

Very fast construction of bounded-degree spanning graphs via the semi-random graph process*

Omri Ben-Eliezer[†] Lior Gishboliner[‡] Dan Hefetz[§] Michael Krivelevich[¶]

Abstract

Semi-random processes involve an adaptive decision-maker, whose goal is to achieve some predetermined objective in an online randomized environment. They have algorithmic implications in various areas of computer science, as well as connections to biological processes involving decision making. In this paper, we consider a recently proposed semi-random graph process, defined as follows: we start with an empty graph on n vertices, and in each round, the decision-maker, called **Builder**, receives a uniformly random vertex v , and must immediately (in an online manner) choose another vertex u , adding the edge $\{u, v\}$ to the graph. **Builder's** end goal is to make the constructed graph satisfy some predetermined monotone graph property. There are also natural offline and non-adaptive modifications of this setting. We consider the property \mathcal{P}_H of containing a spanning graph H as a subgraph. It was asked by N. Alon whether for every bounded-degree H , **Builder** can construct a graph satisfying \mathcal{P}_H with high probability in $O(n)$ rounds. We answer this question positively in a strong sense, showing that any graph with maximum degree Δ can be constructed with high probability in $(3\Delta/2 + o(\Delta))n$ rounds, where the $o(\Delta)$ term tends to zero as $\Delta \rightarrow \infty$. This is tight (even for the offline case) up to a multiplicative factor of $3 + o_\Delta(1)$. Furthermore, for the special case where H is a forest of maximum degree Δ , we show that H can be constructed with high probability in $O(\log \Delta)n$ rounds. This is tight up to a multiplicative constant, even for the offline setting. Finally, we show a separation between *adaptive* and *non-*

adaptive strategies, proving a lower bound of $\Omega(n\sqrt{\log n})$ on the number of rounds necessary to eliminate all isolated vertices w.h.p. using a non-adaptive strategy. This bound is tight, and in fact there are non-adaptive strategies for constructing a Hamilton cycle or a K_r -factor, which are successful w.h.p. within $O(n\sqrt{\log n})$ rounds.

1 Introduction

Recently, the following *semi-random graph process* was proposed by P. Michaeli, and analyzed by Ben-Eliezer, Hefetz, Kronenberg, Parczyk, Shikhelman, and Stojaković [4]. A single adaptive player, called **Builder**, starts with an empty graph G on a set V of n vertices. The process then proceeds in *rounds*, where in each round **Builder** is offered a uniformly random vertex v , and chooses an edge of the form $\{v, u\}$ to add to the graph G . **Builder's** objective is typically to construct a graph that satisfies some predetermined monotone graph property; for example, to make G an expander with certain parameters, or to have G contain a Hamilton cycle. The natural question arising in this context is the following:

*Given a monotone graph property \mathcal{P} , how many rounds of the semi-random graph process are required for **Builder** to construct (with high probability¹) a graph which satisfies \mathcal{P} ?*

Semi-random problems of this type, involving both randomness and intelligent choices made by a “decision-maker”, have been widely studied in the algorithmic literature. One of the first (and most famous) results on such processes, established by Azar et al. [2], concerns sequential allocation of n balls into n bins, where the goal is to minimize the number of balls in the fullest bin. It is well-known that if each ball is simply assigned to a bin uniformly at random, then w.h.p., the fullest bin will contain $\Theta(\ln n / \ln \ln n)$ balls at the end of the process. However, as was shown in [2], very limited “intelligent intervention” substantially improves the above bound: if, instead of the random assignment, for any ball we are given *two (random) choices* of bins to pick from,

¹With probability that tends to one as $n \rightarrow \infty$; abbreviated w.h.p. henceforth.

*Full version [3] in <https://arxiv.org/abs/1907.08031>.

[†]Blavatnik School of Computer Science, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, 6997801, Israel. Email: omrib@mail.tau.ac.il.

[‡]School of Mathematical Sciences, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, 6997801, Israel. Email: liorgis1@mail.tau.ac.il. Supported in part by ERC Starting Grant 633509.

[§]Department of Computer Science, Ariel University, Ariel 40700, Israel. Email: danhe@ariel.ac.il. Research supported by ISF grant 822/18.

[¶]School of Mathematical Sciences, Raymond and Beverly Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, 6997801, Israel. Email: krivelev@tauex.tau.ac.il. Partially supported by USA-Israel BSF grants 2014361 and 2018267, and by ISF grant 1261/17.

then the trivial strategy of always choosing the least loaded bin out of the two offered, results w.h.p. in the maximum bin load dropping to $\Theta(\ln \ln n)$ – an exponential improvement. This idea has inspired many subsequent theoretical and practical results in various contexts within computer science, see e.g. [9, 25, 28] for a small sample of these.

In a sense, semi-random processes can be viewed as settings where an *online algorithm* aims to achieve a predetermined objective in a *randomized environment*. As opposed to the “standard” setting where online algorithms are measured in terms of their worst-case performance, in the semi-random setting the task is to design online algorithms that achieve their goal with high (or at least constant) probability, and require as few rounds as possible. Further discussion of related semi-random graph models, such as the so-called Achlioptas model, can be found in Section 1.2.1.

Aside from their role within the algorithmic literature, semi-random processes seem to be somewhat related to *decision making in nature*, and in particular to *memory formation in the brain*, an intriguing connection that has not yet been thoroughly investigated, to the best of our knowledge. See Section 1.2.2 for more details.

In this paper we continue the investigation into the semi-random graph process. The first work on this topic [4] proved upper and lower bounds on the number of rounds required to w.h.p. satisfy various properties of interest. Among the upper bounds were an $O(n^{1-\varepsilon})$ bound for the property of containing a copy of any fixed graph H (here ε depends on H), an $O(n)$ bound for containing a perfect matching or a Hamilton cycle, and an $O(\Delta n)$ bound for the property of having minimum degree Δ , as well as for the property of Δ -vertex-connectivity. These results prompted N. Alon to ask whether it is the case that every given (spanning) graph of bounded maximum degree can be constructed w.h.p. in $O(n)$ rounds in this model. Formally, define \mathcal{P}_H as the property of containing an (unlabeled) copy of H , i.e., as the property that there exists an injection $\varphi: V(H) \rightarrow V(G)$ (where G is the graph constructed by Builder), so that $\{\varphi(i), \varphi(j)\} \in E(G)$ for every $\{i, j\} \in E(H)$.

