, say,  $V^{0.9}$  edges.

Although our main interest is in graphs, we mention that pseudo-random functions may also fail to retain global properties of truly random functions. For instance, considering the universe of functions  $H_n = \{f : \{0,1\}^n \rightarrow \{0,1\}^n\}$ , we note that

- A random function uniformly taken from  $H_n$  is rarely a permutation. However, Luby and Rackoff [LR] have presented pseudo-random functions which are guaranteed to be permutations.
- A random function uniformly taken from  $H_n$  has an unbiased parity, namely, w.p. exactly  $\frac{1}{2}$  the parity is either 0 or 1. Yet, it is not hard to construct pseudo-random functions all having parity, say, 0.

For more results concerning pseudo-random functions preserving global properties of random functions the reader is referred to [GGN].

### 1.3 Constructing Pseudo-Random Graphs Preserving Properties of Random Graphs

These examples imply that if the performance of the application relies on global properties of random graphs, then arbitrary pseudo-random graphs cannot be used instead. The aim of this Thesis is to provide new constructions of pseudo-random graphs which are guaranteed to preserve random graphs properties. We present three constructions of pseudo-random graphs each preserving several random graphs properties simultaneously. These three constructions can be combined into a single construction which

simultaneously preserve or at least approximate all random graphs properties mentioned earlier.

Our general approach would be to apply modifications to pseudo-random graphs. These modifications would force the desired random graphs properties for the exponentially large graphs in question, but would be mild enough as to preserve pseudo-randomness with respect to distinguishers which can only inspect a polynomial number of edges.

### 1.3.1 Pseudo-Random Graphs Preserving the Clique Number, Independence Number and Chromatic Number of Random Graphs

Random graphs of size  $V$  have clique number and independence number  $\approx 2 \log(V)$  and chromatic number  $\approx \frac{V}{2 \log(V)}$ . The concentration of measure around these typical values is remarkable. Pseudo-Random graphs could be forced to retain exactly the same typical values with a similar concentration of measure.

Namely, let  $E_{s,V}$  denote the expected number of  $s$ -cliques in a random graph of size  $V$ , and let  $s(V) = 2 \log(V) - \Theta(\log \log(V))$  be the maximal  $s$  for which  $E_{s,V} > 1$ . Consider the graph properties  $X_{Clique}$ ,  $X_{Indp}$  and  $X_{Color}$ , assigning to each graph it's clique number, it's independence number and it's chromatic number resp.

Then the following properties are random graphs properties,

- $P_{Clique} = (X_{Clique}, s(V), \frac{1}{s(V)})$ ,
- $P_{Indp} = (X_{Indp}, s(V), \frac{1}{s(V)})$ ,

- $P_{Color} = (X_{Color}, \frac{V}{s(V)}, \frac{1}{\sqrt{\log(V)}})$ .

(Note that  $X_{Clique}, X_{Indp}$  are essentially fixed up to an additive factor of  $\pm 1$ ).

Assuming that pseudo-random functions exist, we provide a construction of pseudo-random graphs simultaneously preserving these three random graphs properties (for an arbitrarily prime number of vertices  $V$ ).

We observe that similar constructions yield pseudo-random graphs with guaranteed  $X_{Indp} > V^c$  and  $X_{Color} < \frac{V}{V^c}$  for arbitrary  $0 < c < 1$ . This provides an extreme evidence of pseudo-random graphs defying properties of random graphs.

### 1.3.2 Pseudo-Random Graphs Preserving Arbitrary Sparse Monotone Properties

Considering graphs of order  $V$ , and an arbitrary  $c < 1$ , random graphs are  $V^c$ -connected, Hamiltonian, and have a perfect matching with overwhelming probability.

To handle these random graphs properties the following terminology is introduced. We say that a series of graphs  $\{g_V\}_{V \in \mathbb{N}}$  where each  $g_V \in G_V$  is an efficiently computable sparse graphs (ECS-graphs) if it is:

- **Efficiently Computable** - Given as input a potential edge  $e = \{u, w\}$ , we can decide in polynomial time in  $\log V$  whether  $e$  appears in  $g_V$ .
- **Sparse** - The fraction  $\frac{E(g_V)}{V^2}$  is negligibly small in  $\log V$ , where  $E(g_V)$  denotes the number of edges in  $g_V$ .

For instance the series  $\{g_V^{Ham}\}_{V \in \mathbb{N}}$ , where each  $g_V^{Ham}$  is the simple Hamiltonian path  $0 \mapsto 1 \mapsto 2 \mapsto \dots \mapsto (V-1) \mapsto 0$ , is clearly sparse and efficiently

computable.

We next define a monotone Boolean graph property  $X$  to be an ECS-property if  $X$  is implied by some ECS-graphs  $\{g_V^{ECS}\}_{V \in \mathbb{N}}$  in the following sense. For any graph  $g \in G_V$  containing a copy of  $g_V^{ECS}$  as a sub-graph, it holds that  $X(g) = 1$ . For instance Hamiltonicity is an ECS-property since any graph on  $V$  vertices containing  $g_V^{Ham}$  as a sub-graph is clearly Hamiltonian.

Assuming that pseudo-random functions exist, we provide pseudo-random graphs preserving any prescribed combination of ECS-s. We thus obtain pseudo-random graphs preserving random graphs properties such as  $V^c$ -connectivity, Hamiltonicity, and having a perfect matching. However, such constructions also provide pseudo-random graphs boldly defying random graphs properties, like pseudo-random graphs having cliques of size  $V^c$  (for a fixed  $c < 1$ ), whereas random graphs rarely have cliques larger than  $2 \log V$ .

### 1.3.3 Pseudo-Random Graphs Approximating the Connectivity Number and the Minimal and Maximal Degrees of Random Graphs

Random graphs on  $V$  vertices are  $(\approx \frac{1}{2}V)$ -regular and  $(\approx \frac{1}{2}V)$ -connected. Namely, with overwhelming probability, each vertex has  $\approx \frac{1}{2}V$  neighbors, and each pair of vertices has  $\approx \frac{1}{2}V$  disjoint paths.

It is unclear whether we can construct pseudo-random graphs preserving exactly this property. However, we can prove that for **arbitrary** pseudo-random graphs, the required connectivity and regularity of degree is approx-

imated in the following sense. For any non-negligible fraction  $\Delta_V \stackrel{\text{def}}{=} \frac{1}{(\log V)^c}$  (where  $c$  is fixed), and for all graphs (apart from a set of graphs having negligible probability), it holds that,

- All but a  $\Delta_V$  fraction of vertices have  $\frac{1}{2}V(1 \pm \Delta_V)$  neighbors.
- All but a  $\Delta_V$  fraction of the pairs of vertices have  $\frac{1}{2}V(1 \pm \Delta_V)$  disjoint paths.

Note that **in addition** to that, the pseudo-random graphs of subsection 1.3.2 are also guaranteed to have (for any prescribed  $d < 1$ ) at least  $V^d$  neighbors for any vertex, and at least  $V^d$  disjoint paths connecting any pair of vertices.

### 1.3.4 Strengthening the Pseudo-Randomness of the Graphs

So far, we have considered computational indistinguishability from random graphs as the basic formalization for the type of randomness that 'random looking' graphs should retain. Suppose that in addition to that, we wish our graphs to be  $(n^k)$ -wise independent in the sense that the distribution of any  $n^k$  edges is uniform (or at least close to uniform). This could be achieved for any prescribed  $k$  as follows.

First, several efficient constructions for  $(n^k)$ -wise independent Boolean functions were given over the years (e.g. [J]). Clearly, representing graphs by such Boolean functions immediately gives  $(n^k)$ -wise independent graphs. Next, when arbitrary  $(n^k)$ -wise independent functions are XORed with arbitrary pseudo-random functions the resulting functions are both  $(n^k)$ -wise independent as well as pseudo-random [IW]. Finally, we prove that when

pseudo-random functions which are also  $(n^k)$ -wise independent are used as the basis for our constructions (rather than using arbitrary pseudo-random functions), then the resulting graphs are almost  $(n^k)$ -wise independent in the sense that the distribution of any  $n^k$  edges is statistically close to uniform.

## 1.4 Roadmap

In chapter 2 preliminary notations and definitions are provided. In chapter 3 we formally define pseudo-random functions. We also provide some basic tools concerning pseudo-randomness. In chapter 4 we formally define pseudo-random graphs. We then present constructions of pseudo-random graphs preserving properties of random graphs. Several definitions given in the introduction are repeated in chapters 3 and 4 for self containment. We summarize and suggest open problems in chapter 5. The appendix presents proofs for some random graphs properties.

# Chapter 2

## Preliminaries

This chapter provides basic definitions, notations, and tools concerning strings, functions, graphs, algorithms, and concentration of measure.

### 2.1 Strings and Functions

- **Strings Concatenation** - The concatenation of the strings  $s_1$  and  $s_2$  is denoted by  $s_1||s_2$ .
- **Integers and Binary Representation**- We identify each integer  $m$  with its binary representation of length  $|m| \stackrel{\text{def}}{=} \lceil \log(m) \rceil$ .
- **Length Functions** - A function  $\ell : \mathbb{N} \rightarrow \mathbb{N}$  is called a length function if  $\ell$  is strictly monotone, polynomially bounded and can be computed on input  $n$  in polynomial time in  $|n|$ .
- **Vanishing Functions** - A function  $f : \mathbb{N} \rightarrow \mathbb{R}$  is called vanishing if  $f(m) \rightarrow 0$  as  $m \rightarrow \infty$ .

- **Negligible Functions** - A (possibly) partial function  $f : \mathbb{N} \rightarrow \mathbb{R}$  with an infinite domain is called negligible if for any  $c > 0$  and all but finitely many values of  $m \in \mathbb{N}$ ,  $f(m) < \frac{1}{m^c}$ .

## 2.2 Computational Notions and Notations

- **Algorithms (with Oracle Access)** - Deterministic algorithms (or randomized algorithms) are defined by deterministic (resp. randomized) Turing machines. Algorithms with oracle access to functions are given by oracle Turing machines.
- **Efficient Algorithms** - Invariably we denote the security parameter by  $n$ . An algorithm  $A$  is efficient if on inputs of length  $n$ ,  $A$  runs in polynomial time in  $n$ .
- **Hard to Invert Functions** - Let  $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$ . Let  $A$  be a (possibly randomized) algorithm, and let  $C = \{C_n\}_{n \in \mathbb{N}}$  be a series of circuits. Let  $\Delta_n^A$  and  $\Delta_n^C$  denote the probabilities that  $A(f(x))$  and  $C_n(f(x))$  resp. are in the pre-image of  $f(x)$ . This probability is uniformly taken over inputs  $x$  of length  $n$  as well as on the possible internal coin tosses of  $A$  and  $C_n$ . We say that  $f$  is hard to invert if  $\Delta_n^A$  is negligible in  $n$  for any efficient algorithm  $A$ , and is hard to invert w.r.t. circuits if  $\Delta_n^C$  is negligible in  $n$  whenever  $C_n$  has size polynomial in  $n$ .
- **One-Way functions** - A function  $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$  is one-way if it is efficiently computable but hard to invert, and is one-way w.r.t. circuits if it is efficiently computable but hard to invert w.r.t. circuits.

## 2.3 Graph Notation and Random Graphs

- **Connectivity** - A graph  $g$  having more than  $k$  vertices is said to be  $k$ -connected if it cannot be disconnected by omitting less than  $k$  vertices. For  $k > 2$ ,  $k$ -connectivity is equivalent to having at least  $k$  vertex-disjoint paths between any pair of vertices. The vertex-connectivity of  $g$ , denoted  $\kappa(g)$ , is the maximal  $k$  s.t.  $g$  is  $k$ -connected.
- **Matching** - A graph  $g$  on  $V$  vertices has a perfect matching if all its vertices (except one vertex in case  $V$  is odd) are covered by  $\lfloor \frac{V}{2} \rfloor$  disjoint edges.
- **Hamiltonicity** - An Hamiltonian cycle in a graph  $g$  is a cycle passing through each vertex exactly once. If  $g$  contains an Hamiltonian cycle, then  $g$  is said to be an Hamiltonian graph.
- **Cliques and Independent Sets** - A clique in a graph  $g$  is a set of vertices all connected to each other, and an independent set in  $g$ , is a set of vertices all disconnected from each other. The clique number of  $g$ , denoted  $\omega(g)$  is the maximal size of a clique in  $g$ , and the independence number, denoted  $\alpha(g)$  is the maximal size of an independent set in  $g$ .
- **Graph Coloring** - A graph  $g$  is  $k$ -colorable if its vertices can be colored with  $k$  colors, such that adjacent vertices have distinct colors. The chromatic number of a graph  $g$ , denoted  $\chi(g)$ , is the minimal number  $k$  such that  $g$  is  $k$ -colorable.
- **Random Graphs** - Random graphs of order  $V$  refer to the uniform distribution over  $G_V$ , the set of all simple, labeled, undirected graphs

on the vertices  $\{0, 1, \dots, V - 1\}$ . This distribution is identical to the distribution where each edge is independently picked w.p.  $\frac{1}{2}$ .

- An arbitrary condition  $C$  is said to **hold for a random graph** if the probability that a random graph of order  $V$  maintains condition  $C$  is  $1 - \epsilon(V)$ , where  $\epsilon(V)$  is vanishing.
- Accordingly, condition  $C$  is said to hold for a random graph with overwhelming probability if  $\epsilon(V)$  is exponentially small in  $\log V$ .

## 2.4 Basic Probability Tools

- **Notation** - For 2 random variables  $X, Y$  taken over the same probability space we let  $\mathbb{E}(X), \text{var}(X)$  and  $\text{cov}(X, Y)$  denote the expectation of  $X$ , the variance of  $X$  and the covariance of  $X, Y$  respectively.
- **Markov's Inequality** - For a positive r.v.  $X$ ,  

$$\Pr[X \geq \Delta] \leq \frac{\mathbb{E}(X)}{\Delta}.$$
- **Chebyshev's Inequality** - For any r.v.  $X$ ,  

$$\Pr[|X - \mathbb{E}(X)| \geq \Delta] \leq \frac{\text{var}(X)}{\Delta^2}.$$
- **Chernoff's Inequality (Additive Version)**- For  $X = \sum_{i=1}^m X_i$  where  $X_1, \dots, X_m$  are independent Bernoulli trials each w.p. of success  $p$ ,  

$$\Pr[X \leq pm(1 - \Delta)] \leq e^{-\frac{1}{p}\Delta^2V},$$

$$\Pr[X \geq pm(1 + \Delta)] \leq e^{-\frac{1}{p}\Delta^2V}.$$

# Chapter 3

## Pseudo-Random Functions - Definitions and Basic Tools

### 3.1 Pseudo-Random Functions - Discussion

Suppose that an algorithm is assumed to have oracle access to a function in the universe  $F_n = F_n^{\ell_1, \ell_2} = \{f | f : \{0, 1\}^{\ell_1(n)} \rightarrow \{0, 1\}^{\ell_2(n)}\}$ , where  $\ell_1, \ell_2$  are arbitrary length functions. This means that an algorithm  $A$  can query a function  $f$  on inputs  $x$  of its choice, and receive the value  $f(x)$  in unit time. Next, consider the case where **random functions** uniformly taken from  $F_n$  are required. However, as the cardinality of  $F_n$  is  $2^{\ell_2(n)2^{\ell_1(n)}}$ , the number of random bits required to specify a function  $f \in F_n$  is  $\ell_2(n)2^{\ell_1(n)}$  which is usually much more than we can afford.

We thus seek an efficient method of utilizing "random looking" functions from  $F_n$ . Goldreich, Goldwasser and Micali [GGM] have introduced pseudo-random functions which can be picked, stored and evaluated efficiently and

can faithfully replace truly random functions in a variety of cryptographic applications. Explicit constructions of pseudo-random functions were provided in [GGM] assuming that one-way functions exist.

The basic idea in [GGM] is that the pseudo-random functions are taken from a small subset  $\bar{F}_n \subset F_n$ , where each function in  $\bar{F}_n$  is succinctly represented by a string  $s$  called **seed** of length polynomial in  $n$ . Picking (and storing) a function from  $\bar{F}_n$  is done using an efficient generator  $G$ . On input  $1^n$ ,  $G$  simply outputs a random seed  $s$ , which specifies a unique function  $f_s \in \bar{F}_n$ . An efficient evaluator  $E$  can later on evaluate the specified function  $f_s$  using the seed  $s$ . Namely, for any input  $x$  of length  $\ell_1(n)$ ,  $E(s, x) = f_s(x)$ . These requirements are captured in the definition of **efficiently computable ensemble of functions** which we shortly present.

To capture the pseudo-randomness of these functions, consider the distribution  $\mathcal{F}_n$  of the functions which  $G$  generates. Here pseudo-randomness means that no probabilistic algorithm  $D$  running in polynomial time in  $n$  can distinguish a function sampled from  $\mathcal{F}_n$  from a function uniformly taken over  $F_n$ . The algorithm  $D$  may query a function on inputs of its choice but only polynomially many queries are allowed.

## 3.2 Pseudo-Random Functions - Definitions

We start by defining efficiently computable functions ensembles.

**Definition 3.1 Functions Ensemble** - *Let  $\ell_1, \ell_2$  be arbitrary length functions, and let  $I \subseteq \mathbb{N}$  be an infinite index set. A functions ensemble with length parameters  $\ell_1, \ell_2$  and index set  $I$  is a sequence  $\mathcal{F} = \{\mathcal{F}_n\}_{n \in I}$  of distri-*

butions, where each distribution  $\mathcal{F}_n$  is taken over the universe of functions  $F_n = \{f|f : \{0,1\}^{\ell_1(n)} \rightarrow \{0,1\}^{\ell_2(n)}\}$ . If each  $\mathcal{F}_n$  is uniformly distributed over  $F_n$ , then the ensemble is called the uniform ensemble.

**Definition 3.2 Efficiently Computable Functions Ensemble** - A functions ensemble  $\mathcal{F}$  is efficiently computable if there exists a series of subsets  $\bar{F}_n \subset F_n$ , and two efficient probabilistic algorithms, a generator  $G$  and an evaluator  $E$ , s.t. for all  $n \in I$ :

- **Efficient Indexing:**  $\bar{F}_n = \{f_s\}_{s \in \text{Range}(G(1^n))}$ , namely, we refer to the output  $s = G(1^n)$  as the index of the function  $f_s$ .
- **Efficient Sampling:** The distributions  $f_{(G(1^n))}$  and  $\mathcal{F}_n$  are identical.
- **Efficient Evaluation:** For any index  $s \in \text{Range}(G(1^n))$ , and for any input  $x$  of length  $\ell_1(n)$ , it holds that  $E(s, x) = f_s(x)$ .

We proceed by defining the notion of pseudo-randomness. Let  $D$  be a probabilistic oracle machine. Consider some fixed input  $x$ , and some distribution  $\mathcal{F}$  on functions from finite strings to finite strings. The output of  $D$  on  $x$ , using internal random coin tosses and oracle access to a function randomly taken from  $\mathcal{F}$  is a random variable denoted by  $D^{\mathcal{F}}(x)$ .

**Definition 3.3 Distinguishing Advantage** - For a probabilistic oracle machine  $D$ , and two functions ensembles  $\mathcal{F} = \{\mathcal{F}_n\}_{n \in I}$  and  $\mathcal{F}' = \{\mathcal{F}'_n\}_{n \in I}$ , the advantage  $D$  has in distinguishing  $\mathcal{F}$  from  $\mathcal{F}'$  is the function

$$\Delta_D^{\mathcal{F}, \mathcal{F}'} : I \rightarrow \mathbb{R}, \quad \Delta_D^{\mathcal{F}, \mathcal{F}'}(n) = |\Pr [D^{\mathcal{F}_n}(1^n) = 1] - \Pr [D^{\mathcal{F}'_n}(1^n) = 1]|.$$

Probabilities are taken over the distributions  $\mathcal{F}_n, \mathcal{F}'_n$  as well as on the internal coin tosses of the distinguisher  $D$ .

**Definition 3.4 Computational Indistinguishability** - *Two functions ensembles  $\mathcal{F}$  and  $\mathcal{F}'$  having the same index set  $I$ , are computationally indistinguishable if for any efficient oracle machine  $D$ , the advantage  $D$  has in distinguishing  $\mathcal{F}$  from  $\mathcal{F}'$  is negligible in  $n$ .*

We can now easily define pseudo-random functions.

**Definition 3.5 Pseudo-Random Functions** - *A functions ensemble  $\mathcal{F}$  is called pseudo-random if it is:*

- *Efficiently computable.*
- *Computationally indistinguishable from the uniform ensemble  $\mathcal{F}'$  with the same length parameters and the same index set.*

### 3.3 Mild Modifications Preserve Pseudo-Randomness

Our next goal is to construct pseudo-random functions having some prescribed properties. The general approach would be to apply some modifications to arbitrary pseudo-random functions. These modifications would enforce the desired properties, but would be mild enough as to preserve the pseudo-randomness of the original functions. Applying mild modifications is done by combining a given construction of pseudo-random functions with another construction of efficiently computable functions.