QUESTION 1.1. (N. ALON, QUESTION 6.2 IN [4])
Consider the semi-random graph process over n vertices. Is it true that for every graph H on n vertices with bounded maximum degree, there exists a strategy which enables Builder to w.h.p. construct a copy of H in $O(n)$ rounds?

The main result of this paper gives a positive answer to the above question (see Theorem 1.3).

The offline setting Before addressing Question 1.1, let us consider the easier *offline* setting, where Builder is provided *in advance* with the full sequence of vertices offered throughout the process. In other words, in this setting Builder does not need to make his decisions online, but can rather choose all edges at once after seeing the sequence of random vertices.

As an example, consider the case where Builder’s goal is to construct a triangle-factor². Observe that if at some point of the process, at least $\frac{2n}{3}$ different vertices have been offered, among which at least $\frac{n}{3}$ vertices have been offered at least twice, then Builder can already construct a triangle-factor (in the offline setting). Indeed, Builder simply partitions the vertices into triples $\{u, v, w\}$, where u was offered at least twice and v was offered at least once, and chooses the edges $\{u, v\}$ and $\{u, w\}$ at rounds when u was offered, and the edge $\{v, w\}$ at a round when v was offered. It is easy to check that $O(n)$ rounds suffice w.h.p. to have at least $\frac{2n}{3}$ different vertices offered, and moreover to have at least $\frac{n}{3}$ vertices offered at least twice. Thus, in the offline setting Builder can construct a triangle-factor in $O(n)$ rounds. This $O(n)$ bound can in fact be generalized in a strong sense to *any* bounded-degree target graph H . Proposition 4.1 in [4] – which presents necessary and sufficient general winning conditions for Builder in the offline setting – implies that Builder wins the game as soon as the list of offered vertices allows the construction of a suitable *orientation* of H . In Section 3.1 we show how this can be used to get the following general offline result.

PROPOSITION 1.2. *Let Δ, n be positive integers, and let H be an n -vertex graph of maximum degree Δ . In the offline version of the semi-random process on n vertices, Builder has a strategy allowing him to construct a copy of H in $(\Delta/2 + o(\Delta))n$ rounds w.h.p.*

The $o(\Delta)$ term here is a function of Δ satisfying $o(\Delta)/\Delta \rightarrow 0$ as $\Delta \rightarrow \infty$. Proposition 1.2 substantially extends Theorem 1.9 in [4], which showed a similar result for the property of having minimum degree at least k (with an explicit dependence on k). Proposition 1.2 is clearly optimal up to the $o(\Delta)$ term, since Δ -regular graphs on n vertices have exactly $\Delta n/2$ edges (and hence trivially require at least this number of rounds). More interestingly, it turns out that the $o(\Delta)$ -term is unavoidable; it follows from [4, Theorem 1.9] that at least $(1/2 + \varepsilon_\Delta)\Delta n$ rounds are required for Builder to construct a graph of minimum degree at least Δ , where $\varepsilon_\Delta > 0$ (and $\varepsilon_\Delta \rightarrow 0$ as $\Delta \rightarrow \infty$).

²For a graph F and an integer n divisible by $|V(F)|$, the n -vertex F -factor is the graph which consists of $n/|V(F)|$ vertex-disjoint copies of F .

Main result: online strategy for constructing bounded-degree spanning graphs We now return to the more challenging *online* setting, where **Builder** is offered vertices one-by-one and must (irrevocably) decide which edge to add immediately after being offered a vertex. Our main result in this paper, Theorem 1.3, asserts that **Builder** can construct any given bounded-degree spanning graph in $O(n)$ rounds w.h.p.

THEOREM 1.3. *Let Δ, n be positive integers, and let H be an n -vertex graph of maximum degree Δ . In the online version of the semi-random graph process on n vertices, **Builder** has a strategy guaranteeing that w.h.p., after*

$$\begin{cases} (\Delta/2 + o(\Delta))n, & \text{if } \Delta = \omega(\log n) \\ (3\Delta/2 + o(\Delta))n, & \text{otherwise} \end{cases}$$

rounds of the process, the constructed graph will contain a copy of H .

As before, the $o(\Delta)$ term here is a function of Δ satisfying $o(\Delta)/\Delta \rightarrow 0$ as $\Delta \rightarrow \infty$. Note that Δ is allowed to depend on n arbitrarily. Theorem 1.3 answers Question 1.1 in a strong sense: not only can any bounded-degree graph be constructed w.h.p. in a linear number of rounds, but in fact, the dependence on the maximum degree is very modest. This result clearly illustrates the power of semi-random algorithms compared to their truly random counterparts; see the discussion below on the appearance of various spanning structures in the random graph process.

The notion of *competitive ratio* [7] refers to the performance of an online algorithm compared to the best offline algorithm for the same problem. In view of the trivial $\Delta n/2$ lower bound and Proposition 1.2, our algorithm is $(3 + o_\Delta(1))$ -competitive for general H , where the $o_\Delta(1)$ term tends to zero as Δ tends to infinity. As an open question, it will be very interesting to determine the optimal competitive ratio of an online algorithm for this problem.

PROBLEM 1.4. *In the online version of the semi-random process on n vertices, is it true that for every n -vertex graph H of maximum degree Δ , **Builder** has a strategy which constructs a copy of H w.h.p. in $(\Delta/2 + o(\Delta))n$ rounds?*

We end this discussion with two additional problems, which ask for tight bounds on the number of rounds required to construct some specific graphs H of particular interest.

PROBLEM 1.5. (CONSTRUCTING A HAMILTON CYCLE) *Is there a number α_{Ham} such that in the online version of the semi-random process on n vertices,*

***Builder** can w.h.p. construct a Hamilton cycle in $(\alpha_{\text{Ham}} + o(1))n$ rounds, but w.h.p. cannot accomplish this in $(\alpha_{\text{Ham}} - o(1))n$ rounds? If so, what is the value of α_{Ham} ?*

PROBLEM 1.6. (CONSTRUCTING A K_r -FACTOR) *For each $r \geq 2$, is there a number α_r such that in the online version of the semi-random process on n vertices, for n which is divisible by r , **Builder** can w.h.p. construct a K_r -factor in $(\alpha_r + o(1))n$ rounds, but w.h.p. cannot accomplish this in $(\alpha_r - o(1))n$ rounds? If so, what is the value of α_r ?*

We note that Problem 1.6 is open even for $r = 2$, in which case **Builder's** goal is to construct a perfect matching. Both the perfect matching problem and the Hamilton cycle problem were already considered in [4], where some bounds for these problems were obtained.