### 3.3.1 Boolean Combinations of Efficiently Computable Ensembles

Suppose we wish to obtain an efficiently computable ensemble by taking, say, the exclusive or of two efficiently computable ensembles  $\mathcal{F}_1 = \{\mathcal{F}_n^1\}_{n \in I}$ ,  $\mathcal{F}_2 = \{\mathcal{F}_n^2\}_{n \in I}$ , having the same length parameters. By this, we mean that on input  $1^n$  we wish to pick two functions  $f_1, f_2$  according to distributions  $\mathcal{F}_n^1, \mathcal{F}_n^2$  resp. and that later on, on any input  $x$  of appropriate length we wish to compute  $f_1(x) \oplus f_2(x)$ . This could be easily achieved using the generators  $G_1, G_2$  and evaluators  $E_1, E_2$  defining  $\mathcal{F}_1$  and  $\mathcal{F}_2$  resp. On input  $1^n$  we simply use  $G_1, G_2$  to generate the two seeds  $s_1, s_2$  representing  $f_1$  and  $f_2$ , and later on, on input  $x$  we simply use  $E_1, E_2$  to evaluate  $f_1(x)$  and  $f_2(x)$ . Formally,

**Definition 3.6 Boolean Combinations of Efficiently Computable Ensembles** - *Consider two efficiently computable ensembles  $\mathcal{F}_1, \mathcal{F}_2$  (with the same length parameters  $\ell_1, \ell_2$  and the same index set  $I$ ), given by the generator and evaluator pairs  $(G_1, E_1)$  and  $(G_2, E_2)$  respectively. Let  $B \in \{\cup, \cap, \oplus\}$ . Then, the functions ensemble  $\mathcal{F} = B(\mathcal{F}_1, \mathcal{F}_2)$  is given by the generator  $G(1^n) = G_1(1^n) || G_2(1^n)$  and by the evaluator  $E(s_1 || s_2, x) = B(E_1(s_1, x), E_2(s_2, x))$ .*

Clearly,  $\mathcal{F}$  is an efficiently computable ensemble (with the same length parameters and the same index set).

### 3.3.2 Mild Modifications of Efficiently Computable Ensembles

Consider a Boolean combination of two efficiently computable ensembles  $\mathcal{F}$  and  $\mathcal{F}^{Mild}$ . We think of this combination as a process in which we first pick

an initial function  $f_1$  from  $\mathcal{F}$ , and then we mildly modify  $f_1$  by combining it with a function  $f_2$  we pick from  $\mathcal{F}^{Mild}$ . The mildness of the modification means that for any  $f_1$  and any input  $x$  the modification rarely changes  $f_1(x)$ . Formally,

**Definition 3.7 Mild Modifications of Efficiently Computable Ensembles**

- Consider the Boolean combination  $\mathcal{F}^{Modf} = \{\mathcal{F}_n^{Modf}\}_{n \in I} = B(\mathcal{F}, \mathcal{F}^{Mild})$ .

Suppose these ensembles are given by the generator and evaluator pairs  $(G^{Modf}, E^{Modf})$ ,  $(G, E)$  and  $(G^{Mild}, E^{Mild})$  respectively. We say that  $\mathcal{F}^{Modf}$  is a mild modification of  $\mathcal{F}$ , if for any  $n \in I$ , any seed  $s_1 \in \text{Range}(G(1^n))$ , and any input  $x$  of appropriate length, it holds that  $\Pr [E^{Modf}(s_1 || s_2, x) \neq E(s_1, x)]$  is negligibly small in  $n$ . Here the probability is taken only on the random generation of the modification seed  $s_2 \stackrel{\text{def}}{=} G^{Mild}(1^n)$ .

**3.3.3 Mild Modifications Preserve Pseudo-Randomness**

We may now describe our primary tool.

**Lemma 3.1 Mild Modifications Preserve Pseudo-Randomness** - Any mild modification  $\mathcal{F}^{Modf} = B(\mathcal{F}^{Psd}, \mathcal{F}^{Mild})$ , of a pseudo-random ensemble  $\mathcal{F}^{Psd}$  is also a pseudo-random ensemble.

**Proof:** As computational indistinguishability is transitive, it suffices to achieve computational indistinguishability between  $\mathcal{F}^{Modf}$  and  $\mathcal{F}^{Psd}$ . We therefore fix an arbitrary distinguisher  $D$ , restricted to polynomially many queries, and prove that it has only a negligible advantage in distinguishing  $\mathcal{F}^{Modf}$  from  $\mathcal{F}^{Psd}$ . We impose no further computational limitations on  $D$ ,

and in particular  $D$  may even be non-uniform (Namely,  $D$  may be given by a family of circuits  $\{D_n\}_{n \in \mathbb{N}}$ ). Consequently, we can assume w.l.o.g. that  $D$  is deterministic, as we can always hard-wire the best distinguishing sequence of coin tosses into each circuit  $D_n$ .

We start with some useful notation. Suppose the three ensembles are given by the generator and evaluator pairs  $(G^{Psd}, E^{Psd})$ ,  $(G^{Mild}, E^{Mild})$  and  $(G^{Modf}, E^{Modf})$ . Fixing  $n$ , let  $f$  denote some **fixed** function which was generated by  $G^{Psd}(1^n)$ . Let  $f_{s_2}$  denote the modification of  $f$  when  $G^{Mild}(1^n) = s_2$ . Namely,  $f_{s_2}(x) \stackrel{\text{def}}{=} B(f(x), E^{Mild}(s_2, x))$ . Finally, we say that  $x$  is a modified input if  $f(x) \neq f_{s_2}(x)$ .

We prove the desired indistinguishability, by fixing  $n$  and an arbitrary initial function  $f$  as above, and showing that the probability that  $D$  queries a modified input  $x$  when  $D$  is given oracle access to  $f_{s_2}$ , is negligible in  $n$ . **This probability is taken only over the generation of the modification seed  $s_2 = G^{Mild}(1^n)$ .**

Next, denote by  $q$  the number of queries of  $D$  to  $f_{s_2}$ , and assume w.l.o.g. that the modification seed  $s_2$  is uniformly taken from some set  $S$ . Let  $S_{Good}^j$  be the set of all good modification seeds  $s_2$  s.t.  $D$  **does not query** upon any modified input during the first  $j$  queries to  $f_{s_2}$ . Clearly  $S_{Good}^q \subseteq \dots \subseteq S_{Good}^1 \subseteq S_{Good}^0 = S$ . Our goal is to show that almost all seeds "pass" all  $q$  queries, or in other words that  $|S_{Good}^q|$  is almost as large as  $|S|$ . To this end, recall that the definition of a mild modification implies a negligible upper-bound  $p = p(n)$  on the probability that  $f(x) \neq f_{s_2}(x)$  for any fixed input  $x$ . Therefore we can show that  $|S_{Good}^q|$  is large by proving by induction on  $j$  that  $|S_{Good}^j| \geq |S| \cdot (1 - j \cdot p)$ .

For  $j = 0$  the claim is trivial so let's assume for  $j - 1$  and prove for  $j$ . Let  $x_j$  denote the  $j$ 'th query of  $D$ , and let  $B_{x_j}$  denote the set of all "bad" modification seeds  $s_2$  s.t.  $f(x_j) \neq f_{s_2}(x_j)$ . Note that  $|B_{x_j}| \leq p \cdot |S|$ , and that

$$\begin{aligned} S_{Good}^j &= S_{Good}^{(j-1)} \setminus B_{x_j}. \text{ Consequently,} \\ |S_{Good}^j| &\geq |S_{Good}^{(j-1)}| - |B_{x_j}| \geq \\ &|S| \cdot (1 - (j-1) \cdot p) - (p \cdot |S|) = \quad (\text{By hypothesis of induction}) \\ &= |S| \cdot (1 - j \cdot p), \end{aligned}$$

which completes the induction proof.

Finally,

$$\begin{aligned} \Pr [D \text{ queries upon a modified input in any of its } q \text{ queries to } f_{s_2}] &= \\ &= 1 - \Pr [s_2 \in S_{Good}^q] = \\ &= 1 - \frac{|S_{Good}^q|}{|S|} \leq \frac{q \cdot p \cdot |S|}{|S|} = q \cdot p. \end{aligned}$$

Since  $p = p(n)$  is negligible and  $q = q(n)$  is polynomially bounded, then  $p \cdot q$  is negligible. This completes the entire proof.

■ (Lemma 3.1).

We conclude this section with the following Lemma [IW].

**Lemma 3.2** *Let  $\mathcal{F} = \oplus(\mathcal{F}^{Psd}, \mathcal{F}^{Eff})$ , where  $\mathcal{F}^{Psd}$  is a pseudo-random functions ensemble and  $\mathcal{F}^{Eff}$  is an efficiently computable functions ensemble. Then  $\mathcal{F}$  is also a pseudo-random functions ensemble.*

**Proof** - Indeed, assume towards contradiction, that some efficient algorithm  $D$  distinguishes  $\mathcal{F}$  from the uniform ensemble  $\mathcal{F}^{Rndm}$  with a non-negligible advantage. Now to distinguish  $\mathcal{F}^{Psd}$  from  $\mathcal{F}^{Rndm}$  we construct a similar algorithm  $D'$  as follows. On input  $1^n$ , and given oracle access to  $f$ ,  $D'$  simulates the execution of  $D^{\bar{f}}(1^n)$  and replies as  $D$  dose, where  $\bar{f} = \oplus(f, f_n^{Eff})$ .

Note that the distribution of the output  $D'^{\mathcal{F}_n^{Psd}}(1^n)$  is identical to the distribution of the output  $D^{\oplus(\mathcal{F}_n^{Psd}, \mathcal{F}_n^{Eff})}(1^n) \equiv D^{\mathcal{F}_n}(1^n)$ . Similarly, the distribution of the output  $D'^{\mathcal{F}_n^{Rndm}(1^n)}$  is identical to the distribution of the output  $D^{\oplus(\mathcal{F}_n^{Rndm}, \mathcal{F}_n^{Eff})}(1^n) \equiv D^{\mathcal{F}_n^{Rndm}}(1^n)$ .

Consequently, the advantage  $D$  achieves in distinguishing  $\mathcal{F}$  from  $\mathcal{F}^{Rndm}$  translates to the same advantage  $D'$  has in distinguishing  $\mathcal{F}^{Psd}$  from  $\mathcal{F}^{Rndm}$  which contradicts the definition of pseudo-randomness.

■ (Lemma 3.2).

# Chapter 4

## Pseudo-Random Graphs Preserving Properties of Random Graphs

### 4.1 Graphs Notation

We invariably consider only simple, labeled, undirected graphs. Single graphs are denoted by small letters  $g$ , sets of graphs by capital letters  $G$ , and distributions over sets of graphs by calligraphic letters  $\mathcal{G}$ . The order (or size) of a graph  $g$  is its number of vertices denoted by  $V = V(g)$ , whereas the number of edges appearing in  $g$  is denoted by  $E = E(g)$ . The set of all graphs on vertices  $\{0, 1, \dots, V - 1\}$  is denoted by  $G_V$ . Vertices are denoted by small letters  $u, v, w$ , and the neighbors-set of a vertex  $u$  is denoted by  $\Gamma(u)$ . Finally, for a permutation  $\pi$  defined over the vertices of a graph  $g \in G_V$ , the abused notation  $\pi(g)$  stands for the graph over the same vertices, where the

edge  $\{u, w\}$  appears in  $\pi(g)$  iff the edge  $\{\pi^{-1}(u), \pi^{-1}(w)\}$  appears in  $g$ .

## 4.2 Pseudo-Random Graphs - Discussion

Suppose that some application requires huge random graphs of size  $V = 2^n$ . Recall that random graphs of order  $V$  are uniformly taken from  $G_V$ . The graphs in  $G_V$  are canonically represented by the Boolean functions  $F_n = \{f|f : \{0, 1\}^{2n} \rightarrow \{0, 1\}\}$ , as follows. First denote  $2n$ -bit strings by  $u||v$  where  $u$  is the  $n$ -bit prefix and  $v$  is the  $n$ -bit suffix. Next, identify each integer  $i \in \{0, 1, \dots, V - 1\}$  with its binary representation. Now, a specific graph  $g \in G_V$  is represented by a function  $f \in F_n$ , where for  $u < v$  the edge  $\{u, v\}$  appears in  $g$  iff  $f(u||v) = 1$ .

We note that the distribution of graphs represented by a function uniformly taken from  $F_n$ , is identical to the required distribution of random graphs. Therefore to utilize random graphs it suffices to acquire oracle access to a random Boolean function. To achieve efficient sampling, storage and evaluation of these huge graphs, the random Boolean functions are replaced by pseudo-random Boolean functions, resulting in what we call pseudo-random graphs. In general, to generate graphs of order  $V$ , where  $V$  is not a perfect power of 2, we simply set  $n = n(V) = \lceil \log(V) \rceil$ , so  $n$  is the minimal integer such that  $V \leq 2^n$ .

### 4.3 Pseudo-Random Graphs - Definitions

Formally, we define efficiently computable graphs as efficiently computable functions with length parameters suitable for representing graphs. We then identify pseudo-random graphs with the functions representing the graphs (rather than with the graphs themselves).

**Definition 4.1 Graphs Ensemble** - *A functions ensemble  $\mathcal{F} = \{\mathcal{F}_{n(V)}\}_{V \in I}$  is said to be a graphs ensemble if for each  $V \in I$ ,  $n(V) = \lceil \log(V) \rceil$ .*

**Definition 4.2 Efficiently Computable Graphs Ensemble** - *A functions ensemble  $\mathcal{F}$  is said to be an efficiently computable graphs ensemble if it is an efficiently computable ensemble and a graphs ensemble.*

**Definition 4.3 Pseudo-Random Graphs** - *A functions ensemble  $\mathcal{F}$  is said to be a pseudo-random graphs ensemble if it is a pseudo-random ensemble and a graphs ensemble.*

Note that given our fixed representation scheme of graphs by functions, we can identify each function  $f \in F_{n(V)}$  with the unique graph  $g \in G_V$  it represents. Therefore each distribution  $\mathcal{F}_{n(V)}$  over the universe of functions  $F_{n(V)}$  induces a distribution  $\mathcal{G}_V$  over the universe of graphs  $G_V$ . We allow ourselves to identify the distributions  $\mathcal{F}_{n(V)}$  and  $\mathcal{G}_V$ . Accordingly, when we discuss the distribution of a graph property (e.g. the chromatic number) over pseudo-random graphs, we actually refer to the series of distributions this r.v. has over the  $\mathcal{G}_V$ -s.

## 4.4 Random Graphs' Properties

Our goal is to construct pseudo-random graphs maintaining properties of random graphs, such as being connected, or having chromatic number  $\approx \frac{V}{2 \log V}$  with overwhelming probability.

To discuss properties of random graphs (which are properties of distributions), we must first consider properties of single graphs, called graph properties. A graph property is a function assigning a real number to each graph. For instance, the graph property  $X_{Color}$  assigns to each graph  $g$  its chromatic number, whereas the graph property  $X_{Conn}$  assigns to  $g$  the value 1 if  $g$  is connected and the value 0 otherwise.

We next consider the distribution of graph properties over random graphs. For instance, it is well known that a random graph  $\mathcal{G}$  on  $V$  vertices obtains the following requirements with probability  $1 - 2^{-\Omega(n(V))}$

- Having maximal degree  $X_{MaxDeg}(\mathcal{G}) = \frac{1}{2}V(1 \pm 2\sqrt{\frac{\log V}{V}})$ .
- Having chromatic number  $X_{Color}(\mathcal{G}) = \frac{V}{s(V)}(1 \pm \frac{1}{\sqrt{\log(V)}})$   
where  $s(V) \approx 2 \log V$  is easy to compute for all  $V$ .
- Being connected - Having  $X_{Conn}(\mathcal{G}) = 1(1 \pm 0)$ .

We note that for each graph property  $X \in \{X_{MaxDeg}, X_{Color}, X_{Conn}\}$ , the typical behavior of a random graph  $\mathcal{G}$  of order  $V$  was expressed as having  $\Pr[X(\mathcal{G}) \neq \mu_X(V)(1 \pm \delta_X(V))]$  negligibly small in  $n(V)$ . Here  $\mu_X(V)$  is the typical value of  $X(\mathcal{G})$  and  $\delta_X(V)$  tends to 0. Inspired by this, we define distributional graph properties which can either hold or not hold for arbitrary ensembles of distributions on graphs.

**Definition 4.4 Distributional Graph Properties** - *The triplet  $P = (X, \mu, \delta)$  where  $X$  is a graph property and  $\mu, \delta : \mathbb{N} \rightarrow \mathbb{R}$ , is called a distributional graph property if  $\delta(V) \xrightarrow{V \rightarrow \infty} 0$ .*

**Definition 4.5 Properties Holding for Distributions on Graphs** - *A distributional graph property  $P = (X, \mu, \delta)$  is said to hold for a graphs ensemble  $\mathcal{G}' = \{\mathcal{G}'_V\}_{V \in I}$ , if for any  $V \in I$ ,  $\Pr[X(\mathcal{G}'_V) \neq \mu(V) \cdot (1 \pm \delta(V))]$  is negligibly small in  $n(V)$ .*

**Definition 4.6 Properties of Random Graphs** - *A distributional graph property  $P = (X, \mu, \delta)$  is a property of random graphs if it holds for the ensemble of uniform distributions over  $G_V$ .*

For instance, by the previous discussion, the following distributional graph properties are properties of random graphs.

- $P_{Clique} = (X_{Clique}, s(V), \frac{1}{s(V)})$ .
- $P_{MaxDeg} = (X_{MaxDeg}, \frac{1}{2}V, 2\sqrt{\frac{\log V}{V}})$ .
- $P_{Conn} = (X_{Conn}, 1, 0)$ .

In the following sections we present 3 constructions of pseudo-random graphs, where each construction simultaneously preserves several random graphs properties. These constructions can be combined into a single construction simultaneously preserving or at least approximating all random graphs properties mentioned in the introduction. As mentioned earlier, our general approach would be to modify arbitrary pseudo-random graphs. These modifications would enforce the desired properties, but would be mild enough

as to preserve the pseudo-randomness of the original graphs. The 3 constructions handle the following random graphs properties.

- Having Clique number and independence number  $\approx 2 \log(V)$  and having chromatic number  $\approx \frac{V}{2 \log(V)}$ .
- Being Hamiltonian, having a perfect matching, and obtaining  $V^c$ -connectivity for fixed  $c < 1$  (sparse monotone properties).
- Achieving  $(\approx \frac{1}{2}V)$ -regularity and  $(\approx \frac{1}{2}V)$ -connectivity.

## 4.5 Pseudo-Random Graphs Preserving the Clique Number, Independence Number and Chromatic Number of Random Graphs

Random graphs of order  $V$  have clique number  $\omega(\mathcal{G}) \approx 2 \log(V)$ , independence number  $\alpha(\mathcal{G}) \approx 2 \log(V)$  and chromatic number  $\chi(\mathcal{G}) \approx \frac{V}{2 \log(V)}$ . The concentration of measure around these typical values is remarkable. We construct pseudo-random graphs which retain the same typical values of  $\omega(\mathcal{G}), \alpha(\mathcal{G})$  and  $\chi(\mathcal{G})$  with a similar concentration of measure.

To obtain more quantitative statements, let  $E_{s,V}$  denote the expected number of  $s$ -cliques in a random graph of size  $V$ , and let  $s(V)$  be the maximal  $s$  for which  $E_{s,V} > 1$ . It is not hard to verify that  $s(V) = 2 \log(V) - \Theta(\log \log(V))$  and that  $s(V)$  is efficiently computable given  $V$  in binary.

**Theorem 4.1 (Bollobás '88) Clique Number, Independence Number and Chromatic Number of Random Graphs** - *Consider the graph properties  $\omega, \alpha$  and  $\chi$ , assigning to each graph its clique number, its independence number and its chromatic number respectively.*

*Then the following properties are random graphs properties,*

- $P_{Clique} = (\omega, s(V), \frac{1}{s(V)})$ .
- $P_{Indp} = (\alpha, s(V), \frac{1}{s(V)})$ .
- $P_{Color} = (\chi, \frac{V}{s(V)}, \frac{1}{\sqrt{\log(V)}})$ .

**Remark 4.1** Note that  $\omega$  and  $\alpha$  are essentially fixed (up to an additive factor of  $\pm 1$ ).

For proofs, the interested reader is referred to [AS].

Assuming that pseudo-random functions exist, we construct pseudo-random graphs simultaneously preserving these three random graphs properties (for an arbitrarily prime number of vertices  $V$ ). We first provide a construction of pseudo-random preserving the required clique number and independence number, and later modify this construction to preserve the required chromatic number as well. The following definition is useful.

**Definition 4.7  $k(n)$ -wise Independent Functions** - *Let  $k : \mathbb{N} \rightarrow \mathbb{N}$ . An ensemble of Boolean functions  $\mathcal{F} = \{\mathcal{F}_n\}_{n \in I}$  is called  $k(n)$ -wise independent, if for all  $n \in I$ , and for arbitrary distinct inputs  $x_1, \dots, x_{k(n)}$ , it holds that the values  $\mathcal{F}_n(x_1), \dots, \mathcal{F}_n(x_{k(n)})$  are uniformly distributed over  $\{0, 1\}^{k(n)}$ .*

Recall that Joffe has provided efficiently computable  $k(n)$ -wise independent functions for any function  $k(n)$  polynomially bounded in  $n$  [J]. Such functions clearly give  $k(n)$ -wise independent graphs in the sense that any set of up to  $k(n)$  edges are independently picked w.p.  $\frac{1}{2}$ .

**Construction 4.1 Pseudo-Random Graphs  $\mathcal{G}^{Indp}$  Preserving the Clique Number and Independence Number of Random Graphs -** Let  $\mathcal{G}^{Psd}$  be some pseudo-random graphs, and let  $\mathcal{G}^{k-wise}$  be some efficiently computable graphs which are  $(4 \log^2 V)$ -wise independent and have the same length parameters as  $\mathcal{G}^{Psd}$ . Then the graphs ensemble  $\mathcal{G}^{Indp}$  is defined by  $\mathcal{G}^{Indp} \stackrel{\text{def}}{=} \oplus(\mathcal{G}^{Psd}, \mathcal{G}^{k-wise})$ .