Constructing a bounded-degree spanning forest Theorem 1.3 establishes that the (typical) number of rounds needed to construct a general spanning graph H of maximum degree Δ is $O(\Delta n)$. This is clearly tight for graphs whose average degree is $\Theta(\Delta)$. It is now natural to ask if we can break the $\Theta(\Delta n)$ barrier for graphs with a much smaller average degree, such as trees. The next result answers this question positively.

THEOREM 1.7. *Let Δ, n be positive integers and let T be an n -vertex forest of maximum degree Δ . In the online version of the semi-random process on n vertices, **Builder** has a strategy guaranteeing that w.h.p., after $O(\log \Delta)n$ rounds of the process, the constructed graph will contain a copy of T .*

The next proposition shows that the dependence on n and Δ in Theorem 1.7 is tight even for the offline version of the semi-random process.

PROPOSITION 1.8. *For every $\Delta \geq 1$ and $n \geq n_0(\Delta)$, there exists a forest T with n vertices and maximum degree Δ satisfying the following. In the offline version of the semi-random graph process, w.h.p. **Builder** needs $\Omega(\log \Delta)n$ rounds in order to construct a copy of T .*

Non-adaptive strategies It is fairly natural to inquire whether imposing the restriction of *non-adaptivity* handicaps **Builder**, and if so, to which extent exactly. Here, by non-adaptivity we mean that, in a sense, **Builder's** choices are decided upon beforehand, and do not depend on the situation at any given round of the process. The precise definition that we use is as follows. A *non-adaptive strategy* consists of a family \mathcal{L} of adjacency lists $\mathcal{L} = \{L^w : w \in [n]\}$, where for each $w \in [n]$, the list $L^w = (L^w(i) : i = 1, \dots, n-1)$ is a permutation of $[n] \setminus \{w\}$. The lists $\mathcal{L} = \{L^w : w \in [n]\}$

are specified *in advance* (i.e., before the sequence of random vertices starts being exposed). Playing according to such a strategy means that during the vertex exposure process w_1, w_2, \dots , if in a given round vertex w appears for the i th time, $i \geq 1$, then Builder is obliged to connect w to the i th vertex $L^w(i)$ on its list. (To avoid ambiguities, let us assume that if w has already been connected to $L^w(i)$, then Builder simply skips his move. This assumption will not change much in our analysis.)

It turns out that the non-adaptivity assumption indeed hampers Builder — it takes him typically $\Omega(n\sqrt{\log n})$ rounds to get rid of isolated vertices, as stated in Theorem 1.9 below. This is in rather sharp contrast with the situation for general (i.e. adaptive) strategies. Indeed, it is easy to see that Builder can construct a connected graph in $n - 1$ rounds (with probability 1); constructing a graph with no isolated vertices can be done w.h.p. even faster, in $(\ln 2 + o(1))n$ rounds (see [4]); and finally, Theorem 1.3 shows that in fact every bounded-degree graph can be constructed in $O(n)$ rounds. The three theorems below are for the online version of the semi-random graph process.

THEOREM 1.9. *In the semi-random process on n vertices, any non-adaptive strategy requires w.h.p. $\Omega(n\sqrt{\log n})$ rounds to construct a graph in which none of the n vertices is isolated.*

It turns out that the lower bound of Theorem 1.9 is tight in a strong sense: by executing an appropriate non-adaptive strategy for $O(n\sqrt{\log n})$ rounds, Builder can construct w.h.p. several important spanning structures, such as a Hamilton cycle and a K_r -factor (for fixed r). This is expressed in the following two theorems.

THEOREM 1.10. *In the semi-random process on n vertices, there is a non-adaptive strategy allowing Builder to construct a Hamilton cycle in $8n\sqrt{\log n}$ rounds w.h.p.*

THEOREM 1.11. *For every $r \geq 2$ there is $C = C(r)$ such that for every n which is divisible by r , there is a non-adaptive strategy for the semi-random process on n vertices, allowing Builder to construct a K_r -factor in $Cn\sqrt{\log n}$ rounds w.h.p.*

Due to space constraints, the full proofs of the last three theorems are omitted here, and can be found in the full version of this paper [3].

We remark that while the above definition of non-adaptivity is deterministic in nature (in the sense that the lists $\{L^w : w \in [n]\}$ are predetermined), the proof of Theorem 1.9 can be easily adapted (with the same asymptotic lower bound) to apply also to “random non-adaptive strategies”, i.e. strategies in which for every

$w \in [n]$ and $i \geq 1$, the vertex which Builder connects to w at the i th time that w is sampled is drawn from some predetermined probability distribution on $[n] \setminus \{w\}$.

The situation in the (“purely”)-random graph process A common theme in our results is that introducing “intelligent choices” into a random setting can allow for a dramatic improvement. (Perhaps the first appearance of this theme is in the aforementioned work of Azar et al. [2].) To illustrate this phenomenon in the setting of random graph processes, let us compare our Theorem 1.3 with the situation in which all edge-choices are made completely randomly. Recall that the *random graph process* $\tilde{G} = (G_m)_{m=0}^N$ is defined by choosing a random permutation e_1, \dots, e_N of all $N = \binom{n}{2}$ edges of K_n , and letting G_m be the graph whose edge-set is $\{e_1, \dots, e_m\}$. We refer the reader to [10, 13] for an overview of this classical object, as well as the standard random graph models $G(n, m)$ and $G(n, p)$, which are mentioned below. Note that for each $0 \leq m \leq \binom{n}{2}$, the graph G_m is distributed as the Erdős-Rényi graph $G(n, m)$, i.e. as a random graph chosen uniformly among all graphs with m edges and n (labeled) vertices.

The appearance of bounded-degree spanning graphs in the random graph process (or equivalently, in $G(n, m)$) has been thoroughly investigated³. For starters, a standard first moment argument (see e.g. [13]) shows that a copy of $K_{\Delta+1}$ only appears in the random graph process after roughly $n^{2-2/\Delta}$ rounds. Thus, obtaining even a single $K_{\Delta+1}$ -copy (let alone a $K_{\Delta+1}$ -factor) requires much more than a linear number of rounds. Determining the typical time of the appearance of a $K_{\Delta+1}$ -factor turned out to be a difficult problem. Following a long line of research, this problem was settled by (a special case of) a celebrated result by Johansson, Kahn and Vu [14], which states that a $K_{\Delta+1}$ -factor appears in the random graph process at around $m = n^{2-2/(\Delta+1)}(\log n)^{1/\binom{\Delta+1}{2}}$. A more general discussion on the appearance of bounded-degree graphs H other than clique-factors (and on the closely-related notion of *universality*) can be found in the recent work of Ferber, Kronenberg, and Luh [8]; see in particular Conjecture 1.5 there.

Similar superlinear lower bounds (on the number of edges required in order to typically contain a $K_{\Delta+1}$ -factor, or even a single $K_{\Delta+1}$) are known or can be shown for various other random graph models, such as

³We note that the results surveyed here were actually proved for the binomial random graph $G(n, p)$, which is the graph obtained by selecting each of the $\binom{n}{2}$ edges of K_n with probability p and independently. It is well known (see e.g. [13, Section 1.4] or [10, Section 1.1]) that $G(n, m)$ is closely related to $G(n, p)$ when $p = m/\binom{n}{2}$, allowing to transfer results between the two models.

the random regular graph, or the model G_{k-out} (where one connects each vertex to exactly k other randomly chosen vertices, discarding repetitions). Thus, while it was shown in [4] that the semi-random graph process can *simulate* the random models $G(n, p)$, $G(n, m)$, and G_{k-out} , the above discussion indicates that these cannot help in solving Question 1.1, and we must utilize the power of the intelligent player, Builder, in a more imaginative way.

A similar comparison can be made between Theorem 1.7 and the emergence of (bounded-degree) spanning trees in the random graph process. It is well known that in the random graph process, w.h.p. the last isolated vertex disappears only at around $m = \frac{1}{2}n \log n$. Thus, a superlinear number of rounds is required in order to contain a spanning tree w.h.p. Again, we observe here that intelligent choices speed up the time required to reach the given goal: it takes $\Theta(n \log n)$ rounds for the random graph process to contain even a single spanning tree, whilst in our semi-random graph process, the number of rounds required to contain a prescribed bounded-degree spanning tree is only $O(n)$.

It is worth mentioning that a recent breakthrough of Montgomery [22], which confirms a conjecture of Kahn [15], shows that for a fixed Δ , w.h.p. all spanning trees with maximum degree Δ appear in G_m after $m = Cn \log n$ rounds (where $C = C(\Delta)$ is a large enough constant). We refer the reader to [22] for further references to many other related works on this subject.

1.1 Our Techniques We next describe the techniques and ideas underlying the proof of Theorem 1.3. We start by describing our main building block, a technique called *role-switching*. To keep matters simple, let us focus for now on the case where Builder’s goal is to construct a triangle-factor. Later on, we shall describe how role-switching can be employed to efficiently construct any spanning graph H of a given maximum degree.

Consider the following simple strategy for constructing a triangle-factor: partition the n vertices *in advance* to triplets, and for each triplet, fix a cyclic orientation of the three edges contained in the triplet; when a vertex is offered, connect it to the vertex in its triplet to which it sends an edge in the orientation. Evidently, if each of the n vertices has been offered, then at the end of the process all triplets induce triangles, as required⁴. Unfortunately, this w.h.p. only happens

⁴Alternatively, Builder can orient some of the triplets transitively (and again play by choosing edges going out of the vertex he is offered). It is evident that for any prescribed orientation, in order for a triplet to span a triangle, it is necessary that each vertex is offered at least as many times as its out-degree in the

after $\Omega(n \log n)$ rounds. Still, all is not lost; upon closer inspection, one can see that after $C_1 n$ rounds (for a suitable large constant C_1), *almost all* triplets already induce a triangle. Indeed, the probability that a given vertex is not offered in the course of $C_1 n$ rounds of the process is $(1 - 1/n)^{C_1 n} \leq e^{-C_1}$. Thus, after Builder employs the above strategy for the first $C_1 n$ rounds, the number of triplets containing an unoffered vertex is w.h.p. bounded by, say, $2e^{-C_1 n}$; call these triplets *bad*, and note that all other triplets (which are said to be *good*) already induce a triangle.

Call the first $C_1 n$ rounds of the process *the non-adaptive phase*. At the end of this phase, Builder’s high-level objective changes: instead of waiting for vertices from bad triplets to be offered, in the next $C_2 n$ rounds (where C_2 is again a large enough constant) Builder tries to proactively “fix” these bad triplets with the help of good ones, with what we call a *role-switching technique*. Crucially, this technique does not require vertices in bad triplets to be offered at all, instead carefully utilizing the fact that the large majority of vertices at this point belong to good triplets.

We now give a toy example of the role-switching technique. This example, while very far from the actual algorithm, illustrates some key points in our approach. The basic idea is as follows. Suppose that $T = \{u, v, w\}$ is a *bad* triplet and that $T' = \{u', v', w'\}$ and $T'' = \{u'', v'', w''\}$ are *good* triplets, inducing triangles. We describe a mechanism to repartition these nine vertices into three disjoint triplets inducing triangles, without ever being offered any of the bad vertices u, v, w . Specifically, we aim to “switch roles” between u and u' , and between v and v'' . Suppose that during the current batch of $C_2 n$ rounds, each of the vertices $u', v', w', u'', v'', w''$ is offered at least once, and v'' is offered at least twice. For the purpose of this discussion, let us also assume that v' and w' are offered before u' is, and u', u'' and w'' are offered before v'' is. Our strategy is as follows:

- When v' and w' are offered, we connect them to u . This makes $\{u, v', w'\}$ a triangle.
- The first time that u' is offered, we connect it to w . At this current point, we say that u' completed the role-switch with u . What we mean here is that at this point, u can assume the role of u' (as it is already connected to both v' and w'), and u' can assume the role of u and try to form a triangle with v and w . So from now on we put aside the triplet $\{u, v', w'\}$ (which “took care” of u by including it in

orientation. (For the transitive orientation, we need a specific vertex to be offered at least twice, and another specific vertex to be offered at least once.)

a triangle), and focus on the remaining two triplets $\{u', v, w\}, \{u'', v'', w''\}$. If before the goal of v'' was to connect to u and w (in order to switch roles with v), its new goal is to connect to u' and w (since u and u' switched roles).