**Claim 4.1** Assuming that one-way functions exist, the ensemble  $\mathcal{G}^{Indp}$  provides pseudo-random graphs preserving the random graphs properties,

- $P_{Clique} = (\omega, s(V), \frac{1}{s(V)})$ .
- $P_{Indp} = (\alpha, s(V), \frac{1}{s(V)})$ .

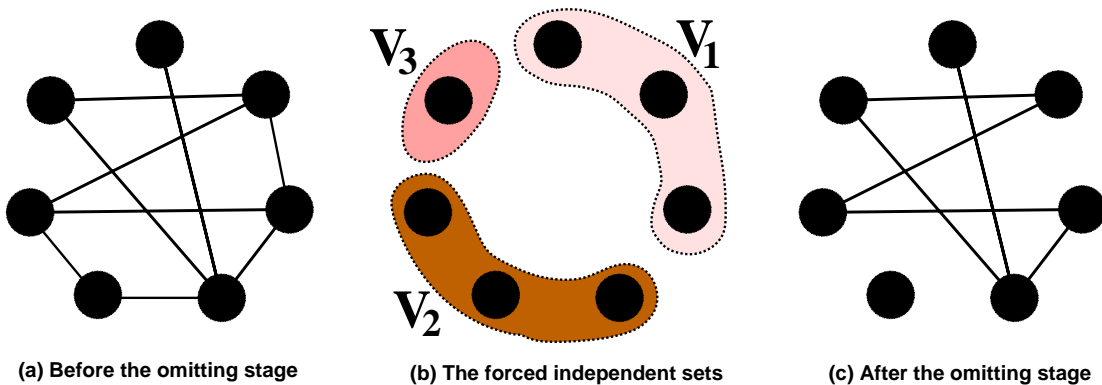
**Proof** - First, note that  $\mathcal{G}^{Indp}$  are  $(4 \log^2 V)$ -wise independent functions, since it's easy to verify that xoring  $k(n)$ -wise independent functions with arbitrary functions, yields  $k(n)$ -wise independent functions.

Next, we note that the analysis in [AS] proving that random graphs obtain properties  $P_{Clique}, P_{Indp}$  uses only the fact that for random graphs, any set of up to  $2\binom{s(V)}{2}$  edges are independently picked w.p.  $\frac{1}{2}$ , a requirement which also holds for  $(4 \log^2 V)$ -wise independent graphs (since  $4 \log^2 V \geq 2\binom{s(V)}{2}$ ).

Finally,  $\mathcal{G}^{Indp}$  are pseudo-random functions, since lemma 3.2 implies that xoring pseudo-random functions with arbitrary efficiently computable functions yields pseudo-random functions. ■

We finally construct pseudo-random graphs simultaneously preserving **all three** random graphs properties, for arbitrary prime order  $V$ . We start with an **informal description**. To construct a graph of order  $V$ , we start with the pseudo-random graph  $\mathcal{G}_V^{Indp}$  given by construction 4.1. To enforce a  $\approx \frac{V}{2 \log(V)}$  coloring, we take a "random looking" partition of the vertices into  $\approx \frac{V}{2 \log(V)}$  disjoint sets  $V_i$  each of size  $\approx 2 \log(V)$ , and we delete from  $\mathcal{G}_V^{Indp}$  the edges inside each set  $V_i$  (so each  $V_i$  requires only a single color). We thus obtain the final graph  $\mathcal{G}_V^{Color}$  which is clearly  $\approx \frac{V}{2 \log(V)}$  colorable, and clearly preserves the monotone decreasing properties of  $\mathcal{G}^{Indp}$  of almost surely having  $\omega$  at most  $\approx 2 \log(V)$ , and  $\alpha$  at least  $\approx 2 \log(V)$ . However, we shall have to argue that despite the omitting stage,  $\mathcal{G}_V^{Color}$  almost surely exhibits  $\omega$  at least  $\approx 2 \log(V)$ ,  $\alpha$  at most  $\approx 2 \log(V)$ , and  $\chi$  at least  $\approx \frac{V}{2 \log(V)}$ , as well.

The following figure demonstrates a construction of a specific graph where  $V = 7$ ,  $s(V) = 3$ . In figure (a) a graph  $g_V^{Indp}$  is generated. In figure (b) the forced independent sets  $V_1, V_2, V_3$  are determined. In figure (c) the internal edges of  $V_1, V_2, V_3$  are omitted to obtain  $g_V^{Color}$ .



Formally,

**Construction 4.2 - Pseudo Random Graphs Preserving the Chromatic Number of Random Graphs** - Let  $I$  denote the set of prime numbers. To construct a graph  $\mathcal{G}_V^{Color}$  of order  $V$ , according to the ensemble

$$\mathcal{G}^{Color} = \{\mathcal{G}_V^{Color}\}_{V \in I},$$

**The Initial Stage** - Construct a  $(4 \cdot \log^2(V))$ -wise independent pseudo-random graph  $\mathcal{G}_V^{Indp}$  according to construction 4.1.

**The Omitting Stage** - Uniformly pick a shift  $r \in \{1, 2, \dots, V-1\}$ , and let  $w_j = (j \cdot r) \bmod V$ . By the primality of  $V$ ,  $(w_0, w_1, w_2, \dots, w_{V-1})$  is a permutation of the vertices. Next, partition the vertices into  $\lceil \frac{V}{s(V)} \rceil$  equivalence classes  $V_1^r, V_2^r, \dots, V_{\lceil \frac{V}{s(V)} \rceil}^r$  each (except the last) of size  $s(V)$  as follows:

$$\underbrace{w_0, w_1, \dots, w_{s(V)-1}}_{V_1^r}, \underbrace{w_{s(V)}, w_{s(V)+1}, \dots, w_{2s(V)-1}}_{V_2^r}, \dots, \underbrace{w_{((j-1) \cdot s(V))}, \dots, w_{(j \cdot s(V)-1)}}_{V_j^r}, \dots$$

Formally,

$$V_j^r = \begin{cases} \{(i \cdot r) \bmod V \mid (j-1) \cdot s(V) \leq i < j \cdot s(V)\} & 1 \leq j \leq \lfloor \frac{V}{s(V)} \rfloor \\ \{0, 1, \dots, V-1\} \setminus (\bigcup_{\ell=1}^{\lfloor \frac{V}{s(V)} \rfloor} V_\ell^r) & j = \lceil \frac{V}{s(V)} \rceil. \end{cases}$$

Finally, delete from  $\mathcal{G}_V^{Indp}$  all edges connecting vertices  $w, w'$  inside the same equivalence class  $V_j^r$ , to obtain  $\mathcal{G}_V^{Color}$ .

**Theorem 4.2** Assuming that one-way functions exist, the graphs  $\mathcal{G}^{Color}$  of construction 4.2 are pseudo-random graphs preserving the graph properties,

- $P_{Clique} = (\omega, s(V), \frac{1}{s(V)})$ .
- $P_{Indp} = (\alpha, s(V), \frac{1}{s(V)})$ .

- $P_{Color} = (\chi, \frac{V}{s(V)}, \frac{1}{\sqrt{\log(V)}})$ .

**Proof** - Theorem 4.2 clearly follows from the 7 following facts, where all probabilities are taken over the entire construction 4.2,

- 1 - The graphs  $\mathcal{G}^{Color}$  are pseudo-random.
- 2 -  $\omega(\mathcal{G}_V^{Color}) \leq s(V) + 1$  w.p.  $1 - 2^{-\Omega(n(V))}$ .
- 3 -  $\alpha(\mathcal{G}_V^{Color}) \geq s(V) - 1$  w.p.  $1 - 2^{-\Omega(n(V))}$ .
- 4 - All graphs  $\mathcal{G}_V^{Color}$  are  $(\frac{V}{s(V)})$ -colorable.
- 5 -  $\omega(\mathcal{G}_V^{Color}) \geq s(V) - 1$  w.p.  $1 - 2^{-\Omega(n(V))}$ .
- 6 -  $\alpha(\mathcal{G}_V^{Color}) \leq s(V) + 1$  w.p.  $1 - 2^{-\Omega(n(V))}$ .
- 7 -  $\chi(\mathcal{G}_V^{Color}) \geq \frac{V}{s(V)+1}$  w.p.  $1 - 2^{-\Omega(n(V))}$ .

We prove these facts in the following order: facts 2-4, fact 1, fact 5, facts 6-7.

**Proving facts 2,3,4 -**

Fact 4 follows by coloring each equivalence class  $V_j^r$  with a different color. Next, recall that by claim 4.1, taking probabilities only on the construction of the initial graph  $\mathcal{G}_V^{Indp}$  implies that  $\omega(\mathcal{G}_V^{Indp}) > s(V) + 1$  w.p.  $2^{-\Omega(n(V))}$ , and  $\alpha(\mathcal{G}_V^{Indp}) < s(V) - 1$  w.p.  $2^{-\Omega(n(V))}$ . As we only delete edges from  $\mathcal{G}_V^{Indp}$  to obtain  $\mathcal{G}_V^{Color}$ , facts 2 and 3 follow. ■ (facts 2,3,4)

**Proving fact 1 -**

We first prove that  $\mathcal{G}^{Color}$  are efficiently computable. As  $\mathcal{G}^{Indp}$  are efficiently computable it suffices to prove that given the shift  $r$ , and given an arbitrary edge  $\{w_1, w_2\}$ , we can efficiently decide whether the equivalence classes of  $w_1, w_2$  are the same. Denote the inverse of the shift  $r$  in the field  $\mathbb{Z}_V$  by  $r^{-1}$ . Now, for an arbitrary vertex  $u$ ,

$$u \in V_j^r \iff \exists a \in \{0, 1, \dots, V-1\} \text{ s.t. } u = (a \cdot r) \bmod V, \text{ and } \lfloor \frac{a}{s(V)} \rfloor + 1 = j.$$

Hence, when  $w \in V_j$ , then the index  $j$  is given by

$\lfloor \frac{(w \cdot r^{-1}) \bmod V}{s(V)} \rfloor + 1$ , which is easy to compute and we are done.

To prove pseudo-randomness, recall Lemma 3.1 stating that mild modifications preserve pseudo-randomness. By this Lemma, it suffices to fix an arbitrary initial graph  $\mathcal{G}_V^{Indp}$ , and an arbitrary edge  $e$ , and to prove that the probability that  $e$  is deleted from  $\mathcal{G}_V^{Indp}$  during the omitting stage is negligibly small in  $n(V)$ . Here probabilities are taken only over the omitting stage, so the probability space is simply the uniform distribution of the shift  $r$  over  $\{1, 2, \dots, V - 1\}$ .

Recall that the index of the equivalence class of an arbitrary vertex  $w$ , is given by  $\lfloor \frac{(w \cdot r^{-1}) \bmod V}{s(V)} \rfloor + 1$ . Hence,

$$\begin{aligned} \text{The edge } e = \{w_1, w_2\} \text{ is deleted from } \mathcal{G}_V^{Indp} &\implies \\ w_1, w_2 \text{ are in the same equivalence class} &\iff \\ \lfloor \frac{(w_1 \cdot r^{-1} \bmod V)}{s(V)} \rfloor + 1 = \lfloor \frac{(w_2 \cdot r^{-1} \bmod V)}{s(V)} \rfloor + 1 &\implies \\ \lfloor \frac{|(w_1 \cdot r^{-1}) \bmod V - (w_2 \cdot r^{-1}) \bmod V|}{s(V)} \rfloor = 0 &\iff \end{aligned}$$

( assuming w.l.o.g. that  $(w_1 \cdot r^{-1}) \bmod V > (w_2 \cdot r^{-1}) \bmod V$  )

$$\begin{aligned} \lfloor \frac{((w_1 - w_2) \cdot r^{-1}) \bmod V}{s(V)} \rfloor = 0 &\iff \\ ((w_1 - w_2) \cdot r^{-1}) \bmod V \in \{1, 2, \dots, s(V) - 1\}. & \end{aligned}$$

Now, by the primality of  $V$ , as  $r$  is uniformly distributed over  $\{1, 2, \dots, V - 1\}$ , then  $(w_1 - w_2) \cdot r^{-1}$  is also uniformly distributed over  $\{1, 2, \dots, V - 1\}$ .

Therefore, taking probabilities only over the omitting stage gives,

$$\begin{aligned} \Pr[e \text{ is omitted during the omitting stage}] &\leq \\ \Pr[((w_1 - w_2) \cdot r^{-1}) \bmod V \in \{1, 2, \dots, s(V) - 1\}] &= \\ \frac{|\{1, 2, \dots, s(V) - 1\}|}{|\{1, 2, \dots, V - 1\}|} &= \frac{s(V) - 1}{V - 1}. \end{aligned}$$

As the least term is negligible in  $n(V)$  we are done. ■ (fact 1)

### Proving facts 5,6,7 -

Note that in facts 5-7 probabilities are taken over the entire construction 4.2.

This construction involves exactly 2 independent random components:

- The random shift  $r$  defining the omitting stage.
- The random seed  $s$  used to produce  $\mathcal{G}_V^{Indp}$ .

Consequently, to prove facts 5-7, it suffices to fix an arbitrary shift  $r$ , (or in other words to fix the forced independent sets  $V_1^r, V_2^r, \dots, V_{\lceil \frac{V}{s(V)} \rceil}^r$ ), and then prove the following facts 8-10 where this time **probabilities are taken only over the random seed  $s$  producing  $\mathcal{G}_V^{Indp}$** :

$$8 - \omega(\mathcal{G}_V^{Color}) \geq s(V) - 1 \text{ w.p. } 1 - 2^{-\Omega(n(V))}.$$

$$9 - \alpha(\mathcal{G}_V^{Color}) \leq s(V) + 1 \text{ w.p. } 1 - 2^{-\Omega(n(V))}.$$

$$10 - \chi(\mathcal{G}_V^{Color}) \geq \frac{V}{s(V)+1} \text{ w.p. } 1 - 2^{-\Omega(n(V))}.$$

As the following analysis is independent of the shift  $r$ , we abbreviate and denote each  $V_j^r$  by  $V_j$ .

**Proving fact 8 -**

Fixing the forced independent sets  $V_1, V_2, \dots, V_{\lceil \frac{V}{s(V)} \rceil}$ , we need to prove that  $(s(V) - 1)$ -cliques appear in  $\mathcal{G}_V^{Color}$  with high probability. To this end, we show that the probability of failing to obtain  $(s(V) - 1)$ -cliques in  $\mathcal{G}_V^{Color}$  is not much higher than the probability of failing to obtain  $(s(V) - 1)$ -cliques in  $\mathcal{G}_V^{Indp}$  (which is known to be  $V^{-\Omega(1)}$  by claim 4.1).

Indeed, let  $\mathbb{T}$  denote the collection of all subsets of vertices of cardinality  $s(V) - 1$ , and let  $\mathbb{T}^{Color} \subset \mathbb{T}$  denote the collection of those subsets containing at most one vertex from each  $V_j$  (i.e.  $\mathbb{T}^{Color}$  is the collection of all 'potential cliques' that might appear in  $\mathcal{G}_V^{Color}$ ). For each  $T \in \mathbb{T}$ , let  $X_T$  denote the random variable indicating whether  $T$  induces a clique in  $\mathcal{G}_V^{Indp}$ . Note that

for any  $T \in \mathbb{T}^{Color}$ ,  $T$  induces a clique in  $\mathcal{G}_V^{Indp}$  if and only if it induces a clique in  $\mathcal{G}_V^{Color}$ . Finally, let  $X^{Color} = \sum_{T \in \mathbb{T}^{Color}} X_T$  and let  $X = \sum_{T \in \mathbb{T}} X_T$ . These random variables count the number of  $(s(V) - 1)$ -cliques in  $\mathcal{G}_V^{Color}$  and in  $\mathcal{G}_V^{Indp}$  respectively.

Our goal is to show that  $\Pr[X^{Color} = 0] = \Theta(\Pr[X = 0]) = V^{-\Omega(1)}$ , namely,  $\Pr[X^{Color} > 0] = 1 - V^{-\Omega(1)}$ . We stress that the **distributions of all random variables  $X_T, X^{Color}, X$  are taken only over the seed generating  $\mathcal{G}_V^{Indp}$ .**

We shall use the following claim about  $\mathcal{G}_V^{Indp}$  proved in the original analysis of the clique-number of random graphs in [AS].

**Claim 4.2**  $\frac{\text{var}(X)}{\mathbb{E}^2(X)} = V^{-\Omega(1)}$ .

We shall also use the following claims which we prove later on.

**Claim 4.3**  $\mathbb{E}(X^{Color}) = \Theta(\mathbb{E}(X))$ .

**Claim 4.4**  $\text{var}(X^{Color}) \leq \text{var}(X)$ .

Assuming these claims we get,

$$\begin{aligned} \Pr[\text{There is no } (s(V) - 1)\text{-clique in } \mathcal{G}_V^{Color}] &= \\ \Pr[|X^{Color} - \mathbb{E}(X^{Color})| \geq \mathbb{E}(X^{Color})] &\leq \quad (\text{Chebyshev's inequality}) \\ \frac{\text{var}(X^{Color})}{(\mathbb{E}(X^{Color}))^2} &\leq \quad (\text{By claims 4.3 and 4.4}) \\ \Theta\left(\frac{\text{var}(X)}{(\mathbb{E}(X))^2}\right) &= V^{-\Omega(1)} \quad (\text{By claim 4.2}), \end{aligned}$$

as required.

We next prove claims 4.3 and 4.4.

**Proving claim 4.3 -**

We first lower-bound the cardinality of  $\mathbb{T}^{Color}$  as follows. We restrict ourselves to counting only those  $T \in \mathbb{T}^{Color}$  s.t.  $|T \cap V_{\lceil \frac{V}{s(V)} \rceil}| = 0$ . There are exactly  $\binom{\lfloor \frac{V}{s(V)} \rfloor}{s(V)-1}$  choices for the independent sets  $V_j$ , and then, there are exactly  $(s(V))^{s(V)-1}$  ways of picking a single vertex from each  $V_j$ . Thus  $|\mathbb{T}^{Color}| \geq \binom{\lfloor \frac{V}{s(V)} \rfloor}{s(V)-1} \times (s(V))^{s(V)-1}$ . Next, the  $(4 \log^2 V)$ -wise independence of  $\mathcal{G}_V^{Indp}$  implies that each  $(s(V) - 1)$ -vertex-set induces a clique in  $\mathcal{G}_V^{Indp}$  w.p.  $(\frac{1}{2})^{\binom{s(V)-1}{2}}$ . Consequently,  $\mathbb{E}(X^{Color}) \geq \binom{\lfloor \frac{V}{s(V)} \rfloor}{s(V)-1} \times (s(V))^{s(V)-1} \times (\frac{1}{2})^{\binom{s(V)-1}{2}}$ . Similarly,  $\mathbb{E}(X) = \binom{V}{s(V)-1} \times (\frac{1}{2})^{\binom{s(V)-1}{2}}$ .

Clearly, for any graph  $g$ ,  $X^{Color}(g) \leq X(g)$ , so  $\mathbb{E}(X^{Color}) \leq \mathbb{E}(X)$ . Therefore to prove claim 4.3 it suffices to show that  $\mathbb{E}(X^{Color}) = \Omega(\mathbb{E}(X))$ . Indeed

letting  $s = s(V)$ :

$$\begin{aligned} \frac{\mathbb{E}(X^{Color})}{\mathbb{E}(X)} &\geq \frac{\binom{\lfloor \frac{V}{s} \rfloor}{s-1} \times (s)^{s-1} \times (\frac{1}{2})^{\binom{s-1}{2}}}{\binom{V}{s-1} \times (\frac{1}{2})^{\binom{s-1}{2}}} = \\ &(s)^{s-1} \times \frac{\prod_{j=0}^{(s-2)} (\lfloor \frac{V}{s} \rfloor - j)}{\prod_{j=0}^{(s-2)} V - j} \geq \\ &(s)^{s-1} \times \prod_{j=0}^{j=(s-2)} \frac{\frac{V}{s} (1 - \frac{s(j+1)}{V})}{V(1 - \frac{j}{V})} = \\ &\prod_{j=0}^{j=(s-2)} \frac{1 - \frac{s(j+1)}{V}}{1 - \frac{j}{V}} \geq \\ &\prod_{j=0}^{j=(s-2)} 1 - \frac{s \cdot s}{V} = (1 - \frac{s^2}{V})^s \geq 1 - \frac{s^3}{V} = \Omega(1). \blacksquare \text{ (claim 4.3)} \end{aligned}$$

#### Proving claim 4.4 -

For a pair of vertex-sets  $T, T' \in \mathbb{T}$ , the variables  $X_T, X_{T'}$  are dependent iff  $|T \cap T'| \geq 2$ . We denote this dependence by  $T \sim T'$ . Clearly, for  $T, T' \in \mathbb{T}$ , the covariance  $\text{cov}(X_T, X_{T'})$  is non-negative. Consequently, since  $\mathbb{T}^{Color} \subset \mathbb{T}$ :

$$\sum_{T \in \mathbb{T}^{Color}} \text{var}(X_T) + 2 \sum_{T \sim T', T, T' \in \mathbb{T}^{Color}} \text{cov}(X_T, X_{T'}) \leq$$

$\sum_{T \in \mathbb{T}} \text{var}(X_T) + 2 \sum_{T \sim T', T, T' \in \mathbb{T}} \text{cov}(X_T, X_{T'}) = \text{var}(X)$ . ■ (claim 4.4)

By proving claims 4.3 and 4.4 the proof fact 8 is completed. ■ (fact 8)

### Reducing fact 10 to fact 9

We note that for any graph  $g$  on  $V$  vertices  $\chi(g) \geq \frac{V}{\alpha(g)}$ . Indeed, consider some optimal coloring, and let  $C_1, \dots, C_{\chi(g)}$  be a partitioning of the vertices, s.t. each  $C_j$  is an independent-set having a distinct color. Then,  $V = \sum_{i=1}^{\chi(g)} |C_i| \geq \chi(g) \times \alpha(g)$ . Consequently, fact 10 is immediately implied by fact 9.

### Proving fact 9

Note that fact 9 states that  $(s(V) + 2)$ -independent-sets rarely appear in  $\mathcal{G}_V^{Color}$ . We remark that at a first glance this seems surprising: After all, the omitting stage forces  $\lfloor \frac{V}{s(V)} \rfloor$  independent-sets of size  $s(V)$ , and adding only 2 more vertices to any of these **disjoint** sets provides a  $(s(V) + 2)$  independent-set.

### Notation and Structure of Proof for fact 9

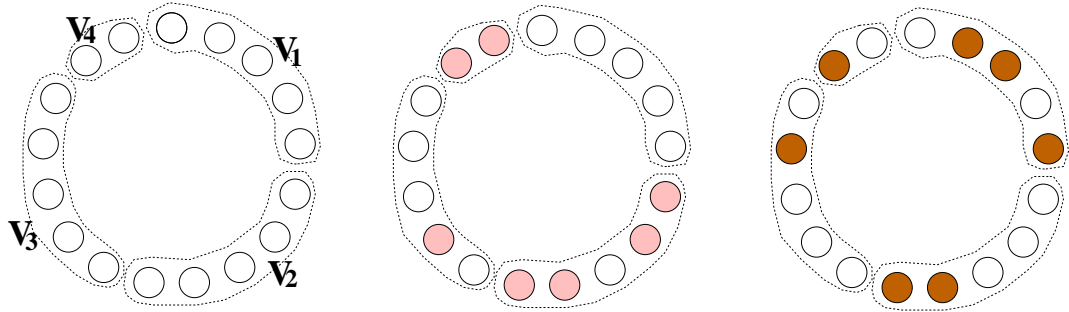
Recall that the forced independent sets  $V_1, V_2, \dots, V_{\lfloor \frac{V}{s(V)} \rfloor}$  are fixed. Let  $\mathbb{S}$  denote the collection of all vertex-sets of cardinality  $s(V) + 2$ . For any  $S \in \mathbb{S}$ , let  $X^S$  denote the random variable indicating whether  $S$  induces an independent-set in  $\mathcal{G}_V^{Color}$ . Let  $p^S$  denote the probability that  $X^S = 1$ , and let  $X = \sum_{S \in \mathbb{S}} X^S$  count the number of  $(s(V) + 2)$ -independent-sets appearing in  $\mathcal{G}_V^{Color}$ . We stress that the distributions of the random variables  $X, X^S$  are taken only over the seed generating  $\mathcal{G}_V^{Indp}$ .

Next, for any  $S \in \mathbb{S}$ , let  $\ell^S$  count the number of non-empty intersections  $S \cap V_{i_1}, \dots, S \cap V_{i_{\ell^S}}$ , where  $i_1 < \dots < i_{\ell^S}$ . Also let  $\vec{r}^S = (r_1^S, \dots, r_{\ell^S}^S)$ , indicate

$V_j, \mathbb{S}$   
 $S, X^S$   
 $X, p^S$   
 $\ell^S$   
 $\vec{r}^S, r_j^S$

the cardinalities of these intersections, namely  $r_j^S = |S \cap V_{i_j}|$ , and let  $u^S = \max\{r_j^S\}_{j=1}^{\ell^S}$ , denote the maximal intersection size.

For instance, consider the case  $V = 17$ ,  $s(V) = 5$ ,  $s(V) + 2 = 7$ . Figure (a) shows the partition of the 17 vertices into 4 vertex-sets  $V_1, V_2, V_3, V_4$ . Figure (b) shows a vertex-set  $S \in \mathbb{S}$  having  $\vec{r}^S = (0, 4, 1, 2)$ ,  $u^S = 4$ ,  $\ell^S = 3$ . Figure (c) shows another vertex-set  $T \in \mathbb{S}$  having  $\vec{r}^T = (3, 2, 1, 1)$ ,  $u^T = 3$ ,  $\ell^T = 4$ :



(a) The forced independent sets      (b) Vertex-set  $S$  ●      (c) Vertex-set  $T$  ●

Finally, let  $M_u$  denote the number of vertex-sets  $S \in \mathbb{S}$  having  $u^S = u$ ,  $M_u, \bar{M}_\ell$  and let  $\bar{M}_\ell$  denote the number of vertex-sets  $S \in \mathbb{S}$  having  $\ell^S = \ell$ .

Having this notation in hand our goal is to upper bound  $\Pr[X \geq 1]$ , where all probabilities discussed henceforth are taken only over the seed which generates  $\mathcal{G}_V^{Indp}$ .

To this end we introduce 5 subsets  $\mathbb{S}_1, \dots, \mathbb{S}_5 \subset \mathbb{S}$ , s.t.  $\mathbb{S} \subseteq \bigcup_{i=1}^5 \mathbb{S}_i$ , and show an upper-bound of  $V^{-\Omega(1)}$  on  $\sum_{S \in \mathbb{S}_i} \Pr[X^S = 1]$ , for each  $i = 1, \dots, 5$ .

We fix an arbitrary  $\frac{1}{6} \leq c \leq \frac{1}{5}$ , and define the sub-sets  $\mathbb{S}_i$  as follows:  $c$

- Large  $u^S$ :  
 $\mathbb{S}_1 = \{S \in \mathbb{S} | (\frac{3}{2} + c) \log V \leq u^S \leq s(V)\}.$
- Medium  $u^S$ :  
 $\mathbb{S}_2 = \{S \in \mathbb{S} | (1 + c) \log V \leq u^S \leq (2 - c) \log V\}.$
- Small  $u^S$ :  
 $\mathbb{S}' = \{S \in \mathbb{S} | 1 \leq u^S \leq (1 + c) \log V\}.$ 
  - Small  $u^S$  and large  $\ell^S$ :  
 $\mathbb{S}_3 = \{S \in \mathbb{S}' | (1 + c) \log V \leq \ell^S \leq s(V) + 2\}.$
  - Small  $u^S$  and medium  $\ell^S$ :  
 $\mathbb{S}_4 = \{S \in \mathbb{S}' | c \log V \leq \ell^S \leq (2 - c) \log V\}.$
  - Small  $u^S$  and small  $\ell^S$ :  
 $\mathbb{S}_5 = \{S \in \mathbb{S}' | 1 \leq \ell^S \leq (1 - 2c) \log V\}.$

Before handling each  $\mathbb{S}_i$  separately, we recall that  $s(V) = 2 \log V \pm o(\log V)$ , and provide some general useful upper-bounds which we prove only on page 54.

**Claim 9.1** For all  $u$ ,  $u = 1, \dots, s(V)$ ,

1.  $M_u \leq 2 \lceil \frac{V}{s(V)} \rceil \times \binom{s(V)}{u} \times \binom{V-s(V)}{s(V)+2-u}.$
2.  $M_u \leq V^{[2 \log V - u + o(\log V)]}.$

**Claim 9.2** For all  $\ell$ ,  $\ell = 1, \dots, s(V) + 2$ ,

1.  $\bar{M}_\ell \leq \binom{\lceil \frac{V}{s(V)} \rceil}{\ell} \times \binom{s(V)+1}{\ell-1} \times s(V)^{s(V)+2}.$

$$2. \bar{M}_\ell \leq V^{\ell+o(\log V)}.$$

**Claim 9.3** For any vertex-set  $S \in \mathbb{S}$ ,

$$1. p^S \leq 2^{-\frac{1}{2}[\sum_{i=1}^{\ell^S} r_i^S - (\sum_{j=1, j \neq i}^{\ell^S} r_j^S)]}.$$

$$2. p^S \leq V^{-[s(V)+2-u^S] \times (1 \pm o(1))}.$$

$$3. p^S \leq 2^{-u^S [s(V)+2-u^S]}.$$

**Claim 9.4** For any vertex-set  $S \in \mathbb{S}$ ,

$$1. \text{ If } \ell^S > 2, \text{ then } p^S \leq 2^{-\binom{s(V)+2}{2} + \binom{s(V)-\ell^S+3}{2}}.$$

$$2. \text{ If } \ell^S > c \log V, \text{ then } p^S \leq V^{-\frac{1}{2} \ell^S (4 - \frac{\ell^S}{\log V}) + o(\log V)}.$$

It will be convenient to handle the  $\mathbb{S}_i$ -s in the order  $\mathbb{S}_2, \mathbb{S}_1, \mathbb{S}_5, \mathbb{S}_4, \mathbb{S}_3$ .

#### 4.5.1 Handling vertex-sets $S$ having medium $u^S$

**Lemma 9.1**  $\sum_{S \in \mathbb{S}_2} \Pr[X^S = 1] \leq V^{-\Omega(\log V)}$ .

**Proof** - First, let  $\bar{d}(u) = \frac{u}{\log V}$ , and note that by part 3 in claim 9.3, for any vertex-set  $S \in \mathbb{S}$  having  $u^S = u$  the probability that  $S$  induces an independent-set is upper-bounded by

$$\begin{aligned} \bar{p}^u &= 2^{-u^S \times [s(V)+2-u^S]} \leq \\ &2^{-\bar{d}(u) \log V [2 \log V \pm o(\log V) - \bar{d}(u) \log V]} \leq \\ &V^{-\bar{d}(u) [(2-\bar{d}(u)) \log V] \pm o(\log V)}. \end{aligned}$$

Next, recall that  $M_u$ , the number of vertex-sets  $S \in \mathbb{S}$  having  $u^S = u$ , is upper-bounded (part 2 in claim 9.1) as follows:

$$M_u \leq V^{2 \log V - u + o(\log V)} = V^{[2 - \bar{d}(u)] \log V + o(\log V)}.$$

Finally, note (by, say, taking derivatives) that for all  $u' \in R_2 = \{u | (1 + c) \log V \leq u \leq (2 - c) \log V\}$ , we have  $[2 - \bar{d}(u')][1 - \bar{d}(u')] \leq -c(1 - c)$ .

Consequently,

$$\begin{aligned} \sum_{S \in \mathbb{S}_1} \Pr[X^S = 1] &\leq \\ \sum_{u \in R_2} M_u \times \bar{p}_u &\leq \\ \sum_{u \in R_2} V^{[2 - \bar{d}(u)] \log V + o(\log V)} \times V^{-\bar{d}(u)[(2 - \bar{d}(u)) \log V] \pm o(\log V)} &\leq \\ \sum_{u \in R_2} V^{\log V [2 - \bar{d}(u)][1 - \bar{d}(u)] \pm o(\log V)} &\leq \\ \sum_{u \in R_2} V^{-c(1 - c) \log V \pm o(\log V)} &\leq \\ |R_2| \times V^{-\Omega(\log V)} &= V^{-\Omega(\log V)}. \end{aligned}$$

■ (Handling vertex-sets  $S$  having medium  $u^S$ )

#### 4.5.2 Handling vertex-sets $S$ having large $u^S$

**Lemma 9.2**  $\sum_{S \in \mathbb{S}_1} \Pr[X^S = 1] \leq V^{-\Omega(1)}$ .

**Proof -** By part 1 in claim 9.1, the number of sets  $S \in \mathbb{S}$  having  $u^S = u$  is upper-bounded by  $2^{\lceil \frac{V}{s(V)} \rceil} \binom{s(V)}{u} \binom{V - s(V)}{s(V) + 2 - u}$ , and by part 3 in claim 9.3, each set  $S$  induces an independent-set w.p. at most  $2^{-u^S [s(V) + 2 - u^S]}$ . Therefore we take

$$\mathbb{U}_u = 2^{\lceil \frac{V}{s(V)} \rceil} \binom{s(V)}{u} \binom{V - s(V)}{s(V) + 2 - u} \times 2^{-u \times (s(V) + 2 - u)},$$

as an upper-bound for  $\sum_{S \in \mathbb{S}_1, u^S = u} \Pr[X^S = 1]$ .

Next, we set  $\tilde{d}(u)$  s.t.  $(\frac{3}{2} + \tilde{d}(u)) \log V = u$ . Note that for  $S \in \mathbb{S}_1$ , we have  $u^S \in R_1 = \{u | (\frac{3}{2} + c) \log V \leq u \leq s(V)\}$ .

We are interested in finding the maximal  $\mathbb{U}_u$  where  $u$  ranges over  $R_1$ . To this end, we examine,

$$\begin{aligned}
\frac{\mathbb{U}_{u+1}}{\mathbb{U}_u} &= \\
&\left(\frac{s(V)-u}{u+1} \times \frac{s(V)+2-u}{V-2s(V)+u-1}\right) \times 2^{2u} \times 2^{-(s(V)+1)} \geq \\
&\left(\frac{1}{u+1} \times \frac{1}{V}\right) \times 2^{2 \log V(\frac{3}{2}+\tilde{d}(u))} \times 2^{-(2\pm o(1)) \log V} = \\
&(V^{-1\pm o(1)}) \times V^{2(\frac{3}{2}+\tilde{d}(u))} \times V^{-2\pm o(1)} = \\
&V^{2\tilde{d}(u)\pm o(1)} \geq \quad \quad \quad [\text{since for } u \in R_1, \tilde{d}(u) \geq c] \\
&V^{2c\pm o(1)}.
\end{aligned}$$

Thus, for sufficiently large  $V$ , the maximal  $\mathbb{U}_u$  is  $\mathbb{U}_{s(V)}$ , and then,

$$\begin{aligned}
\sum_{S \in \mathbb{S}_1} \Pr[X^S = 1] &\leq \\
\sum_{u \in R_1} \mathbb{U}_u &\leq |R_1| \times \mathbb{U}_{s(V)} \leq \\
s(V) \times 2^{\lceil \frac{V}{s(V)} \rceil} \times \binom{s(V)}{s(V)} \times \binom{V-s(V)}{2} \times 2^{-2s(V)} &\leq \\
V^{o(1)} \times V \times V^{o(1)} \times V^2 \times V^{-4\pm o(1)} &\leq V^{-1\pm o(1)},
\end{aligned}$$

Which completes the proof. ■ (Handling vertex-sets  $S$  having large  $u^S$ )

#### 4.5.3 Handling vertex-sets $S$ having a small $u^S$ and small $\ell^S$

**Lemma 9.3**  $\sum_{S \in \mathbb{S}_5} \Pr[X^S = 1] \leq V^{-\Omega(\log V)}$ .

**Proof** - First, recall we have set  $\bar{d}(u) = \frac{u}{\log V}$ . Note that for  $S \in \mathbb{S}_5$ , we have  $2 - \bar{d}(u^S) \geq 1 - c$ . Also note that for a vertex-set  $S$  having  $u^S = u$ , the probability that  $S$  induces an independent-set is upper-bounded (part 2 in claim 9.3) by

$$\bar{p}^u \leq V^{-[s(V)+2-u](1\pm o(1))} =$$

$$\begin{aligned}
V^{-(2-\bar{d}(u)) \log V(1\pm o(1))} &\leq \\
V^{-(1-c) \log V(1\pm o(1))}. &
\end{aligned}$$

Next, recall that  $\bar{M}_\ell$  denotes the number of vertex-sets  $S \in \mathbb{S}$  having  $\ell^S = \ell$ , and that  $\bar{M}_\ell$  is upper-bounded by  $V^{\ell+o(\log V)}$  (part 2 in claim 9.2). Finally, note that for  $S \in \mathbb{S}_5$  we have  $\ell^S \in R_5 = \{\ell \mid 1 \leq \ell \leq (1-2c) \log V\}$ . Therefore,

$$\begin{aligned}
\sum_{S \in \mathbb{S}_5} \Pr[X^S = 1] &\leq \\
\sum_{\ell \in R_5} \bar{M}_\ell \times \bar{p}^u &\leq \\
\sum_{\ell \in R_5} V^{\ell+o(\log V)} \times V^{-(1-c) \log V(1\pm o(1))} &\leq \\
\sum_{\ell \in R_5} V^{(1-2c) \log V + o(\log V)} \times V^{-(1-c) \log V(1\pm o(1))} &\leq \\
\sum_{\ell \in R_5} V^{-c \log V(1\pm o(1))} &= \\
|R_5| \times V^{-\Omega(\log V)} &= V^{-\Omega(\log V)}.
\end{aligned}$$

■ (Handling vertex-sets  $S$  having small  $u^S$  and small  $\ell^S$ )

#### 4.5.4 Handling vertex-sets $S$ having a small $u^S$ and medium $\ell^S$

**Lemma 9.4**  $\sum_{S \in \mathbb{S}_4} \Pr[X^S = 1] \leq V^{-\Omega(\log V)}$ .

**Proof-** Recall that by part 2 in Lemma 9.2  $\bar{M}_\ell \leq V^{\ell+o(\log V)}$ , and by part 2 in Lemma 9.4, whenever  $\ell = \ell^S > c \log V$ , then  $p^S \leq V^{-\frac{1}{2}\ell(4-\frac{\ell}{\log V})+o(\log V)}$ .

Now,

$$\begin{aligned}
V^{\ell+o(\log V)} \times V^{-\frac{1}{2}\ell(4-\frac{\ell}{\log V})+o(\log V)} &= \\
V^{\ell(-1+\frac{\ell}{2\log V})+o(\log V)} &= \quad \left[ \text{since } \ell \leq (2-c) \log V \right] \\
V^{\ell(-\frac{c}{2})+o(\log V)} &=
\end{aligned}$$

$$V^{-\Omega(\log V)}.$$

Therefore,

$$\begin{aligned} \sum_{S \in \mathbb{S}_4} \Pr[X^S = 1] &\leq \\ \sum_{\ell=c \log V}^{(2-c) \log V} V^{-\Omega(\log V)} &= V^{-\Omega(\log V)}. \end{aligned}$$

■ (Handling vertex-sets  $S$  having small  $u^S$  and medium  $\ell^S$ )

#### 4.5.5 Handling vertex-sets $S$ having a small $u^S$ and large $\ell^S$

**Lemma 9.5**  $\sum_{S \in \mathbb{S}_3} \Pr[X^S = 1] \leq V^{-\Omega(1)}$ .

**Proof** - By part 1 in Lemma 9.2  $\bar{M}_\ell \leq \binom{\lceil \frac{V}{s(V)} \rceil}{\ell} \binom{s(V)+1}{\ell-1} s(V)^{s(V)+2}$ , and by part 1 in Lemma 9.4, whenever  $\ell = \ell^S > 2$ , then  $p^S \leq 2^{-(\binom{s(V)+2}{2}) + (\binom{s(V)+3-\ell}{2})}$ .

Therefore we take  $\mathbb{L}_\ell = \binom{\lceil \frac{V}{s(V)} \rceil}{\ell} \binom{s(V)+1}{\ell-1} s(V)^{s(V)+2} \times 2^{-(\binom{s(V)+2}{2}) + (\binom{s(V)+3-\ell}{2})}$  as an upper bound for  $\sum_{S \in \mathbb{S}, \ell^S = \ell} \Pr[X^S = 1]$ .

We next find the maximal  $\mathbb{L}_\ell$  where  $\ell$  ranges over  $(1+c) \log V \leq \ell \leq s(V) + 2$ . We examine

$$\begin{aligned} \frac{\mathbb{L}_{\ell+1}}{\mathbb{L}_\ell} &= \\ \frac{\binom{\lceil \frac{V}{s(V)} \rceil - \ell}{\ell+1} \times \frac{s(V) - \ell + 2}{\ell} \times 2^{-(s(V) - \ell + 2)}}{\binom{\lceil \frac{V}{s(V)} \rceil}{\ell} \binom{s(V)+1}{\ell-1} s(V)^{s(V)+2} \times 2^{-(\binom{s(V)+2}{2}) + (\binom{s(V)+3-\ell}{2})}} &\geq \\ \frac{V^{1-o(1)}}{V^{o(1)}} \times \frac{1}{V^{o(1)}} \times 2^{(-2 \log V + (1+c) \log V)(1 \pm o(1))} &= \\ V^{1 \pm o(1)} \times V^{-2 + (1+c) \pm o(1)} &= V^{c \pm o(1)}, \end{aligned}$$

so for sufficiently large  $V$ , the maximal  $\mathbb{L}_\ell$  is  $\mathbb{L}_{s(V)+2}$ .

We consequently seek to upper-bound  $\mathbb{L}_{s(V)+2}$ . To this end the following claims are required.

**Claim 9.5**  $\binom{V}{s(V)+2} 2^{-(\binom{s(V)+2}{2})} \leq V^{-1 \pm o(1)}$ .

**Claim 9.6**  $\frac{\binom{\lceil \frac{V}{s(V)+2} \rceil}{s(V)+2}}{\binom{V}{s(V)+2}} = s(V)^{-(s(V)+2)}(1 \pm o(1))$ .

Using claims 9.5 and 9.6 in the following inequality gives

$$\begin{aligned} \mathbb{L}_{s(V)+2} &= \\ &\binom{\lceil \frac{V}{s(V)+2} \rceil}{s(V)+2} s(V)^{s(V)+2} 2^{-\binom{s(V)+2}{2}} = \\ &\left[ \frac{\binom{\lceil \frac{V}{s(V)+2} \rceil}{s(V)+2}}{\binom{V}{s(V)+2}} s(V)^{s(V)+2} \right] \times \left[ \binom{V}{s(V)+2} 2^{-\binom{s(V)+2}{2}} \right] \leq \\ &\left[ \frac{s(V)^{s(V)+2}}{s(V)^{s(V)+2} (1 \pm o(1))} \right] \times [V^{-1 \pm o(1)}] = V^{-1 \pm o(1)}. \end{aligned}$$

Therefore, for sufficiently large  $V$ ,

$$\begin{aligned} \sum_{S \in \mathbb{S}_3} \Pr[X^S = 1] &= \\ \sum_{\ell=(1+c) \log V}^{s(V)+2} \mathbb{L}_\ell &\leq \\ (s(V) + 2) \mathbb{L}_{s(V)+2} &\leq V^{-\Omega(1)}. \end{aligned}$$

To complete the proof Lemma 9.5 we prove claims 9.5 and 9.6.