- When u'' and w'' are offered we connect them to v , thus making $\{u'', w'', v\}$ a triangle.
- The first time v'' is offered we connect it to w , and the second time v'' is offered we connect it to u' . This makes $\{u', v'', w\}$ a triangle. We now say that v'' completed a role-switch with v . And indeed, v can assume the role of v'' as it forms a triangle with u'' and w'' , and v'' can assume the role of v as it forms a triangle with u' and w .

After all of these connections are made, the disjoint triplets $\{u', v'', w\}, \{u, v', w'\},$ and $\{u'', v, w''\}$, all induce triangles (and are thus good), as desired. Observe that under the above assumption (regarding the order of appearances of the vertices $u', v', w', u'', v'', w''$), vertices in T' are only connected (when offered) to vertices in T , while vertices in T'' are connected to both T and T' when they are offered.

Observe that the same strategy can be successfully applied for any possible order of appearances of the vertices $u', v', w', u'', v'', w''$. If, for example, u' and v' are offered before w' is, then w' (and not u') will be switched with one of the vertices in $\{u, v, w\}$. The same holds for the order of appearances of u'', v'', w'' . We will not get into the details here, since the example described above is in any case only a rough illustration of role-switching.

Quantitative aspects of role-switching We now give a rough quantitative explanation of how the role-switching technique can be scaled to fix *all* bad triplets simultaneously in a linear number of rounds. After the first (non-adaptive) phase is complete, the ratio of good triplets to bad ones is w.h.p. of order $\Theta(e^{C_1})$ (in general, the constant terms hiding in $O(\cdot)$ and $\Theta(\cdot)$ expressions here are absolute, and do not depend on C_1 or C_2). Consequently, at the end of the first phase, we can assign to each bad triplet $T = \{u, v, w\}$ a collection $\mathcal{A}_1(T)$ of $\Theta(e^{C_1})$ good triplets.

In the second phase, lasting C_2n rounds, all good triplets in $\mathcal{A}_1(T)$ simultaneously aim to perform role-switching with T . Specifically, we partition $\mathcal{A}_1(T)$ into two parts $\mathcal{A}'_1(T)$ and $\mathcal{A}''_1(T)$ of roughly the same size. Suppose that at least one triplet from $\mathcal{A}'_1(T)$ and one triplet from $\mathcal{A}''_1(T)$ have all of their vertices offered at least twice during the second phase. We say that T is *successful* in this case. Indeed, if without loss of generality $T' \in \mathcal{A}'_1(T)$ is the first triplet to have all

three vertices offered at least once, then at this point the first two bullets of the role-switching strategy are completed, and a vertex from T' switches roles with u . Now, the first triplet $T'' \in \mathcal{A}''_1(T)$ to have two of its vertices u'', w'' offered, and afterwards the third vertex v'' offered twice, will complete a role-switch between v'' and v (last two bullets) and ensure that the union of these triplets forms three disjoint triangles, as desired.

Straightforward calculations show that for a large enough C_2 , the probability that any given bad triplet T remains bad at the end of the phase is $O(e^{-e^{C_1}})$. By using standard concentration inequalities, one can show that the total number of bad triplets at the end of this phase is w.h.p. only $O(e^{-e^{C_1}}n)$. In comparison, the number of bad triplets *before* this phase was of order $\Theta(e^{-C_1}n)$. So we see that by following the role-switching strategy for C_2n rounds we achieve an exponential improvement (i.e., reduction) in the number of bad triplets.

The remaining (at most $O(e^{-e^{C_1}}n)$) bad triplets are taken care of in the third and final phase, which is very similar to the second one, but only requires $o(n)$ rounds. Here we rely on the fact that the good-to-bad triplet ratio is already very large, which in turn substantially decreases the number of rounds needed for role-switching to occur w.h.p. The analysis, which is relatively straightforward yet slightly tedious (and thus omitted from the introduction), is based on an amplification argument similar to the one used in the analysis of the second phase.

From triangle-factor to general H The strategy presented for the case where H is a triangle-factor is a special case of our strategy for general $H = (V_H, E_H)$ of maximum degree Δ . Like above, the strategy runs in three phases, where the first phase is non-adaptive, and the two later phases employ role-switching. In the beginning of the *non-adaptive* phase, Builder fixes an (arbitrary) bijection $\varphi : V(H) \rightarrow [n]$. The meaning of this is that each vertex v of Builder's empty graph $G = ([n], \emptyset)$ is assigned a "role" $\varphi^{-1}(v) \in V_H$ to be played by v in the copy of H that Builder is constructing. In other words, Builder intends to make the bijection φ an embedding of H into his graph (or at least to embed "almost all" of H in this manner). Classical graph-theoretic arguments assert that there is an orientation O of the edges of H so that the out-degree of any vertex is at most $\Delta/2 + 1$. Leveraging this, we employ the following non-adaptive strategy for $(\Delta/2 + o(\Delta))n$ rounds: in each round where $\varphi(x) \in [n]$ is offered (for $x \in V_H$), we pick some $y \in V_H$ to which there exists an edge from x in O , and connect $\varphi(x)$ to $\varphi(y)$; once all such edges are exhausted, we call x *good* (with respect to φ). All vertices that are not good in the end of this

phase are considered *bad*. Henceforth, we denote the set of good vertices by A , and its complement (i.e. the set of bad vertices) by B .

After the first phase, the number αn of bad vertices is very small; for an appropriate choice of the $o(\Delta)$ term above, it holds w.h.p. that, say, $\alpha = o(e^{-\Delta^{0.1}})$. All other $(1 - \alpha)n$ vertices are good with respect to φ . Crucially, the following desirable invariant holds for the set A of good vertices: $\varphi|_A$ is an *isomorphism* from the subgraph of H induced on A to a subgraph of Builder’s graph G . To complete the construction of H , from now on we use role-switching, as defined below, to repeatedly transform bad vertices into good ones, all the while maintaining the above desirable invariant. Our goal is that in the end of the process, all vertices will be good, and thus (by the invariant) G will contain a copy of H as a subgraph.

Formally, for any $x \in V(H)$ we denote the set of “intended” neighbours of $\varphi(x)$ in $[n]$ by $N_G(x) := \{\varphi(z) : z \in V_H, \{x, z\} \in E_H\}$. A *role-switch* is an operation carried between a bad vertex $y \in B$ and a good vertex $x \in A$ whose neighbours in H are also good.

- In the first part of an attempted switch, Builder connects the intended neighbours of $\varphi(x)$ to $\varphi(y)$; that is, each time a vertex from $N_G(x)$ is offered, he connects it to $\varphi(y)$.
- In the second part, Builder connects $\varphi(x)$ to the intended neighbours of y : each time $\varphi(x)$ is offered, Builder connects it to some good vertex $w \in N_G(y) \cap \varphi(A)$ that is not already connected to $\varphi(x)$.
- As soon as all such vertices w are connected to $\varphi(x)$, we switch between the values of $\varphi(x)$ and $\varphi(y)$, add y to A as a new good vertex, and consider the role-switch completed.

The power of the role-switching method stems from the fact that it makes y good (and maintains the above invariant) without the need to ever be offered y or any of its intended neighbours.

Since the ratio of good to bad vertices at the end of the first phase is exponential in Δ , we can assign to each bad vertex a collection of $e^{\Delta^{\Omega(1)}}$ good vertices x whose neighbours are also good (and all sets $\{\varphi(x)\} \cup N_G(x)$ are pairwise-disjoint). In the second phase, which lasts $(\Delta + o(\Delta))n$ rounds, for any bad vertex y we simultaneously try to role-switch between y and any of the good vertices x assigned to it. The analysis (as in the triangle-factor case) shows that after this phase is over, w.h.p. the good-to-bad ratio substantially increases, becoming doubly exponential in Δ . In the final phase,

lasting $O(n)$ rounds, we utilize the improved good-to-bad ratio and assign a larger collection of good vertices (of size doubly exponential in Δ) to each remaining bad vertex, so that w.h.p. all such vertices become good by the end of the phase. The total number of rounds is $(\frac{\Delta}{2} + o(\Delta))n + (\Delta + o(\Delta))n + O(n) = (\frac{3\Delta}{2} + o(\Delta))n$.

Constructing spanning forests We now describe the main ideas underlying the proofs of Theorem 1.7 and Proposition 1.8, concerning the construction of spanning forests. The full details can be found in Section 4.

The proof of the (online) upper bound of Theorem 1.7 has the same general scheme as the proof of Theorem 1.3; namely, it consists of an initial phase in which Builder embeds almost the entire target forest T , and a “role-switching” phase in which Builder gradually alters this initial embedding to obtain a complete copy of T . Still, there are two crucial differences between the proofs of these two theorems. The first difference concerns the way in which the initial embedding is constructed. In the proof of Theorem 1.7, this is done by first putting aside a set B of $n/\text{poly}(\Delta)$ vertices of degree at most 2, and then greedily constructing a copy of $T' := T \setminus B$ vertex by vertex. It is easy to see that Builder completes a copy of T' as soon as he is offered $|V(T')|$ different vertices, which happens w.h.p. in $O(\log \Delta)n$ rounds. The second difference is that in the role-switching phase, only vertices of degree 2 are used as candidates for role-switching. This allows Builder to complete the embedding faster; w.h.p. he does so in $O(n)$ additional rounds.

For the (offline) lower bound of Proposition 1.8, consider first the case where our forest T is a star on n vertices. We wish to prove a lower bound of $\Omega(n \log n)$ for this case. In order for Builder to be able to construct a copy of T (even in the offline setting), it is necessary that to have at least one of the following conditions: either there should exist a vertex which was offered at least, say, $\log^2 n$ times during the process, or the total number of vertices not offered at all throughout the process should be less than $\log^2 n$. It is straightforward to check that w.h.p., neither of these conditions holds after $cn \log n$ rounds (where $c > 0$ is a small constant). To prove Proposition 1.8 for any value of Δ (and not just for $\Delta = n - 1$), we replace our spanning star with a star-forest consisting of $n/(\Delta + 1)$ stars, each having Δ leaves. A similar analysis shows that constructing this star-forest requires $\Omega(n \log \Delta)$ rounds w.h.p.

1.2 Related Work

1.2.1 Semi-random processes Perhaps the most studied semi-random graph process is now known as

the *Achlioptas process*, and was proposed by Dimitris Achlioptas in 2000. Similarly to our semi-random process, the Achlioptas process is a one-player game in which the player, **Builder**, gradually constructs a graph. The process runs in rounds, where in each round, two uniformly random edges are picked from the set of all $\binom{n}{2}$ edges of the n -vertex complete graph, or alternatively (depending on the version of the process) from all edges untaken at this point. These two edges are offered to **Builder**, who then must choose exactly one of them and add it to the graph.

While in our random graph process **Builder's** goal is always to make his graph *satisfy* some given graph property, in the context of the Achlioptas process the goal is often to *avoid* satisfying a given property for as long as possible. In fact, Achlioptas's original question was whether **Builder** can delay the appearance of a giant component beyond its typical time of appearance in the ("purely"-)random graph process. This question was answered positively by Bohman and Frieze [5], see also [6, 16, 27, 1, 26]. Similar problems have also been studied for other properties or objectives; for example, the problem of avoiding a fixed subgraph [17, 24], or the problem of speeding up the appearance of a Hamilton cycle [18]. Achlioptas-like processes involving two choices were also investigated in other contexts, see e.g. [23] for a geometric perspective.

1.2.2 The neuroscience connection: memory formation in the brain An intriguing natural process that seems somewhat related to our semi-random graph model is that of *memory formation in the human brain*, whose investigation is one of the main frontiers of modern neuroscience. While this process is not yet completely understood, it is already well-known [12, 19, 20] that it shares several characteristics with our semi-random process; crucially, long-term memories are stored in the brain using physical structures that can essentially be viewed as weighted subgraphs, which makes the efficient construction (and destruction) of such graphs desirable for creation or modification of memories.