**Proof for claim 9.5** - Recall that  $E_{s,V} = \binom{V}{s} 2^{-\binom{s}{2}}$  denotes the expected number of  $s$ -cliques in a random graph of size  $V$ , and that  $s(V)$  was defined as the maximal  $s$  for which  $E_{s,V} > 1$ . Therefore  $E_{s(V)+1,V} \leq 1$ .

Next,

$$\begin{aligned} \frac{E_{s(V)+2,V}}{E_{s(V)+1,V}} &= \\ \frac{V-s(V)-1}{(s(V)+2)2^{s(V)+1}} &\leq \\ \frac{V}{2^{(2-o(1)) \log V}} &= V^{-1 \pm o(1)}, \end{aligned}$$

which completes the proof. ■ (claim 9.5).

**Proof for claim 9.6** - Use the estimation

$$\binom{N}{K} = \frac{1}{\sqrt{2\pi K}} \left(\frac{e \cdot N}{K}\right)^K \times \left[\left(1 + \Theta\left(\frac{1}{K}\right)\right) e^{\Theta\left(\frac{1}{N} K^2\right)}\right]^{-1}.$$

We note that for  $K = \Theta(\log V)$ , and for  $N = V^{\Omega(1)}$  we get

$$\binom{N}{K} = \frac{1}{\sqrt{2\pi K}} \left(\frac{e \cdot N}{K}\right)^K (1 \pm o(1)).$$

Therefore,

$$\begin{aligned} \frac{\binom{\lceil \frac{V}{s(V)} \rceil}{s(V)+2}}{\binom{V}{s(V)+2}} &= \\ \frac{\lceil \frac{V}{s(V)} \rceil^{(s(V)+2) \times (1 \pm o(1))}}{V^{(s(V)+2) \times (1 \pm o(1))}} &= [s(V)]^{-(s(V)+2)} (1 \pm o(1)). \end{aligned}$$

- (claim 9.6).
- (Handling vertex-sets  $S$  having small  $u^S$  and large  $\ell^S$ )

We have finished proving an upper-bound of  $V^{-\Omega(1)}$  on  $\sum_{S \in \mathbb{S}_i} \Pr[X^S = 1]$ , for each  $i = 1, \dots, 5$ . Thus, to complete the proof fact 9 it suffices to prove claims 9.1, 9.2, 9.4 and 9.3 stated in page 47.

**Proof for claim 9.1 -**

Starting with the first upper-bound, we first count only the sets in  $S_u^1 = \{S \in \mathbb{S} | u^S = u > |S \cap V_{\lceil \frac{V}{s(V)} \rceil}|\}$ . For  $S \in S_u^1$  let  $V_i$  be the first independent set achieving  $|S \cap V_i| = u^S$ , and note that  $|V_i| = s(V)$ . There are no more than  $\lceil \frac{V}{s(V)} \rceil$  possible choices for  $V_i$ . Once  $V_i$  is fixed, there are precisely  $\binom{s(V)}{u}$  ways to choose  $(S \cap V_i)$ , and no more than  $\binom{V-s(V)}{s(V)+2-u}$  possible choices for  $(S \setminus V_i)$ . Combining this gives,  $|S_u^1| \leq \lceil \frac{V}{s(V)} \rceil \binom{s(V)}{u} \binom{V-s(V)}{s(V)+2-u}$ .

We next handle the remaining sets  $S_u^2 = \{S \in \mathbb{S} | u^S = u = |S \cap V_{\lceil \frac{V}{s(V)} \rceil}|\}$ . We prove that  $|S_u^2| \leq |S_u^1|$ , by showing a 1:1 mapping  $\Psi : S_u^2 \rightarrow S_u^1$ . We define  $\Psi(S)$  by specifying  $\Psi(S) \cap V_j$  for each  $j = 1, \dots, \lceil \frac{V}{s(V)} \rceil$ . First, for  $j = 2, \dots, \lceil \frac{V}{s(V)} \rceil - 1$ , let  $\Psi(S) \cap V_j \stackrel{\text{def}}{=} S \cap V_j$ . Next,  $\Psi(S)$  picks from  $V_1$  the

same number of elements and in the same lexicographic order as  $S$  picks from  $V_{\lceil \frac{V}{s(V)} \rceil}$ . Similarly  $\Psi(S)$  picks from  $V_{\lceil \frac{V}{s(V)} \rceil}$  the same number of elements and in the same lexicographic order as  $S$  does from  $V_1$ .

It is easy to verify that  $\Psi(S)$  is 1:1. Consequently,  $M_u = |S_u^1| + |S_u^2| \leq 2|S_u^1| = 2\lceil \frac{V}{s(V)} \rceil \binom{|V_j|}{u} \binom{V-|V_j|}{s(V)+2-u}$ , which proves the first upper-bound.

To derive the second upper-bound from the first one, note that,

1.  $\lceil \frac{V}{s(V)} \rceil \leq V$ .
2.  $\binom{s(V)}{u} \leq 2^{s(V)} \leq 2^{2 \log V + o(\log V)} = V^{2+o(1)}$ .
3.  $\binom{V-s(V)}{s(V)+2-u} \leq V^{s(V)+2-u} = V^{2 \log V - o(\log V) - u}$ .

■ (claim 9.1)

**Proof for claim 9.2 -**

We count the number of vertex-sets  $S$  of size  $s(V) + 2$  having  $\ell^S = \ell$ . There are precisely  $\binom{\lceil \frac{V}{s(V)} \rceil}{\ell}$  possible choices for the non-empty intersecting sets  $V_{i_1}, \dots, V_{i_\ell}$ . Once the sets  $V_{i_j}$  are fixed, we choose the cardinalities  $\vec{r}^S = (r_1^S, \dots, r_\ell^S)$  of the intersections. There are only  $\binom{s(V)+1}{\ell-1}$  possible choices of  $\vec{r}^S$  having  $r_i^S \geq 1$  and  $r_1^S + \dots + r_\ell^S = s(V) + 2$ . Finally, given the sets  $V_{i_j}$ , and given the cardinalities  $\vec{r}^S$ , the number of possible choices for  $S \cap V_{i_1}, \dots, S \cap V_{i_\ell}$  is:

$$\prod_{j=1}^{\ell} \binom{|V_{i_j}|}{r_j^S} \leq \prod_{j=1}^{\ell} |V_{i_j}|^{r_j^S} \leq \prod_{j=1}^{\ell} s(V)^{r_j^S} \leq s(V)^{s(V)+2}.$$

This proves the first upper-bound.

To derive the second upper-bound from the first one, note that,

1.  $\binom{\lceil \frac{V}{s(V)} \rceil}{\ell} \leq V^\ell$ .
2.  $\binom{s(V)+1}{\ell-1} \leq 2^{s(V)+1} \leq 2^{2 \log V(1 \pm o(1))} = V^{2 \pm o(1)}$ .
3.  $s(V)^{s(V)+2} = \Theta(2 \log V)^{\Theta(2 \log V)} = V^{\Theta(\log \log V)}$ .

■ (claim 9.2)

**Proof for claim 9.3 -**

Consider the edges connecting vertices inside some  $S \in \mathbb{S}$ . Precisely  $\sum_{j=1}^{\ell^S} \binom{r_j^S}{2}$  edges were deleted during the omitting stage, and the rest of the  $\binom{s(V)+2}{2}$  edges inside  $S$  are determined by  $\mathcal{G}_V^{Indp}$ . As  $\mathcal{G}_V^{Indp}$  is  $(4 \times (\log V)^2)$ -wise independent, the rest of the edges are independently picked each w.p.  $\frac{1}{2}$ .

Therefore,  $p^S = 2^{\Phi_S}$ , where

$$\Phi_S = -\binom{s(V)+2}{2} + \sum_{j=1}^{\ell^S} \binom{r_j^S}{2}.$$

An elementary calculation gives,

$$\begin{aligned} \Phi_S &= -\binom{\sum_{j=1}^{\ell^S} r_j^S}{2} + \sum_{j=1}^{\ell^S} \binom{r_j^S}{2} = \\ &= -\frac{1}{2} \times \sum_{i=1}^{\ell^S} r_i^S \left( \sum_{j=1, j \neq i}^{\ell^S} r_j^S \right), \end{aligned}$$

which proves the first upper-bound.

Next, by maximality of  $u^S$  we get for any  $i = 1, \dots, \ell^S$ ,

$$\begin{aligned} \sum_{j=1, j \neq i}^{\ell^S} r_j^S &= \\ \left( \sum_{j=1}^{\ell^S} r_j^S \right) - r_i^S &\geq \\ (s(V) + 2) - u^S &. \end{aligned}$$

Consequently,

$$\Phi_S \leq -\frac{1}{2} \sum_{i=1}^{\ell^S} r_i^S \times [s(V) + 2 - u^S] =$$

$$\begin{aligned}
& - \frac{1}{2}[s(V) + 2] \times [s(V) + 2 - u^S] = \\
& - \frac{1}{2}[2 \log V(1 \pm o(1))] \times [s(V) + 2 - u^S].
\end{aligned}$$

so,

$$p^S \leq V^{-[s(V)+2-u^S](1 \pm o(1))}.$$

This proves the second upper-bound.

For the third upper-bound, let  $V_j$  achieve the maximal intersection  $|S \cap V_j| = u^S$ . Then the  $u^S \times (s(V) + 2 - u^S)$  edges connecting  $(S \cap V_j)$  to  $(S \setminus V_j)$  are independently picked by  $\mathcal{G}_V^{Indp}$ , each w.p.  $\frac{1}{2}$ , which implies that  $p^S \leq 2^{-u^S[s(V)+2-u^S]}$ . ■ (claim 9.3)

#### **Proof for claim 9.4 -**

Recall that for  $S \in \mathbb{S}$ , the vector  $\vec{r}^S = (r_1^S, \dots, r_{\ell^S}^S)$  indicates the cardinalities of the non-empty intersections  $S \cap V_j$ . We first fix some  $\ell > 2$ , and prove that if  $S^*$  maximizes  $p^S$  among all vertex-sets having  $\ell^S = \ell$ , then  $\vec{r}^{S^*}$  is some permutation of  $(s(V) + 3 - \underbrace{\ell, 1, \dots, 1}_{\ell-1 \text{ elements}})$ .

Indeed, assume w.l.o.g. that  $r_1^{S^*} \geq \dots \geq r_{\ell}^{S^*}$ . It suffices to prove that  $r_2^{S^*} = 1$ . Assume towards contradiction that  $r_2^S \geq 2$ . Clearly, there exists another vertex-set  $S' \in \mathbb{S}$  having  $\vec{r}^{S'}$  identical to  $\vec{r}^{S^*}$  except for  $r_2^{S'} = r_2^{S^*} - 1$  and  $r_1^{S'} = r_1^{S^*} + 1$ .

$$\begin{aligned}
& \text{Since for any vertex-set } S, p^S = 2^{-(\binom{s(V)+2}{2} + \sum_{i=1}^{\ell^S} \binom{r_i^S}{2})}, \text{ then,} \\
& \frac{p^{S'}}{p^{S^*}} = \\
& \frac{2^{\binom{r_1^{S^*}+1}{2} + \binom{r_2^{S^*}-1}{2}}}{2^{\binom{r_1^{S^*}}{2} + \binom{r_2^{S^*}}{2}}} = \\
& 2^{(r_1^{S^*} - r_2^{S^*} + 1)} \geq 2,
\end{aligned}$$

which contradicts the optimality of  $S^*$ . Therefore  $\vec{r}^{S^*} = (s(V) + 3 - \ell, \underbrace{1, \dots, 1}_{\ell-1 \text{ elements}})$ .

Hence, to upper-bound  $p^S$  where  $\ell^S = \ell$ , we may assume w.l.o.g. that  $\vec{r}^S = (s(V) + 3 - \ell, \underbrace{1, \dots, 1}_{\ell-1 \text{ elements}})$ .

Next, by part 1 in claim 9.3

$$p^S = 2^{-\frac{1}{2} \sum_{i=1}^{\ell} r_i^S} \cdot (\sum_{j=1, j \neq i}^{\ell} r_j^S).$$

If  $\ell > c \log V$  we get,

$$\begin{aligned} & \sum_{i=1}^{\ell} r_i^S \cdot (\sum_{j=1, j \neq i}^{\ell} r_j^S) = \\ & r_1^S \cdot (\sum_{j=1, j \neq 1}^{\ell} r_j^S) + \sum_{i=2}^{\ell} r_i^S (\sum_{j=1, j \neq i}^{\ell} r_j^S) = \\ & (s(V) + 3 - \ell)(\ell - 1) + (\ell - 1)(s(V) + 1) = \\ & (\ell - 1)(2s(V) - \ell + 4) = \\ & (\ell - o(\log V))(4 \log V - \ell \pm o(\log V)) = \\ & \ell(4 \log V - \ell) \pm o((\log V)^2). \end{aligned}$$

Thus,

$$p^S \leq V^{-\frac{1}{2} \ell(4 - \frac{\ell}{\log V}) + o(\log V)},$$

which proves the first upper-bound.

Next, if  $\ell > 2$  we may still assume that  $\vec{r}^S = (s(V) + 3 - \ell, \underbrace{1, \dots, 1}_{\ell-1 \text{ elements}})$ . hence,

$$\begin{aligned} p^S &= 2^{-\binom{s(V)+2}{2} + \sum_{j=1}^{\ell} \binom{r_j^S}{2}} \leq \\ & 2^{-\binom{s(V)+2}{2} + \binom{u^S}{2}} = \\ & 2^{-\binom{s(V)+2}{2} + \binom{s(V)+3-\ell}{2}}, \end{aligned}$$

which proves the second upper-bound. ■ (claim 9.4)

This completes the proof for fact 9 which completes the entire proof

of Theorem 4.2. ■

**Remark 4.2** It is easy to verify that if we use forced independent sets of size  $|V_j| = V^c$  where  $0 < c < 1$ , rather than having  $|V_j| = \Theta(\log V)$ , then we still get pseudo-random graphs. However, this time we are guaranteed to have independence number at least  $V^c$  and chromatic number at most  $V^{1-c}$ .

## 4.6 Pseudo-Random Graphs Preserving Sparse Monotone Properties

For a fixed  $c < 1$ , a random graph is  $\lfloor V^c \rfloor$ -connected, Hamiltonian, and has a perfect matching with overwhelming probability. Formally, consider the graph property  $X_c$  indicating whether a graph is  $\lfloor V^c \rfloor$ -connected, Hamiltonian, and has a perfect matching. Then the distributional graph property  $P_c = (X_c, 1, 0)$  is a random graphs property as proved in sections A.1 A.3 A.4 in the appendix. This monotone property  $P_c$  is an example for what we call efficiently computable sparse properties (ECS-properties in short). We start with a construction of pseudo-random graphs preserving property  $P_c$ , and later define ECS-properties and generalize our construction to obtain pseudo-random graphs preserving arbitrary ECS-properties.

### 4.6.1 Handling Hamiltonicity, Perfect Matching, and $V^c$ -Connectivity

Informally, our construction goes as follows. For a fixed  $V$ , we start by taking a fixed "sparse" graph  $g_V^{Fix}$  for which  $X_c(g_V^{Fix}) = 1$ . We next take some pseudo-random graph  $\mathcal{G}_V^{Psd}$  and we add to  $\mathcal{G}_V^{Psd}$  the edges of the graph  $\pi(g_V^{Fix})$ , where  $\pi$  is some "random looking" permutation on the vertices of  $g_V^{Fix}$ . This

means that an edge  $\{u, w\}$  appears in  $\pi(g_V^{Fix})$  iff the edge  $\{\pi^{-1}(u), \pi^{-1}(w)\}$  appears in  $g_V^{Fix}$ . We thus obtain the final graph  $\mathcal{G}_V^{Pc} = OR(\mathcal{G}_V^{Psd}, \pi(g_V^{Fix}))$  in which an edge  $e$  appears if it either appears in  $\mathcal{G}_V^{Psd}$  or in  $\pi(g_V^{Fix})$ . Having  $g_V^{Fix}$  appear as a sub-graph in the final graph  $\mathcal{G}_V^{Pc}$  ensures that  $X_c(\mathcal{G}_V^{Pc}) = 1$ , whereas the fact that  $g_V^{Fix}$  is sparse and that  $\pi$  is random looking is shown to imply that the pseudo-randomness of  $\mathcal{G}_V^{Psd}$  is preserved by  $\mathcal{G}_V^{Pc}$ .

We start by constructing the fixed graphs  $g_V^{Fix}$ .

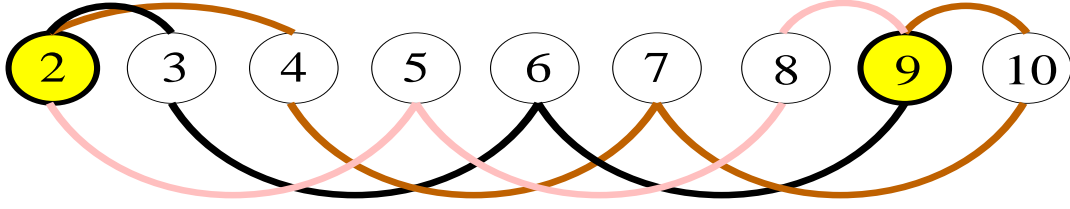
**Construction 4.3** *The graphs series  $\{g_V^{Fix}\}_{V \in \mathbb{N}}$  is defined as follows. Each  $g_V^{Fix} \in G_V$  and each vertex  $u \in g_V^{Fix}$  is connected to all vertices  $(u+i) \bmod V$  for  $i = 1, 2, \dots, \lfloor V^c \rfloor$ , and for  $i = -1, -2, \dots, -\lfloor V^c \rfloor$ .*




**Claim 4.5** *Construction 4.3 provides graphs s.t. for all  $V$ ,  $X_c(g_V^{Fix}) = 1$ .*

**Proof** - Fixing  $V$ , let  $g = g_V^{Fix}$ . An Hamiltonian path in  $g$  is given by the edges  $\{w, (w+1) \bmod V\}$  for  $w = 0, 1, \dots, V-1$ , and taking each other edge in this path provides a perfect matching.

Finally, fixing two vertices  $w_1, w_2$  in  $g$  we provide  $k = \lfloor V^c \rfloor$  disjoint paths between them.

For example, consider the case where  $V = 10$  and  $k = 3$ . The following figure shows the 3 required disjoint paths between vertex 2 to vertex 9.



First path   
 Second path   
 Third path 

Formally, assume w.l.o.g. that  $w_1 < w_2$ . Let  $m$  be the minimal integer s.t.  $w_2 = w_1 + mk + \ell$  for some  $1 < \ell \leq k$ . Then the required paths  $p_1, \dots, p_k$  are:

$$p_i = \begin{cases} (w_1 \mapsto (w_1 + \ell) \mapsto (w_1 + \ell + k) \mapsto \dots \mapsto (w_1 + \ell + mk) = w_2) & i = \ell \\ (w_1 \mapsto (w_1 + i) \mapsto (w_1 + i + k) \mapsto \dots \mapsto (w_1 + i + mk) \mapsto w_2) & i \neq \ell \end{cases}$$

■ (claim 4.5).

We next define the graphs ensemble  $\mathcal{G}^{P_c}$  given by  $\pi(g^{Fix})$ .

**Construction 4.4** *Let  $I$  be the set of prime numbers. The graphs ensemble  $\mathcal{G}^{P_c} = \{\mathcal{G}_V^{P_c}\}_{V \in I}$  is defined as follows. Given a prime  $V$ , uniformly pick a pair  $(a, b) \in \{1, 2, \dots, V-1\} \times \{0, 1, \dots, V-1\}$ , and let  $\pi_{a,b}(w) = (a \cdot w + b) \bmod V$ . Finally, let  $\mathcal{G}_V^{P_c} = \pi_{a,b}(g_V^{Fix})$ .*

**Remark 4.3** By the primality of  $V$ , the pair  $(a, b)$  defines a permutation  $\pi_{a,b}$  over the vertices  $\{0, 1, \dots, V-1\}$  as desired.

We finally construct the pseudo-random graphs  $\mathcal{G}$  defined for an arbitrary prime order  $V$  and preserving the random graphs property  $P_c$ .

**Construction 4.5 Pseudo-Random Graphs  $\mathcal{G}$  Preserving Property**

$P_c$  - Let  $I$  be the set of prime numbers. Let  $\mathcal{G}^{Psd}$  be some family of pseudo-random graphs over index set  $I$  and let  $\mathcal{G}^{Pc}$  be as in construction 4.4. The graphs ensemble  $\mathcal{G}$  is defined by  $\mathcal{G} = OR(\mathcal{G}^{Psd}, \mathcal{G}^{Pc})$ .

**Remark 4.4** For convenience we denote the ordered pair  $(\pi_{a,b}^{-1}(w_1), \pi_{a,b}^{-1}(w_2))$  by  $\pi_{a,b}^{-1}(w_1, w_2)$ . It is easy to verify that for arbitrary vertices  $w_1 \neq w_2, u_1 \neq u_2$ , it holds that  $\pi_{a,b}^{-1}(w_1, w_2) = (u_1, u_2) \iff a = \frac{w_2 - w_1}{u_2 - u_1}$  and  $b = w_1 - u_1 \cdot a$  (arithmetic is done in the field  $\mathbb{Z}_V$ ). Consequently, taking probabilities over the random picking of the pair  $(a, b)$  gives,

$$\Pr[\pi_{a,b}^{-1}(w_1, w_2) = (u_1, u_2)] = \frac{1}{V-1} \times \frac{1}{V}.$$

**Theorem 4.3** Recall that for  $c < 1$  we let  $X_c$  denote the graph property indicating whether a graph is  $[V^c]$ -connected, Hamiltonian, and has a perfect matching. Then, assuming that pseudo-random functions exist, the graphs of construction 4.5 are pseudo-random graphs preserving the distributional property  $P_c = (X_c, 1, 0)$ .