The memory formation process is typically governed by a centralized decision maker, usually the part of the brain called the *hippocampus*. In this context, *neurons* play the role of vertices, whereas the physical connections between them – known as *synapses* – play the role of edges. It is known that a combination of electric signals ("spikes") and specialized proteins (see [19]) sent from the hippocampus to neurons, controls the long-term storage of memories, and that returning signals from the neurons are also involved in the process.

In any case, while the exact mechanisms of memory

formation and destruction are not yet fully understood, they are probably far more complicated than the semi-random graph process we investigate here. However, there are various random models for communication between neurons, see e.g. the book of Gerstner and Kistler on spiking neuron models [11]. We believe that further study of suitable semi-random processes (with biologically-inspired models for randomness) and their connections to complex natural processes is an interesting venue for future research.

1.3 Paper Organization and Notation In Section 2 we state several auxiliary results. Section 3 contains the proofs of Theorem 1.3 and Proposition 1.2, as well as the description and analysis of Strategy 3.6, which is the key tool used in our proofs. Theorem 1.7 and Proposition 1.8 are proved in Section 4. Due to space limitations, the proofs of Theorems 1.9, 1.10 and 1.11 will only appear in the full version of this paper. Since the statements of Theorems 1.3 and 1.7 are asymptotic (in both n and Δ), we always assume, where needed, that n and Δ are sufficiently large. All logarithms are base e . We omit floor and ceiling signs whenever these are not crucial.

2 Preliminaries

We start by stating three known concentration inequalities that will be used in this paper. The first is a standard Chernoff-type bound (see, e.g., [13]), the second is a simplified version of Azuma's inequality (see, e.g., [13, Theorem 2.27]), and the third is a simplified version of Talagrand's inequality (see, e.g., [21]).

LEMMA 2.1. *Let X be a binomial random variable. Then, for every $\lambda \geq 0$, it holds that*

$$\mathbb{P}[X \leq \mathbb{E}[X] - \lambda] \leq e^{-\frac{\lambda^2}{2\mathbb{E}[X]}}$$

and that

$$\mathbb{P}[X \geq \mathbb{E}[X] + \lambda] \leq e^{-\frac{\lambda^2}{2(\mathbb{E}[X] + \lambda/3)}}.$$

LEMMA 2.2. [13, Theorem 2.27] *Let X be a non-negative random variable, not identically 0, which is determined by T independent trials w_1, \dots, w_T . Suppose that $c \in \mathbb{R}$ is such that changing the outcome of any one of the trials can change the value of X by at most c . Then, for every $\lambda \geq 0$, it holds that*

$$\mathbb{P}[X \leq \mathbb{E}[X] - \lambda] \leq e^{-\frac{\lambda^2}{2c^2T}}$$

and that

$$\mathbb{P}[X \geq \mathbb{E}[X] + \lambda] \leq e^{-\frac{\lambda^2}{2c^2T}}.$$

LEMMA 2.3. [21, Pages 80-81] Let X be a non-negative random variable, not identically 0, which is determined by T independent trials w_1, \dots, w_T . Suppose that $c, g > 0$ are such that

1. Changing the outcome of any one of the trials can change the value of X by at most c .
2. For every s , if $X \geq s$ then there is a set of at most $g \cdot s$ trials whose outcomes certify⁵ that $X \geq s$.

Then, for every $0 \leq \lambda \leq \mathbb{E}[X]$, it holds that

$$\mathbb{P} \left[|X - \mathbb{E}[X]| > \lambda + 60c\sqrt{g\mathbb{E}[X]} \right] \leq 4e^{-\frac{\lambda^2}{8c^2g\mathbb{E}[X]}} .$$

We will also need the following lemma regarding “balanced” orientations of graphs.

LEMMA 2.4. Let H be an n -vertex graph of maximum degree Δ . Then there exists an orientation O of the edges of H which satisfies the following two conditions:

- (a) $d_O^+(u) \leq \lfloor \Delta/2 \rfloor + 1$ for every $u \in V(H)$,
- (b) there exists a set $A \subseteq V(H)$ of size $|A| \geq \frac{n}{\Delta^2+1}$ such that $d_O^+(u) = 0$ for every $u \in A$.

Proof. Let H^2 denote the square of H , that is, the graph obtained from H by adding an edge between any two vertices at distance 2 in H . Let $A \subseteq V(H)$ be a maximum independent set in H^2 ; clearly $|A| \geq \frac{n}{\Delta^2+1}$. Let $H_0 = H \setminus A$. If $d_{H_0}(u)$ is even for every $u \in V(H_0)$, then let $H_1 = H_0$. Otherwise, let H_1 denote the graph obtained from H_0 by adding a new vertex x and connecting it by an edge to every vertex of odd degree in H_0 . Orient the edges of each connected component of H_1 along some Eulerian cycle of that component. Orient every edge of $E_H(V(H) \setminus A, A)$ from $V(H) \setminus A$ to A . Delete x and denote the resulting oriented graph by O . Observe that A is independent in H and thus O is an orientation of all the edges of H . It is evident that O satisfies (b). Moreover, O satisfies (a) as A is independent in H^2 and thus $d_H(u, A) \leq 1$ for every $u \in V(H) \setminus A$. \square

3 Constructing Spanning Graphs of Given Maximum Degree

In this section we prove Theorem 1.3 and Proposition 1.2. The tools we develop here will also be used in the proof of Theorem 1.7 in Section 4. We start

⁵To be precise, this means that if (w_1, \dots, w_T) is such that $X \geq s$, then there is $k \leq g \cdot s$ and $1 \leq i_1 < \dots < i_k \leq T$ such that changing the outcome of any trials other than w_{i_1}, \dots, w_{i_k} does not change the fact that $X \geq s$.

by introducing some definitions and sketching a rough outline of the proof scheme that we will use to prove Theorem 1.3. From this point onward, we fix an n -vertex graph H with maximum degree Δ . We assume that the ground-set of vertices for the semi-random process is $[n]$. For a graph G with vertex-set $[n]$ and a bijection $\varphi : V(H) \rightarrow [n]$, we say that a subset $A \subseteq V(H)$ is (G, φ) -good if φ maps all edges of H contained in A to edges of G ; that is, A is (G, φ) -good if $\{\varphi(x), \varphi(y)\} \in E(G)$ for every $x, y \in A$ such that $\{x, y\} \in E(H)$. The graph G under consideration will always be Builder’s graph at some given moment during the process. It will usually be clear which moment we are considering, and so we will omit G from the notation, simply writing “ φ -good”. Note that, if at some point during the process, there is a bijection $\varphi : V(H) \rightarrow [n]$ such that $V(H)$ is φ -good, then Builder has succeeded in constructing a copy of H .