**Proof -** The graphs of construction 4.5 are efficiently computable since the graphs  $\mathcal{G}^{Psd}$  and  $\{g_V^{Fix}\}_{V \in \mathbb{N}}$  are efficiently computable and so is the permutation  $\pi_{a,b}^{-1}$  when the pair  $(a, b)$  is known. Next, since  $g_V^{Fix}$  appears as a sub-graph in any constructed graph of order  $V$ , the graphs of construction 4.5 clearly preserve property  $P_c$ .

We finally prove the pseudo-randomness of these graphs. Recall Lemma 3.1 stating that mild modifications preserve pseudo-randomness. By this Lemma, it suffices to fix an arbitrary edge  $e = \{w_1, w_2\}$ , and to prove that

the probability that  $e$  appears in  $\pi_{a,b}(g_V^{Fix})$  is negligibly small in  $n(V)$ . Here probabilities are taken only over the permutation  $\pi_{a,b}$ , so the probability space is simply the uniform distribution of  $(a,b)$  over  $\{1, 2, \dots, V-1\} \times \{0, 1, \dots, V-1\}$ . Indeed, using the notation  $e \in g$  to denote that the edge  $e$  appears in the graph  $g$  we get

$$\begin{aligned} \Pr [\{w_1, w_2\} \in \pi_{a,b}(g_V^{Fix})] &= \\ \Pr [\pi_{a,b}^{-1}(w_1, w_2) \in g_V^{Fix}] &= \\ \sum_{\{u_1, u_2\} \in g_V^{Fix}} \Pr [(\pi_{a,b}^{-1}(w_1, w_2) = (u_1, u_2))] &= \quad (\text{By remark 4.4}) \\ E(g_V^{Fix}) \times \frac{1}{V(V-1)} &= \quad (\text{Recall } E(g) \text{ counts the number of edges in } g) \\ (\lfloor V^c \rfloor V) \times \frac{1}{V(V-1)}. \end{aligned}$$

As the last term is negligibly small in  $n(V)$  this completes the proof.

■ (Theorem 4.3).

**Remark 4.5** We note that the only property of  $g_V^{Fix}$  used in this proof is being sparse. Namely, having  $\frac{E(g_V^{Fix})}{V^2}$  negligibly small in  $n(V)$ .

## 4.6.2 Handling General ECS-Properties

Similar constructions can be used to obtain pseudo-random graphs preserving other monotone properties we call ECS-properties. To define ECS-properties, we first introduce ECS-graphs.

### Definition 4.8 Efficiently Computable Sparse Graphs (ECS Graphs)-

*A series of graphs (one graph per size  $V$ )  $\{g_V\}_{V \in \mathbb{N}}$ , where each  $g_V \in G_V$  is called efficiently computable sparse graphs if it is,*

- **Efficiently Computable** - *Given as input a potential edge  $e = \{u, w\}$ , we can decide in polynomial time in  $n(V)$  whether  $e$  appears in  $g_V$ .*

- **Sparse** - The fraction  $\frac{E(g_V)}{V^2}$  is negligibly small in  $n(V)$ .

Examples of ECS-graphs follow:

- The graphs of construction 4.3 (providing  $\lfloor V^c \rfloor$ -connectivity).
- The graphs  $\{g_V^{Clique}\}_{V \in \mathbb{N}}$ , where each  $g_V^{Clique}$  is a clique on the first  $\lfloor V^c \rfloor$  vertices for a fixed  $c < 1$ .
- The graphs  $\{g_V^{Star}\}_{V \in \mathbb{N}}$ , where all vertices are connected only to the first vertex (providing diameter=2).

We next define ECS-properties as properties implied by ECS-graphs in the following sense.

**Definition 4.9 ECS Graph Properties-** *A monotone Boolean graph property  $\bar{X}$  is called an ECS-property, if there exists ECS graphs  $\{\bar{g}_V\}_{V \in \mathbb{N}}$ , s.t. for any graph  $g \in G_V$  containing a copy of  $\bar{g}_V$  as a sub-graph,  $\bar{X}(g) = 1$ . We then say that  $\{\bar{g}_V\}_{V \in \mathbb{N}}$  implies the property  $\bar{X}$ .*

For instance, having diameter at most 2 is an ECS property implied by the ESC graphs  $g_V^{Star}$ , since any graph  $g \in G_V$  containing the star  $g_V^{Star}$  as a sub-graph has diameter at most 2. Similarly, containing a  $V^c$ -Clique (for some fixed  $c < 1$ ) is also an ECS-property implied by the ECS graphs  $g_V^{Clique}$ .

**Theorem 4.4** *Consider an arbitrary ECS-property  $\bar{X}$ . Then, assuming that pseudo-random functions exist, there exist pseudo-random graphs preserving the distributional graph property  $\bar{P} = (\bar{X}, 1, 0)$ .*

**Proof** - Let  $\{\bar{g}_V\}_{V \in \mathbb{N}}$  be some ECS graphs implying property  $\bar{X}$ . We claim that replacing the ECS graphs  $g_V^{Fix}$  in construction 4.5 with the ECS graphs  $\bar{g}_V$ , yields pseudo-random graphs preserving the distributional graph property  $\bar{P}$ . Indeed, in the new construction, the generated graph  $\mathcal{G}_V$  of order  $V$  contains a copy of  $\bar{g}_V$  as a subgraph, so clearly  $\bar{X}(\mathcal{G}_V) = 1$ . Next, to obtain the required pseudo-randomness, recall remark 4.5 stating that the only property of  $g_V^{Fix}$  used to prove pseudo-randomness in Theorem 4.3 was being ECS-graphs. Consequently, a similar proof would hold for arbitrary ECS-graphs. ■ (Theorem 4.4).

We finally note that some ECS-properties are random graphs properties, like Hamiltonicity,  $V^c$ -connectivity (for a fixed  $c < 1$ ), containing some fixed sub-graph  $g$  (say, a triangle), or having diameter at most 2. However, other ECS-properties boldly defy random graphs properties. For instance containing a clique of size  $V^c$  for some fixed  $c < 1$  is an ECS-property, whereas random graphs rarely have cliques larger than  $2 \log V$ .

## 4.7 Pseudo-Random Graphs Approximating the Connectivity Number, and Minimal and Maximal Degrees of Random Graphs

Since an arbitrary vertex in a random graph typically has degree  $\approx \frac{1}{2}V$ , it is not surprising that random graphs have minimal and maximal degree  $\approx \frac{1}{2}V$ . As the connectivity number of a graph equals the minimal number of disjoint paths connecting any pair of vertices, the maximal degree clearly

upper bounds the connectivity number. Actually, the connectivity number of random graphs meets this upper bound, namely, all pairs of vertices in a random graph do have  $\approx \frac{1}{2}V$  disjoint paths.

It is unclear whether we can construct pseudo-random graphs s.t. every vertex has the desired  $\approx \frac{1}{2}V$  neighbors, and each pair of vertices has the desired  $\approx \frac{1}{2}V$  disjoint paths. However, we can prove that for **arbitrary** pseudo-random graphs, all graphs (apart from a set of graphs having negligible probability) approximate the required connectivity number and the required maximal and minimal degree in the following sense.

- All but a negligible fraction of the vertices have  $\approx \frac{1}{2}V$  neighbors.
- All but a negligible fraction of the pairs of vertices have  $\approx \frac{1}{2}V$  disjoint paths.

### 4.7.1 Handling the Connectivity Number

To obtain more quantitative statements the following definitions are used.

**Definition 4.10  $\Delta$ -Disjoint Pairs** - *A pair of vertices  $\{u, w\}$  in a graph of order  $V$  is  $\Delta$ -disjoint if it has less than  $\frac{1}{2}V(1 - \Delta)$  disjoint paths.*

**Definition 4.11  $\Delta$ -Disjoint Graphs** - *A graph  $g$  of order  $V$  is  $\Delta$ -disjoint if at least a fraction  $\Delta$  of its vertex-pairs are  $\Delta$ -disjoint.*

We next formalize the high connectivity property of random graphs, which pseudo-random graphs would be shown to preserve. Informally, we wish the probability of picking a  $(\frac{1}{(n(V))^k})$ -disjoint graph to be negligible for any fixed  $k$ .

**Theorem 4.5 High Connectivity of Random and Pseudo-Random Graphs** - For a fixed  $k$ , set  $\Delta_V = \frac{1}{(n(V))^k}$ , and let  $X_k$  denote the graph property indicating whether a graph  $g$ , is **not**  $\Delta_{V(g)}$ -disjoint. Then for any  $k > 0$  the distributional graphs property  $P_k = (X_k, 1, 0)$  is a random graphs property and is also maintained by **any** ensemble of pseudo-random graphs.

**Proof Idea -**

The proof for random graphs appears in section A.4 in the appendix. We complete the proof by assuming that some pseudo-random graphs fail to preserve the high connectivity property, and by showing that this leads to a contradiction to the definition of pseudo-randomness.

The central observation is that when we analyze the number of disjoint paths connecting an arbitrary pair of vertices  $\{u, w\}$ , it suffices to consider only short disjoint paths as follows.

Fixing the pair  $\{u, w\}$ , it turns out that for random graphs we can expect to have  $\approx \frac{1}{4}V$  disjoint paths of length 2, and even more  $\approx \frac{1}{4}V$  disjoint paths of length 3 given by a matching between the neighbors of  $u$  which are not neighbors of  $w$  and the neighbors of  $w$  which are not neighbors of  $u$ . **These two types of paths are consequently referred to as 'short disjoint paths'.**

On the other hand, assume towards contradiction that with a non-negligible probability a pseudo-random graph is  $\Delta_V$ -disjoint. For such graphs, a non-negligible fraction of the vertex pairs  $\{u, w\}$  must have few disjoint paths, and in particular these pairs have less than  $\frac{1}{2}V(1 - \Delta_V)$  short disjoint paths.

Therefore, we can hope to distinguish between random and pseudo-random graphs by correctly estimating the fraction of pairs  $\{u, w\}$  having 'many' or

'few' short disjoint paths.

### The Formal Proof -

Assume towards contradiction that some pseudo-random graphs  $\mathcal{G}^{Psd} = \{\mathcal{G}_V^{Psd}\}_{V \in \mathbb{N}}$  fail to preserve high connectivity. This implies that for some fixed  $k > 0$ , setting  $\Delta_V \stackrel{\text{def}}{=} \frac{1}{(n(V))^k}$ , the following holds. For infinitely many values of  $V$ , the probability that  $\mathcal{G}_V^{Psd}$  is  $\Delta_V$ -disjoint is greater than  $\Delta_V$ .

Towards a contradiction we will construct an efficient algorithm  $D$ , distinguishing  $\mathcal{G}^{Psd}$  from random graphs. On input  $1^{n(V)}$  and given oracle access to a graph  $g \in G_V$ , the execution of  $D^g(1^{n(V)})$  proceeds as follows. The distinguisher  $D$  uniformly picks a random vertex pair  $\{u, w\}$  appearing in  $g$ , then executes  $T^g(1^{n(V)}, u, w)$  and finally replies either 'random' or 'pseudo-random' as  $T$  does. We will provide a randomized test  $T$ , correctly deciding whether the vertex pair  $\{u, w\}$  has 'few' or 'many' short disjoint paths, in the following sense.

#### **Requirement 4.1**

1. For any graph  $g \in G_V$ , and any  $\Delta_V$ -disjoint pair  $\{u, w\}$  in  $g$ ,  
 $T^g(1^{n(V)}, u, w) = \text{'pseudo-random'}$  w.p.  $1 - o(1)$   
*(Probabilities taken only on the internal coin tosses of  $T$ ).*
2. On a random pair  $\{u, w\}$  taken from a random graph  $\mathcal{G}$  of order  $V$ ,  
 $T^{\mathcal{G}}(1^{n(V)}, u, w) = \text{'pseudo-random'}$  w.p.  $V^{-\Omega(1)}$   
*(Probabilities taken on the internal coin tosses of  $T$  as well as on the random choices of  $\mathcal{G}$  and  $\{u, w\}$ ).*

Indeed, the distinguisher  $D$  distinguishes  $\mathcal{G}^{Psd}$  from random graphs:

**Claim 4.6** *Given an algorithm  $T$  maintaining requirement 4.1, the distinguisher  $D$  has a non-negligible advantage in distinguishing  $\mathcal{G}^{Psd}$  from random graphs.*

**Proof -** Given oracle access to a **random graph**  $\mathcal{G}$  of order  $V$ , and taking the following probabilities on the internal coin tosses of  $D$  and  $T$  as well as on the distribution of random graphs, part 2 in requirement 4.1 implies that  $\Pr [D^{\mathcal{G}}(1^{n(V)}) = \text{'pseudo-random'}] \leq V^{-\Omega(1)}$ .

On the other hand, given oracle access to a **pseudo-random graph**  $\mathcal{G}' = \mathcal{G}_V^{Psd}$ , and taking the following probabilities on the internal coin tosses of  $D$  and  $T$  as well as on the distribution of  $\mathcal{G}'$  gives,

$$\begin{aligned} \Pr [D^{\mathcal{G}'}(1^{n(V)}) = \text{'pseudo-random'}] &\geq \\ &\Pr [\mathcal{G}' \text{ is a } \Delta_V\text{-disjoint graph}] \times \\ &\Pr [D \text{ picked a } \Delta_V\text{-disjoint pair } \{u, w\} \mid \mathcal{G}' \text{ is } \Delta_V\text{-disjoint}] \times \\ &\Pr [T^{\mathcal{G}'}(1^{n(V)}, u, w) = \text{'pseudo-random'} \mid \{u, w\} \text{ is } \Delta_V\text{-disjoint}] \geq \\ &\Delta_V \times \Delta_V \times (1 - o(1)) = \Omega(n^{-2k}). \end{aligned}$$

■ (claim 4.6).

Theorem 4.5 clearly follows from claim 4.6, so it suffices to provide the tester  $T$  maintaining requirement 4.1.

To this end, recall  $\Gamma(v)$  denotes the neighbors-set of a vertex  $v$ . For a fixed vertex pair  $\{u, w\}$ , let  $B = \Gamma(u) \cap \Gamma(w)$ ,  $M = \Gamma(u) \setminus \Gamma(w)$ , and  $W = \Gamma(w) \setminus \Gamma(u)$ . Informally, on input  $\{u, w\}$ ,  $T$  first runs a test  $T_1$  to estimate whether  $|B|$  is as large as expected in a random graph, and then runs a test  $T_2$  to estimate whether the matching between  $M$  and  $W$  is as large as expected in a random graph. Formally,

**The Tester  $T$** 

Given as input the string  $1^{n(V)}$  and a vertex-pair  $\{u, w\}$  from a graph  $g \in G_V$ , and given oracle access to  $g$ ,  $T$  executes as follows:

1. **Step 1** -  $T$  executes  $T_1^g(1^{n(V)}, u, w)$ . If  $T_1$  outputs 'pseudo-random', then  $T$  outputs 'pseudo-random'. Otherwise  $T$  continues to step 2.
2. **Step 2** -  $T$  executes  $T_2^g(1^{n(V)}, u, w)$  and outputs either 'random' or 'pseudo-random' as  $T_2$  does.

Before providing the testers  $T_1, T_2$ , note that for a  $\Delta_V$ -disjoint pair, at least one of following conditions holds: Either  $|B| < \frac{1}{4}(1 - \Delta_V)$ , or the maximal matching between  $M$  and  $W$  is smaller than  $\frac{1}{4}(1 - \Delta_V)$ .

Therefore to achieve requirement 4.1 it suffices to obtain:

**Requirement 4.2**

1. Whenever  $|B| < \frac{1}{4}(1 - \Delta_V)$ ,  $T_1^g(1^{n(V)}, u, w) = \text{'pseudo-random'}$  w.p.  $1 - o(1)$   
(Probabilities taken only on the internal coin tosses of  $T_1$ ).
2. Whenever the maximal matching between  $M$  and  $W$  is smaller than  $\frac{1}{4}(1 - \Delta_V)$ ,  $T_2^g(1^{n(V)}, u, w) = \text{'pseudo-random'}$  w.p.  $1 - o(1)$   
(Probabilities taken only on the internal coin tosses of  $T_1$ ).
3. On a random vertex-pair  $\{u, w\}$  taken from a random graph  $\mathcal{G}$  of order  $V$ ,  
 $T_j^{\mathcal{G}}(1^{n(V)}, u, w) = \text{'pseudo-random'}$  w.p.  $2^{-\Omega(n)}$ ,  $j = 1, 2$

(Probabilities taken on the internal coin tosses of  $T$  as well as on the random choices of  $\mathcal{G}$  and  $\{u, w\}$ ).

**The Test  $T_1$**

**Input:**

- 1) The string  $1^{n(V)}$ .
- 2) A vertex-pair  $\{u, w\}$  taken from a graph  $g \in G_V$ .
- 3) Oracle access to the graph  $g$ .

**Execution:**

Uniformly and independently pick  $m = n(\frac{1}{\Delta_V})^2$  vertices  $v_i \neq u, w$  (possibly with repetitions) obtaining a multi-set  $S = \{v_1, \dots, v_m\}$ . Reply 'pseudo-random' if  $|B \cap S| \leq \frac{1}{4}m(1 - \frac{1}{2}\Delta_V)$  and otherwise reply 'undecided'.

**Claim 4.7** *Items 1,3 in requirement 4.2 hold for  $T_1$ .*

**Proof -** Both parts are easy to prove. For instance, we handle item 1. Indeed, when  $|B| < \frac{1}{4}(1 - \Delta_V)$ , one can consider testing whether  $v_1, \dots, v_m \in B$  to be  $m$  independent Bernoulli trials each w.p. of success  $p < \frac{1}{4}(1 - \Delta_V)$ . Consequently, the Chernoff Bound implies that

$$\Pr [T_1^g(1^{n(V)}, u, w) \neq \text{'pseudo-random'}] =$$

$$\Pr [|B \cap S| \geq \frac{1}{4}m(1 - \frac{1}{2}\Delta_V)] < e^{-\frac{1}{p}(\frac{1}{2}\Delta_V)^2 m} = 2^{-\Omega(n)}.$$

■ (claim 4.7).

We proceed with an informal description of the test  $T_2$ . Recall  $m = n(\frac{1}{\Delta_V})^2$  and assume w.l.o.g. that  $m$  divides  $V$ . Consider the partitioning of the vertices into  $\frac{V}{m}$  disjoint sets  $S_1, \dots, S_{\frac{V}{m}}$  each of size  $m$ , given by  $S_i = \{(i-1)m + j | j = 0, 1, \dots, m-1\}$ . Note that for each  $S_i$  we can efficiently

compute the size of the maximal matching between the 2 vertex-sets  $(M \cap S_i)$  and  $(W \cap S_i)$ . We call these matchings '**local matchings**'. Unfortunately there are  $2^{\Omega(n)}$  many sets  $S_i$ , so our goal is only to estimate the fraction of sets  $S_i$  having a 'large' local matching. This is done by checking the matching for  $m$  sets  $S_i$  we pick at random.

Formally, let  $S_i$  be as above and set  $\bar{\Delta}_V$  s.t.  $(1 - \bar{\Delta}_V)^2 = (1 - \Delta_V)$ .

**The Test  $T_2$**

**Input:**

- 1) The string  $1^{n(V)}$ .
- 2) A vertex-pair  $\{u, w\}$  taken from a graph  $g \in G_V$ .
- 3) Oracle access to the graph  $g$ .

**Execution:**

Uniformly and independently pick  $m$  indices  $i_1, \dots, i_m \in \{1, 2, \dots, \frac{V}{m}\}$  (possibly with repetitions). For each index  $i_j$  calculate the maximal matching between  $(M \cap S_{i_j})$  and  $(W \cap S_{i_j})$ . Reply 'pseudo-random' if for at least a single index  $i_j$  the matching is smaller than  $\frac{1}{4}m(1 - \bar{\Delta}_V)$ . Otherwise reply 'random'.

**Claim 4.8** *Conditions 2,3 in requirement 4.2 hold for  $T_2$ .*

**Proof** - We start with condition 1 in requirement 4.2. Consider a pair  $\{u, w\}$  for which the maximal matching between  $M$  and  $W$  is smaller than  $\frac{1}{4}(1 - \Delta_V)$ . We wish to prove that  $T_2^g(1^n, u, w)$  almost surely replies 'pseudo-random'. We define a **large local matching** as a matching of size  $\frac{1}{4}m(1 - \bar{\Delta}_V)$ , between 2 vertex-sets  $(M \cap S_i)$  and  $(W \cap S_i)$ . Now, necessarily, at most a  $1 - \bar{\Delta}_V$  fraction of the sets  $S_i$  have a large local matching. Otherwise even by considering only the  $S_i$ -s with large local matchings we get  $\frac{V}{m} \times (1 - \bar{\Delta}_V)$  local matching each of size  $m(1 - \bar{\Delta}_V)$ , and we can combine these disjoint matchings into a single matching of size  $\frac{1}{4}(1 - \Delta_V)$ . Consequently, one can consider each test whether  $S_{i_1}, \dots, S_{i_m}$  has a large local matching, as  $m$  independent Bernoulli trials each w.p. of success  $p < 1 - \bar{\Delta}_V$ . Therefore,

$$\begin{aligned} \Pr [T_2^g(1^{n(V)}, u, w) \neq \text{'pseudo-random'}] &= \\ \Pr [\bigcap_{j=1}^m (S_{i_j} \text{ has a large matching})] &= \\ \prod_{j=1}^m \Pr [S_{i_j} \text{ has a large matching}] &\leq \\ (1 - \bar{\Delta}_V)^m &= \\ (1 - \Delta_V)^{\frac{1}{2}m} &\leq \\ e^{-\frac{1}{2}\Delta_V m} &= 2^{-\Omega(n)}. \end{aligned}$$

We continue with condition 2 in requirement 4.2. Indeed for random graphs, it is easy to verify (using the Chernoff Bound again), that w.p.  $1 - 2^{-\Omega(n)}$  all sets  $(S_{i_j} \cap M)$  and  $(S_{i_j} \cap W)$   $j = 1, \dots, m$  are of cardinality  $\geq \frac{1}{4}m(1 - \Delta_V)$ . Next, the fact that random bipartite graphs on  $m' \times m'$  vertices have a perfect matching w.p.  $1 - 2^{-\Omega(m')}$  (proved in section A.3 in the appendix), implies that w.p.  $1 - m2^{-\Omega(m')} = 1 - 2^{-\Omega(n)}$  all pairs of sets  $(S_{i_j} \cap M)$  and  $(S_{i_j} \cap W)$  have large local matching, and then  $T_2$  necessarily

replies 'random'.

■ (claim 4.8).

By this  $T$  is shown to obtain requirement 4.1, which completes the proof of Theorem 4.5. ■

## 4.7.2 Handling the Minimal and Maximal Degrees

To obtain quantitative statements the following definition is used.

**Definition 4.12  $\Delta$ -Approximating  $(\frac{1}{2}V)$ -Regularity** - *A graph of order  $V$ , is said to  $\Delta$ -approximate  $(\frac{1}{2}V)$ -regularity, if all but a  $\Delta$  fraction of the vertices  $w$  have  $|\Gamma(w)| = \frac{1}{2}V(1 \pm \Delta)$ .*

We next formalize the  $(\approx \frac{1}{2}V)$ -regularity of random graphs, which pseudo-random graphs would be shown to preserve. Informally, we wish the probability of picking a graph not  $\frac{1}{(n(V))^k}$ -approximating  $(\frac{1}{2}V)$ -regularity to be negligible for any fixed  $k$ .

**Theorem 4.6 Random and Pseudo-Random Graphs Approximate  $(\frac{1}{2}V)$ -Regularity** - *For a fixed  $k$ , set  $\Delta_V = \frac{1}{(n(V))^k}$ , and let  $X_k$  denote the graph property indicating whether a graph  $g$ ,  $\Delta_{V(g)}$ -approximates  $(\frac{1}{2}V)$ -regularity. Then for any  $k > 0$  the distributional graphs property  $P_k = (X_k, 1, 0)$  is a random graphs property and is also maintained by **any** ensemble of pseudo-random graphs.*

**Proof Sketch** - The case of random graphs is discussed in section A.2 in the appendix. We next assume that some pseudo-random graphs fail to preserve the property  $P_k$ , and show that this leads to a contradiction to the definition of pseudo-randomness.

Indeed, consider the efficient distinguisher  $D$  which given oracle access to a graph  $g \in G_V$  and input  $1^{n(V)}$  performs as follows: Uniformly pick a vertex  $w$  and estimate  $|\Gamma(w)|$  by uniformly picking  $m = n(\frac{1}{\Delta_V})^2$  other vertices  $v_1, \dots, v_m$  and checking the size of  $S = \{v_1, \dots, v_m\} \cap \Gamma(w)$ . Reply 'pseudo-random' if  $|S| < \frac{1}{2}(1 - \frac{1}{2}\Delta_V) \times m$ , and otherwise reply 'random'.

It's easy to verify that the assumption that our pseudo-random graphs defy property  $P_k$ , implies that with non-negligible probability we pick  $w$  and  $v_1, \dots, v_m$  for which  $|S| < \frac{1}{2}(1 - \frac{1}{2}\Delta_V) \times m$ . On the other hand with overwhelming probability the sample  $v$  and  $v_1, \dots, v_m$  taken from a random graph have  $|S| > \frac{1}{2}(1 - \frac{1}{2}\Delta_V) \times m$ . Thus,  $D$  distinguishes random graphs from our pseudo-random graphs with non-negligible probability - a contradiction. ■ (Theorem 4.6).

## 4.8 Simultaneously Preserving the Random-Graphs Properties Previously Handled

Consider an arbitrary  $0 < c < 1$  and  $k > 0$ . Recall that we have set  $\Delta_V = \frac{1}{(n(V))^k}$ , and let  $s(V)$  denote the maximal  $s$  for which the expected number of  $s$ -cliques in a random graph over  $V$  vertices is larger than 1. Also recall we have defined which graphs  $g \in G_V$  are  $\Delta_{V(g)}$ -disjoint (page 66), and which graphs  $\Delta_{V(g)}$ -approximate  $(\frac{1}{2}V)$ -regularity (page 74). We consider the following graph properties:

- The indicator  $X_{ECS}$  of whether a graph  $g$  is  $\lceil(1 - c) \log V\rceil$ -connected, Hamiltonian, and has a perfect matching.

- The indicator  $X_{HighConn}$  of whether a graph  $g$  is not  $\Delta_{V(g)}$ -disjoint, and the indicator  $X_{MinMaxDeg}$  of whether  $g$   $\Delta_{V(g)}$ -approximates  $(\frac{1}{2}V)$ -regularity.
- The graph properties  $\omega, \alpha$  and  $\chi$ , assigning to each graph its clique number, independence number and chromatic number respectively.

We conclude this chapter by providing a single construction of pseudo-random graphs  $\mathcal{G}^{Comb}$  of arbitrary prime order  $V$ , simultaneously preserving the following random graphs' properties:

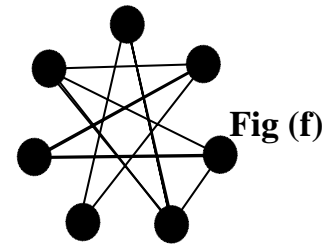
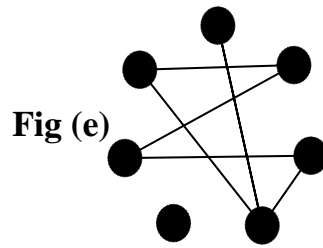
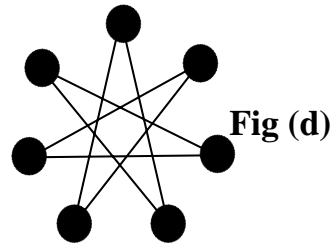
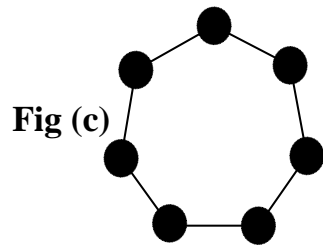
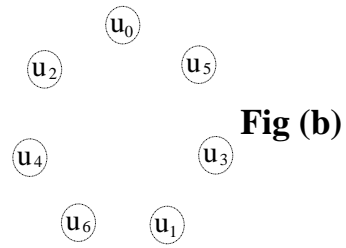
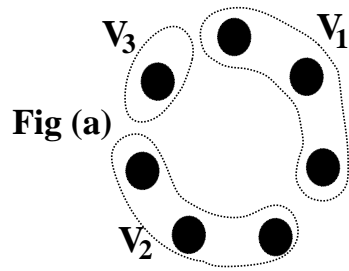
1.  $P_{ECS} = (X_{ECS}, 1, 0)$ .
2.  $P_{HighConn} = (X_{HighConn}, 1, 0)$ .
3.  $P_{MinMaxDeg} = (X_{MinMaxDeg}, 1, 0)$ .
4.  $P_{Clique} = (\omega, s(V), \frac{1}{s(V)})$ .
5.  $P_{Indp} = (\alpha, s(V), \frac{1}{s(V)})$ .
6.  $P_{Color} = (\chi, \frac{V}{s(V)}, \frac{1}{\sqrt{\log(V)}})$ .

We assume for simplicity that  $\lceil \frac{1}{2}(1-c)\log V \rceil$  is an integer, and start by constructing the auxiliary graphs  $\bar{g}^{Fix}$  for which  $X_{ECS}(\bar{g}^{Fix}) = 1$ .

**Construction 4.6** *The graphs series  $\{\bar{g}_V^{Fix}\}_{V \in \mathbb{N}}$  is defined as follows. Each  $\bar{g}_V^{Fix} \in G_V$  and each vertex  $u \in \bar{g}_V^{Fix}$  is connected to all vertices  $(u+i) \bmod V$  for  $i \neq 0, -\frac{1}{2}(1-c)\log V \leq i \leq \frac{1}{2}(1-c)\log V$ .*

**Construction of  $\mathcal{G}_V^{\text{Comb}}$  (Informal Description)** - To ensure having  $X_{ECS}(\mathcal{G}_V^{\text{Comb}}) = 1$ , we obtain  $\mathcal{G}_V^{\text{Comb}}$  by adding a copy of  $\bar{g}_V^{\text{Fix}}$  to  $\mathcal{G}_V^{\text{Color}}$  (Recall that the pseudo-random graphs  $\mathcal{G}_V^{\text{Color}}$  are given by construction 4.2). This means that  $\mathcal{G}_V^{\text{Comb}} = OR(\mathcal{G}_V^{\text{Color}}, \pi(\bar{g}_V^{\text{Fix}}))$ , where  $\pi$  is some "random looking" permutation on the vertices of  $\bar{g}_V^{\text{Fix}}$ . Hence, an edge  $\{u, w\}$  appears in  $\mathcal{G}_V^{\text{Comb}}$  if it either appears in  $\mathcal{G}_V^{\text{Color}}$  or the edge  $\{\pi^{-1}(u), \pi^{-1}(w)\}$  appears in  $\bar{g}_V^{\text{Fix}}$ . The permutation  $\pi$  is defined s.t. the edges of  $\pi(\bar{g}_V^{\text{Fix}})$  never connect vertices inside any of the forced independent sets. Thus the small chromatic number of  $\mathcal{G}_V^{\text{Color}}$  is retained by  $\mathcal{G}_V^{\text{Comb}}$ .

As a simple example, the following figure demonstrates a construction of a specific graph where  $V = 7$ ,  $s(V) = 3$ ,  $V^c = 1$ . Figure (a) shows the forced independent sets  $V_1, V_2, V_3$  of  $\mathcal{G}_V^{\text{Color}}$ . Figure (b) shows the permutation  $\pi = (u_0, u_1, \dots, u_6)$  of the vertices. Figure (c) shows the graph  $\bar{g}_V^{\text{Fix}}$ . Figure (d) shows  $\pi(\bar{g}_V^{\text{Fix}})$ . Figure (e) shows the graph  $\mathcal{G}_V^{\text{Color}}$ . Figure (f) shows the final graph  $\mathcal{G}_V^{\text{Comb}} = OR(\mathcal{G}_V^{\text{Color}}, \pi(\bar{g}_V^{\text{Fix}}))$ . Note that the edges of  $\pi(\bar{g}_V^{\text{Fix}})$  indeed do not connect vertices inside any independent set  $V_j$ .



Formally,

**Construction 4.7** *Let  $I$  be the set of prime numbers. To construct a graph  $\mathcal{G}_V^{Comb}$  according to the ensemble  $\mathcal{G}^{Comb} = \{\mathcal{G}_V^{Comb}\}_{V \in I}$ ,*

**The Initial Stage** - *Construct a  $(4 \log^2(V))$ -wise independent pseudo-random graph  $\mathcal{G}_V^{Indp}$  according to construction 4.1.*

**The Omitting Stage** - *(Resulting in  $\mathcal{G}_V^{Color}$ ) Uniformly pick a shift  $r \in \{1, 2, \dots, V-1\}$ , and let  $w_j = (j \cdot r) \bmod V$ . By the primality of  $V$ ,  $\sigma = (w_0, w_1, w_2, \dots, w_{V-1})$  is a permutation of the vertices. Next, partition the vertices into equivalence classes  $V_j^r$ , where*

$$V_j^r = \begin{cases} \{(i \cdot r) \bmod V \mid (j-1) \cdot s(V) \leq i < j \cdot s(V)\} & 1 \leq j \leq \lfloor \frac{V}{s(V)} \rfloor \\ \{0, 1, \dots, V-1\} \setminus (\bigcup_{\ell=1}^{\lfloor \frac{V}{s(V)} \rfloor} V_\ell^r) & j = \lceil \frac{V}{s(V)} \rceil. \end{cases}$$

*Delete from  $\mathcal{G}_V^{Indp}$  all edges connecting vertices  $w, w'$  inside the same equivalence class  $V_j^r$ , to obtain  $\mathcal{G}_V^{Color}$ .*

**The Addition Stage** - *Define a second permutation over the vertices by letting  $\pi^r(j) = w_{js(V)} = (js(V)r) \bmod V$ . Finally, let  $\mathcal{G}_V^{Comb} = OR(\mathcal{G}_V^{Color}, \pi^r(\bar{g}_V^{Fix}))$ , where  $\bar{g}_V^{Fix}$  is given by construction 4.6.*

**Theorem 4.7** *Assuming that one-way functions exist, then The graphs  $\mathcal{G}^{Comb}$  of construction 4.7 are pseudo-random graphs preserving all random graphs properties 1-6 listed in page 76.*

**Proof** - First, the graphs  $\mathcal{G}^{Comb}$  are a mild modification of the pseudo-random graphs  $\mathcal{G}^{Color}$  and are therefore pseudo-random by Lemma 3.1.

**Properties  $\mathbf{P}_{\text{HighConn}}$  and  $\mathbf{P}_{\text{MinMaxDeg}}$**  - Theorems 4.5 and 4.6 state that properties  $P_{\text{HighConn}}$  and  $P_{\text{MinMaxDeg}}$  hold for arbitrary pseudo-random graphs, and in particular for  $\mathcal{G}^{\text{Comb}}$ .

**Property  $\mathbf{P}_{\text{ECS}}$**  - It's easy to verify (following the proof for claim 4.5) that  $X_{\text{ECS}}(\bar{g}_V^{\text{Fix}}) = 1$ . As each constructed graph  $\mathcal{G}_V^{\text{Comb}}$  contains a copy of  $\bar{g}_V^{\text{Fix}}$  as a sub-graph, we always get  $X_{\text{ECS}}(\mathcal{G}_V^{\text{Comb}}) = 1$  as desired.

**Properties  $\mathbf{P}_{\text{Clique}}$ ,  $\mathbf{P}_{\text{Indp}}$  and  $\mathbf{P}_{\text{Color}}$**  - The main observation, is that for sufficiently large  $V$  the edges added to  $\mathcal{G}_V^{\text{Color}}$  to obtain  $\mathcal{G}_V^{\text{Comb}}$ , do not connect vertices inside the same forced independent-set  $V_j$ .

Indeed, recall that the permutation  $\sigma = (w_0, w_1, w_2, \dots, w_{V-1})$  is given by the omitting stage, and that in  $\bar{g}_V^{\text{Fix}}$  each vertex  $v$  is connected only to vertices  $[(v \pm \ell) \bmod V]$  for  $\ell = 1, \dots, \frac{1}{2}(1-c) \log V$ . Therefore in  $\pi(\bar{g}_V^{\text{Fix}})$  a vertex  $w_i$  is connected only to vertices  $w_m$  where  $m = (i \pm \ell \times s(V)) \bmod V$  for  $\ell = 1, \dots, \frac{1}{2}(1-c) \log V$ . Now clearly, when  $w_i \in V_j$ , then  $w_{(i+\ell \times s(V)) \bmod V} \in [V_{(j+\ell) \bmod V} \cup \dots \cup V_{(j+2\ell) \bmod V}]$ . Consequently, in  $\pi(\bar{g}_V^{\text{Fix}})$ , the neighbors-set of  $w_i$  is given by

$$\Gamma(w_i) = \bigcup_{\ell \neq 0, \ell = -\frac{1}{2}(1-c) \log V}^{\ell = \frac{1}{2}(1-c) \log V} \{w_{[i+\ell \times s(V)] \bmod V}\} \subseteq \bigcup_{t \neq 0, t = j - (1-c) \log V}^{t = j + (1-c) \log V} V_{(t \bmod V)}.$$

This implies that for sufficiently large  $V$ ,  $\Gamma(w_i) \cap V_j = \emptyset$ , and  $\pi(\bar{g}_V^{\text{Fix}})$  connects any vertex  $w_i \in V_j$  only to vertices outside  $V_j$ .

Back to the main proof, recall that  $\omega(g)$ ,  $\alpha(g)$  and  $\chi(g)$  denote the clique-number, independence-number and chromatic-number of a graph  $g$ . As the addition stage connects no vertices inside a forced independent set of  $\mathcal{G}_V^{\text{Color}}$ , the omitting stage still ensures that we always get  $\alpha(\mathcal{G}_V^{\text{Comb}}) \geq s(V)$ , and

$$\chi(\mathcal{G}_V^{Comb}) \leq \lceil \frac{V}{s(V)} \rceil.$$

Next, as we only add edges to  $\mathcal{G}_V^{Color}$  to obtain  $\mathcal{G}_V^{Comb}$ , then  $\mathcal{G}_V^{Comb}$  preserves any monotone increasing property of  $\mathcal{G}_V^{Color}$ . Thus, w.p.  $1 - V^{-\Omega(1)}$  we still have  $\omega(\mathcal{G}_V^{Comb}) \geq s(V) - 1$ ,  $\alpha(\mathcal{G}_V^{Comb}) \leq s(V) + 1$ , and  $\chi(\mathcal{G}_V^{Comb}) \geq \frac{V}{s(V)+1}$ .

Finally, to complete the proof Theorem 4.7, we prove that with overwhelming probability  $\omega(\mathcal{G}_V^{Comb}) \leq s(V) + 1$ . To this end, note that the construction of  $\mathcal{G}_V^{Comb}$  involves exactly 2 random components:

- The random seed  $s'$  deciding the graph  $\mathcal{G}_V^{Indp}$ .
- The random shift  $r$ , deciding both the omitting stage (namely, deciding the forced independent sets  $V_j^r$ ), and the addition stage (namely, deciding the permutation  $\pi^r$ ).

Therefore, it suffices to fix  $V$ , and to fix the independent sets  $V_j = V_j^r$  and the graph  $\pi(\bar{g}_V^{Fix}) = \pi^r(\bar{g}_V^{Fix})$ , and to prove that  $\Pr[s(V) + 2 \text{ cliques appear in } \mathcal{G}_V^{Comb}] \leq V^{-\Omega(1)}$ , where all probabilities discussed henceforth are taken only over the random seed  $s'$  deciding  $\mathcal{G}_V^{Indp}$ .

Indeed, let  $\mathbb{S}$  denote the collection of all vertex-sets  $S$  of cardinality  $s(V) + 2$  having  $|S \cap V_j| \leq 1$  for all  $j$  (We have already handled the case  $|S \cap V_j| \geq 2$  by showing the the addition stage never connects two vertices inside the same  $V_j$ ). Consider the natural ordering of all vertices  $0, 1, \dots, V - 1$ . We say that a vertex  $j$  is a *follower vertex* for a set  $S \in \mathbb{S}$  if  $j \in S$  and there exist another vertex  $i < j, i \in S$  s.t. the edge  $e = \{i, j\}$  appears in  $\pi(\bar{g}_V^{Fix})$ .

Next, let  $\mathbb{W}_k \subseteq \mathbb{S}$  contain only those vertex-sets  $S$  having exactly  $k$  follower vertices. As  $\bar{g}_V^{Fix}$  is  $(1 - c) \log V$  regular, a follower vertex  $j$  of  $S$  can have no more than  $(1 - c) \log V$  internal edges  $\{i, j\}, i \in S$ , that appear in

$\pi(\bar{g}_V^{Fix})$ . As the remaining internal edges of  $S$  are independently picked w.p.  $\frac{1}{2}$  by  $\mathcal{G}_V^{Indp}$ , we get,  $\Pr[S \text{ induces a clique in } \mathcal{G}_V^{Comb}] \leq 2^{-(\binom{s(V)+2}{2})+k(1-c)\log V} = 2^{-(\binom{s(V)+2}{2})}V^{k(1-c)}$ .

We next show that  $|\mathbb{W}_k| \leq \binom{V}{s(V)+2-k}(V^{o(1)})^k$  as follows. Indeed, there are no more than  $\binom{V}{s(V)+2-k}$  possible choices for the non-follower vertices. Once these  $s(V)+2-k$  non-follower vertices  $\bar{v}_1, \dots, \bar{v}_{s(V)+2-k}$  are fixed we pick the  $k$  follower vertices  $v_1, \dots, v_k$  one by one as follows. To pick  $v_j$ , we first pick a previous vertex  $w$  among  $\bar{v}_1, \dots, \bar{v}_{s(V)+2-k}, v_1, \dots, v_{j-1}$ . Having chosen  $w$ , we pick  $v_j$  itself among the neighbors-set of  $w$  in  $\pi(\bar{g}_V^{Fix})$ . For each  $v_j$ , there are no more than  $s(V)+2$  possible choices for the previous vertex  $w$ , and once  $w$  is fixed there are no more than  $(1-c)\log V$  possible choices for  $v_j$  itself. Thus  $|\mathbb{W}_k| \leq \binom{V}{s(V)+2-k}[(1-c)\log V(s(V)+2)]^k = \binom{V}{s(V)+2-k}(V^{o(1)})^k$ . Since  $\binom{V}{s(V)+2-k} = \binom{V}{s(V)+2} \times (V^{-1\pm o(1)})^k$ , we get

$$|\mathbb{W}_k| \leq \binom{V}{s(V)+2}(V^{-1\pm o(1)})^k.$$

Finally, the expected number of vertex-sets  $S \in \mathbb{W}_k$  inducing a clique in  $\mathcal{G}_V^{Comb}$  is upper-bounded by

$$\begin{aligned} E_k &\leq |\mathbb{W}_k| 2^{-(\binom{s(V)+2}{2})} V^{k(1-c)} \leq \\ &\binom{V}{s(V)+2} (V^{-1\pm o(1)})^k 2^{-(\binom{s(V)+2}{2})} V^{k(1-c)} \leq \\ &\binom{V}{s(V)+2} 2^{-(\binom{s(V)+2}{2})} V^{k(-c+o(1))} \leq \\ &\binom{V}{s(V)+2} 2^{-(\binom{s(V)+2}{2})} = V^{-1\pm o(1)} \quad (\text{The equality follows from claim 9.5}). \end{aligned}$$

Thus, the expected number of  $(s(V)+2)$ -cliques appearing in  $\mathcal{G}_V^{Comb}$  is upper-bounded by  $\sum_{k=0}^{s(V)+1} E_k = V^{-\Omega(1)}$ . The proof follows.  $\blacksquare$  (Thm. 4.7).

## 4.9 Strengthening the Pseudo-Randomness of the Constructed Graphs

In this section we discuss strengthening the pseudo-randomness of our constructed graphs in the 2 following ways.

### 4.9.1 Constructing almost $(n^k)$ -wise independent pseudo-random graphs.

So far, we have considered computational indistinguishability from random graphs as the basic formalization for the type of randomness that 'random looking' graphs should retain. Suppose that in addition to that, we wish our graphs to be  $(n^k)$ -wise independent in the sense that the distribution of any  $n^k$  edges should be uniform (or at least close to uniform).

This could be achieved for any prescribed  $k$  as follows. First, several efficient constructions for  $(n^k)$ -wise independent Boolean functions were given over the years (e.g. [J]). Clearly, representing graphs by such Boolean functions immediately gives  $(n^k)$ -wise independent graphs. Next, when arbitrary  $(n^k)$ -wise independent functions are xored with arbitrary pseudo-random functions the resulting functions are both  $(n^k)$ -wise independent and pseudo-random. Finally, we prove that when pseudo-random functions which are also  $(n^k)$ -wise independent are used as the basis for our constructions (rather than using arbitrary pseudo-random functions), then the resulting graphs are almost  $(n^k)$ -wise independent in the sense that the distribution of any  $n^k$  edges is statistically close to uniform.

Indeed, recall that all constructions involve only mild modifications to the original  $(n^k)$ -wise pseudo-random graphs. By definition of a mild mod-

ification, an arbitrary edge is modified with negligible probability. Thus, a union-bound implies that for any set  $S$  of  $n^k$  edges, the probability that some edge  $e \in S$  is modified is negligible as well.

#### 4.9.2 Constructing Strong Pseudo-Random Graphs.

Previously, we have only considered pseudo-randomness w.r.t. polynomial algorithms, but a stronger concept of pseudo-randomness w.r.t. polynomial circuits can be similarly approached. Strong pseudo-randomness implies that only a negligible advantage in distinguishing pseudo-random functions from random functions is allowed, only this time the adversary can be any circuits family  $C = \{C_n\}_{n \in \mathbb{N}}$  where each  $C_n$  takes inputs of length  $n$ , and has polynomial size in  $n$ . A sequence of works [Y, BM, L, HILL, GGM] have shown that strong pseudo-random functions exist (and have explicit constructions) *iff* one-way functions w.r.t. circuits exist.

Consequently, if we assume that one-way functions w.r.t. circuits exist, we may consider any construction given in this work, and replace pseudo-random functions with strong pseudo-random functions. This way we obtain strong pseudo-random graphs preserving exactly the same distributional graph properties that the original constructions maintain. Indeed, the desired distributional graph properties are forced into the resulting graphs regardless of whether pseudo-random graphs or strong pseudo-random graphs are used. As for strong pseudo-randomness, we recall once more that all constructions involve only mild modifications to the original strong pseudo-random functions, and these modifications cannot be detected even by  $poly(n)$ -size circuits, as we have noticed during the proof of Lemma 3.1.

# Chapter 5

## Conclusions

In general, we have managed to provide pseudo-random graphs preserving or at least approximating all random graphs properties we have considered. We have also noticed that we can obtain pseudo-random graphs which exhibit very different properties than random graphs do. The following open questions and directions of research have not been considered in this work:

**Infeasibility Results** - Recall that when we have considered the random graphs property  $P$  of achieving  $(\approx \frac{1}{2}V)$ -connectivity and being  $(\approx \frac{1}{2}V)$ -regular, we haven't provided pseudo-random graphs preserving property  $P$ , but rather settled for some approximation of  $P$ . An open question is to prove the infeasibility of constructing pseudo-random graphs preserving property  $P$ , or any other random graphs property. Of special interest would be an infeasibility result for a property which naturally arises in the study of random graphs. (Indeed, if we settle a 'non-natural' property, we may consider properties related to the notion of Kolmogorov's complexity as follows. Assigning to each graph  $g$  the size of the minimal Turing Machine which enables to an-

swer edge queries on  $g$ , it's easy to verify that random graphs almost surely have super-polynomial description size, whereas pseudo-random graphs have, by definition,  $poly(n)$  descriptions).

**Extending The Query Model** - A natural extension to the notion of pseudo-randomness with respect to  $poly(n)$  edge-queries, is to allow the distinguisher more complex queries (which cannot be efficiently computed using  $poly(n)$  edge-queries). For instance, given a graph  $g$  one may be interested in fixing some  $(\frac{V}{s(V)})$ -coloring of  $g$  and answering (in addition to edge-queries) a sequence of queries where on input vertex  $u$  the color of  $u$  is replied. Several complex queries were considered following this work in [GGN].

**Other Random Objects** - This work demonstrates how to efficiently construct huge 'random looking' objects of size exponential in  $n$ , which preserve 'global' properties of the corresponding random objects. The working example in this Thesis are random undirected graphs, but other objects such as random functions, random matrices, random directed graphs or random bounded-degree graphs are of general interest as well. Some of these examples have been considered in [GGN].

# Appendix A

## Proofs for some Random Graphs Properties

### A.1 Hamiltonicity of Random Graphs

**Lemma A.1** *A random graph of order  $V$  is Hamiltonian w.p.  $2^{-\Omega(\frac{V}{\ln V})}$ .*

**Proof** - Let  $\mathcal{G}_{V,p}$  denote the distribution over the graphs  $G_V$ , where each edge is independently picked w.p.  $p$ . Let  $H$  denote the graph property indicating whether a graph is Hamiltonian or not. Our goal is to prove that  $\Pr[H(\mathcal{G}_{V,0.5}) = 0] \leq 2^{-\Omega(\frac{V}{\ln V})}$ .

To this end, define  $m$  s.t.  $0.5 = m \times \frac{2 \ln V}{V}$ , and assume w.l.o.g. that  $m$  is an integer. Let  $\mathcal{G}_{V, \frac{2 \ln V}{V}}^{(1)}, \dots, \mathcal{G}_{V, \frac{2 \ln V}{V}}^{(m)}$  be  $m$  independent copies of  $\mathcal{G}_{V, \frac{2 \ln V}{V}}$ , and consider the distribution  $\mathcal{G}_V = \bigcup_{j=1}^m \mathcal{G}_{V, \frac{2 \ln V}{V}}^{(j)}$  taken over the graphs  $G_V$ . Namely, an edge appears in  $\mathcal{G}_V$  if it appears in at least one of the graphs  $\mathcal{G}_{V, \frac{2 \ln V}{V}}^{(j)}$ .

First, note that in  $\mathcal{G}_V$  different edges are independently picked, each w.p. smaller than 0.5. Consequently,  
 $\Pr[H(\mathcal{G}_{V,0.5}) = 0] \leq \Pr[H(\mathcal{G}_V) = 0]$ .

Now, clearly, if  $\mathcal{G}_V$  is non-Hamiltonian, then so are all  $\mathcal{G}_{V, \frac{2 \ln V}{V}}^{(j)}$ . Thus,  
 $\Pr[H(\mathcal{G}_V) = 0] \leq$   
 $\Pr[\bigcap_{j=1}^m (H(\mathcal{G}_{V, \frac{2 \ln V}{V}}^{(j)}) = 0)] =$   
 $\prod_{j=1}^m \Pr[H(\mathcal{G}_{V, \frac{2 \ln V}{V}}^{(j)}) = 0] =$   
 $(\Pr[H(\mathcal{G}_{V, \frac{2 \ln V}{V}}) = 0])^m$ .

Finally, we use the well known fact [B1] that  $\Pr[H(\mathcal{G}_{V, \frac{2 \ln V}{V}}) = 0] = o(1)$ ,  
to conclude that for sufficiently large  $V$ ,

$$\Pr[H(\mathcal{G}_{V,0.5}) = 0] \leq$$

$$\Pr[H(\mathcal{G}_V) = 0] \leq$$

$$(\Pr[H(\mathcal{G}_{V, \frac{2 \ln V}{V}}) = 0])^m \leq 0.5^m = 2^{-\Omega(\frac{V}{\ln V})}.$$

■ (Lemma A.1).

## A.2 Minimal and Maximal Degrees of Random Graphs

**Lemma A.2** *A random graph on  $V$  vertices has a minimal and maximal degree*

- $\frac{1}{2}V(1 \pm 2\sqrt{\frac{\log V}{V}})$  w.p.  $1 - V^{-\Omega(1)}$ .
- $\frac{1}{2}V(1 \pm \delta)$  w.p.  $1 - 2^{-\Omega(V)}$ , for any fixed  $\delta$ .

**Proof** - We prove that an arbitrary vertex has the desired degree with overwhelming probability and complete the proof by union bounding over all possible vertices.

Fix an arbitrary vertex  $u$ . For an arbitrary vertex  $w \neq u$ , let  $X_{u,w}$  be the r.v. indicating the event that the edge  $\{u, w\}$  appears in the random graph, so the degree of  $u$  is  $\deg(u) = \sum_{w \neq u} X_{u,w}$ . Considering the  $(V - 1)$  variables  $X_{u,w}$  as  $(V - 1)$  independent Bernoulli trials, each w.p. of success  $\frac{1}{2}$ , the Chernoff Bound implies that

$$\Pr[\deg(u) > \frac{1}{2}(V - 1)(1 + \delta)] < e^{-2\delta^2(V-1)}, \text{ and}$$

$$\Pr[\deg(u) < \frac{1}{2}(V - 1)(1 - \delta)] < e^{-2\delta^2(V-1)}.$$

Consequently a simple calculation yields (for  $V > 3, \delta > \frac{7}{V-1}$ ),

$$\Pr[\deg(u) \neq \frac{1}{2}V(1 \pm \delta)] < 2e^{-\delta^2 V}.$$

Finally, a union bound on all vertices  $u$  gives

$$\epsilon_{V,\delta} \stackrel{\text{def}}{=} \Pr[\exists u \text{ s.t. } \deg(u) \neq \frac{1}{2}V(1 \pm \delta)] < 2Ve^{-2\delta^2 V}.$$

The last inequality implies that for  $\delta = 2(\sqrt{\frac{\log V}{V}})$ , we obtain  $\epsilon_{V,\delta} < \frac{1}{V}$ , and that for any fixed  $\delta$  we obtain  $\epsilon_{V,\delta} < 2^{-(\delta^2 - \frac{2 \log V}{V})V} = 2^{-\Omega(V)}$ .

■ (Lemma A.2).

### A.3 Matchings in Random Graphs

**Lemma A.3 Matching in Random Bipartite Graphs** - *A random bipartite graph on  $V \times V$  vertices has a perfect matching w.p.  $1 - 2^{-\Omega(V)}$ .*

**Proof** - Fixing 2 disjoint sets  $M, W$  each of size  $V$ , we need to prove that as edges between the men  $M$  and women  $W$  are uniformly and independently picked, a perfect matching is obtained with high probability. By Hall's The-

orem we fail to obtain a matching only if for some  $1 \leq k \leq V$  the bad event  $B_k$  that there are  $k$  men connected to **exactly**  $k - 1$  women occurs. Let  $p_k = \Pr(B_k)$ . It clearly suffices to upper-bound  $\sum_{k=1}^V p_k$ .

First, by lemma A.2, w.p.  $1 - 2^{-\Omega(V)}$  any single man is connected to more than  $\frac{V}{3}$  women. Hence  $\sum_{k=1}^{\frac{V}{3}} p_k < 2^{-\Omega(V)}$ .

Next, consider a specific event  $B_k$ , and note that there are precisely  $\binom{V}{k} \binom{V}{k-1}$  ways of choosing the  $k$  men and the  $k - 1$  women these men know. Also note that since all  $k \times (V - (k - 1))$  edges connecting these  $k$  men to the other women are missing, then  $p_k \leq \binom{V}{k} \times \binom{V}{k-1} \times \frac{1}{2}^{k \times (V - (k - 1))}$ .

Consequently, for  $\frac{1}{3}V \leq k \leq V - \sqrt{V}$ ,  
 $p_k < 2^V \times 2^V \times \frac{1}{2}^{\frac{1}{3}V \times \sqrt{V}} = 2^{-\Omega(V^{1.5})}$ ,

and for  $k > V - \sqrt{V}$ ,  
 $p_k < V^{(V-k)} \times V^{(V-k+1)} \times \frac{1}{2}^{V-\sqrt{V}} \leq V^{(\sqrt{V})} \times V^{(\sqrt{V})} \times \frac{1}{2}^{V-\sqrt{V}} = 2^{-\Omega(V)}$ .

Finally, summing on all  $O(V)$  values of  $k$  we get the desired bound  $\sum_{k=1}^V p_k < 2^{-\Omega(V)}$ .

■ (Lemma A.3).

An immediate corollary of lemma A.3 is:

**Lemma A.4 Perfect Matchings in Random Graphs** - *A random graph on  $V$  vertices has a perfect matching w.p.  $1 - 2^{-\Omega(V)}$ .*

**Proof** - Fix two arbitrary disjoint sets of vertices  $M, W$ , each of size  $\lfloor \frac{V}{2} \rfloor$ , and apply lemma A.3 on the random bipartite graph between  $M$  and  $W$ .

■ (Lemma A.4).

## A.4 High Connectivity of Random Graphs

**Lemma A.5 Structured High Connectivity of Random Graphs** - *Let  $\delta_V \stackrel{\text{def}}{=} 2\sqrt{\frac{\log V}{V}}$ . In a random graph on  $V$  vertices it holds w.p.  $1 - V^{-\Omega(1)}$  that all pairs of vertices  $\{u, w\}$  have  $\frac{1}{2}V(1 \pm \delta_V)$  disjoint paths of which*

- $\frac{1}{4}V(1 - \delta_V)$  paths are of length 2.
- $\frac{1}{4}V(1 - \delta_V)$  paths are of length 3, and are given by a matching between  $\Gamma(u) \setminus \Gamma(w)$ , and  $\Gamma(w) \setminus \Gamma(u)$ .

*Such vertex-pairs are said to be **sufficiently connected by short paths**.*

**Proof** - We prove that an arbitrary pair of vertices  $\{u, w\}$  is sufficiently connected by short paths, and complete the proof by union bounding over all possible pairs.

Fixing a pair  $\{u, w\}$  we set  $D = \Gamma(u) \cap \Gamma(w)$ ,  $M = \Gamma(u) \setminus \Gamma(w)$ , and  $W = \Gamma(w) \setminus \Gamma(u)$ .

We start by proving that with overwhelming probability  $|D|, |M|, |W| \geq \frac{1}{4}V(1 - \delta_V)$ . For some fixed vertex  $v$ , let  $X_v$  be the r.v. indicating the event that  $v \in D$ , so  $|D| = \sum_{v \neq u, w} X_v$ . Considering the  $(V - 2)$  variables  $X_v$  as  $(V - 2)$  independent Bernoulli trials, each w.p. of success  $\frac{1}{4}$ , the Chernoff Bound implies the first inequality,

$$\Pr[|D| < \frac{1}{4}(V - 2)(1 - \delta)] < e^{-4\delta^2(V-2)} < e^{-\delta^2 V}$$

(The second inequality holds for any  $V > 3, \delta > \frac{7}{V-1}$ ).

Next, let  $B_D, B_M, B_W$  be the bad events that  $|D|, |M|$ , or  $|W|$  resp. are smaller than  $\frac{1}{4}V(1 - \delta)$ . Clearly, by symmetry  $\Pr(B_D) = \Pr(B_M) = \Pr(B_W)$ .

Therefore a union bound on these 3 events yields:

$$\Pr[\text{Either } B_D, B_M \text{ or } B_W \text{ occurs for } \{u, w\}] < 3e^{-\delta^2 V}.$$

We next prove that with overwhelming probability a  $\frac{1}{4}V(1 - \delta_V)$  matching between  $|M|$  and  $|W|$  exists, conditioned upon the good event  $E$  that  $|M|, |W| \geq \frac{1}{4}V(1 - \delta_V)$ . To this end assume w.l.o.g. that  $|M| = |W| = V' = \frac{1}{4}V(1 - \delta_V)$ . Note that conditioning upon the event  $E$ , the desired matching is a perfect matching in the random bipartite graph between  $|M|$  and  $|W|$ . Lemma A.3 implies that the probability of failing to attain this perfect matching is  $2^{-\Omega(V')} = 2^{-\Omega(V)}$ .

Consequently, a union bound on the two bad events of either failing to have a large matching given that  $|M|, |W|$  are large, or failing to have  $|D|, |M|$  and  $|W|$  large enough we get,

$$\Pr[\{u, w\} \text{ is not sufficiently connected by short paths}] <$$

$$3e^{-\delta^2 V} + 2^{-\Omega(V)} = \frac{1}{V^3} + 2^{-\Omega(V)}$$

(The last inequality holding for any  $\delta \geq 2(\sqrt{\frac{\log V}{V}})$ ).

Finally, a union bound on the  $\binom{V}{2}$  possible vertex-pairs gives,

$$\Pr[\exists \text{ a pair } \{u, w\} \text{ not sufficiently connected by short paths}] < V^{-\Omega(1)}.$$

■ (Lemma A.5).

**Lemma A.6 Connectivity Number of Random Graphs** - *Let  $\delta_V = 2\sqrt{\frac{\log V}{V}}$ . Then a random graph on  $V$  vertices has connectivity number  $\kappa = \frac{1}{2}V(1 \pm \delta_V)$  w.p.  $1 - V^{-\Omega(1)}$ ,*

**Proof** - Since the maximal degree of a specific graph clearly upper bounds  $\kappa$ , the upper bound on  $\kappa$  is an immediate corollary of lemma A.2. The lower bound is an immediate corollary of lemma A.5. ■ (Lemma A.6).

# Bibliography

- [AS] N. Alon and J.H. Spencer. The Probabilistic Method. *John Wiley and Sons, Inc., 1992.*
- [BM] M. Blum and S. Micali. How to Generate Cryptographically Strong Sequences of Pseudo-Random Bits. *In SIAM Journal on Computing, Vol. 31 pages 850-864, 1984.*
- [B1] B. Bollobás. Random Graphs. *Academic Press, 1985.*
- [B2] B. Bollobás. The Chromatic Number of Random Graphs. *In Combinatorica 8 pages 49-55, 1988.*
- [F] R. Fagin. Probabilities in Finite Models. *In J. Symbolic Logic 41: pages 50-58 1969.*
- [GKLT] Y.V.Glebskii, D.I.Kogan, M.I.Liagonkii, V.A.Talanov. Range and degree of realizability of formulas the restricted predicate calculus. *In Cybernetics 5: pages 142-154 1976.*
- [GGM] O. Goldreich, S. Goldwasser and S. Micali. How to Construct Random Functions. *In Journal of the ACM, Vol. 33, No. 4, pages 276-288,1985.*

- [GGN] O. Goldreich, S. Goldwasser and A. Nussboim. On the Implementation of Huge Random Objects. *In 44'th IEEE Symposium on Foundations of Computer Science, pages 68-79, 2003.*
- [HILL] J. Håstad, R. Impagliazzo, L.A. Levin, M. Luby. A Pseudo-Random Generator from any One-Way Function. *In SIAM Journal on Computing, Vol. 28, Num. 4, pages 1364-1396, 1999.*
- [IW] R. Impagliazzo, A. Wigderson.  $P=BPP$  unless  $E$  has sub-exponential circuits - Derandomizing the XOR-Lemma. *In Proceedings of the 29'th annual ACM symposium on Theory of computing, pages: 220-229, 1997.*
- [J] A. Joffe. On a Set of Almost Deterministic  $k$ -wise Independent Random Variables. *In Annual of Probability 2, pages 1961-1962, 1974.*
- [L] L. Levin. One-Way functions and pseudo-random generators. *In Proceedings of the 17'th ACM Symposium on Theory of Computing, pages 413-420, 1984.*
- [LR] M. Luby and C. Rackoff. How to Construct Pseudo-Random Permutations from Pseudo-Random Functions. *In SIAM Journal on Computing, Vol. 17, pages 373-386, 1998.*
- [NR] M. Naor and O. Reingold. Constructing Pseudo-Random Permutations with a Prescribed Cyclic Structure. *In Journal of Crypto. Vol. 15(2), pages 97-102, 2002.*

- [S] A. Shamir. On the Generation of Cryptographically Strong Pseudo-Random Sequences. *In 8'th ICALP, Lecture notes in Comp. Sci. 62. Springer-Verlag, pages 544-550, 1981.*
- [Y] A.C. Yao. Theory and Application of Trapdoor Functions. *In 23'rd IEEE Symposium on Foundations of Computer Science, pages 80-91, 1982.*