Our strategy consists of two stages: in the first stage, Builder fixes an arbitrary bijection $\varphi : V(H) \rightarrow [n]$ and plays so as to construct a φ -good set which is as large as possible. We show (see Lemma 3.4 below) that Builder can w.h.p. guarantee the existence of a φ -good set which covers “almost all” of the vertices of H , within $(\Delta/2 + o(\Delta))n$ rounds. This part of the argument is fairly straightforward, and is given in Section 3.1. In the second stage, which is far more involved and constitutes the heart of the proof, Builder tries to iteratively extend this φ -good set by updating the embedding φ . We will show that by using a suitable “role-switching” strategy (i.e., Strategy 3.6), Builder can ensure that w.h.p. $V(H)$ will be φ -good after $(\Delta + o(\Delta))n$ additional rounds. This is done in Lemma 3.13.

3.1 The Initial Embedding, Proposition 1.2, and the Easy Case of Theorem 1.3

In this section we describe and analyze the first stage of Builder’s strategy. Along the way we prove Proposition 1.2, as well as the easy part of Theorem 1.3, which corresponds to the regime $\Delta = \omega(\log n)$. In what follows, we will need the following lemma.

LEMMA 3.1. Let n, d be positive integers, let $\alpha \in (0, 0.1)$, and suppose that $n \gg 1/\alpha$ and $d \gg \log(1/\alpha)$. In the course of $(d + o(d))n$ rounds of the semi-random process on n vertices, the number of vertices $i \in [n]$ which were offered at most d times is w.h.p. less than αn .

Proof. Set $\ell := (d + \sqrt{6d \log(1/\alpha)})n = (d + o(d))n$ (here we use our assumption that $d \gg \log(1/\alpha)$), and let w_1, \dots, w_ℓ be the first ℓ random vertices offered to Builder. Let U be the set of all vertices $i \in [n]$ such that $|\{1 \leq j \leq \ell : w_j = i\}| \leq d$. Evidently,

$|\{1 \leq j \leq \ell : w_j \in U\}| \leq d \cdot |U|$. So in order to prove the lemma, it is enough to show that w.h.p., every $U \subseteq [n]$ of size αn satisfies the inequality $X_U := |\{1 \leq j \leq \ell : w_j \in U\}| > d \cdot |U| = d\alpha n$.

Fix any $U \subseteq [n]$ of size $|U| = \alpha n$. Note that $X = X_U$ has the distribution $\text{Bin}(\ell, \frac{|U|}{n})$, and in particular $\mathbb{E}[X] = \ell|U|/n = \alpha\ell$. By Lemma 2.1 with $\lambda = \alpha\ell - d\alpha n$, we have

$$\begin{aligned} \mathbb{P}[X < d\alpha n] &= \mathbb{P}[X < \mathbb{E}[X] - \lambda] \\ &\leq \exp\left(-\frac{\alpha^2 n^2 \cdot (\ell/n - d)^2}{2\alpha\ell}\right) \\ &\leq \exp\left(-\frac{\alpha n \cdot (\ell/n - d)^2}{2d + o(d)}\right) \\ &= e^{-(3+o_d(1))\alpha n \log(1/\alpha)}. \end{aligned}$$

Now our assertion follows by taking the union bound over all

$$\binom{n}{\alpha n} \leq (e/\alpha)^{\alpha n} = e^{(\log(1/\alpha)+1) \cdot \alpha n} \leq e^{2\alpha \log(1/\alpha)n}$$

sets $U \subseteq [n]$ of size $|U| = \alpha n$, and using the assumption that $n \gg 1/\alpha$. \square

Proof of Proposition 1.2. Since the statement of the proposition is asymptotic in Δ , we may (and will) assume that Δ is large enough, where needed. Moreover, in light of Theorem 1.3, we may assume that $\Delta = O(\log n)$ (because otherwise, Builder can w.h.p. construct a copy of H in $(\Delta/2 + o(\Delta))n$ rounds even in the online version of the semi-random process). Let O be an orientation of the edges of H as in Lemma 2.4. Let w_1, w_2, \dots denote the sequence of random vertices Builder is offered. Let $m(O)$ denote the smallest integer m for which there exists a bijection $\varphi : V(H) \rightarrow [n]$ such that $\varphi(u)$ appears in (w_1, w_2, \dots, w_m) at least $d_O^+(u)$ times for every $u \in V(H)$. It follows by Proposition 4.1 in [4] that the number of rounds needed for Builder to construct a copy of H in the offline version of the semi-random process is at most $m(O)$. Hence, in order to complete the proof of this proposition, it suffices to show that w.h.p., in the course of $m := (\Delta/2 + o(\Delta))n$ rounds, at least $n - n/(\Delta^2 + 1)$ vertices are offered at least $\lfloor \Delta/2 \rfloor + 1$ times each. But this is just the statement of Lemma 3.1 with parameters $d = \lfloor \Delta/2 \rfloor$ and $\alpha = 1/(\Delta^2 + 1)$. \square

In what follows, Builder will employ the following simple (non-adaptive) strategy.

STRATEGY 3.2. Let H be an n -vertex graph of maximum degree Δ . Fix an orientation O of the edges

of H which satisfies property (a) of Lemma 2.4. Let $\varphi : V(H) \rightarrow [n]$ be an arbitrary bijection. At any given round of the process, having been offered a random vertex w , Builder chooses an arbitrary vertex $u \in \varphi(N_O^+(\varphi^{-1}(w)))$ which is not adjacent to w in his current graph, and claims the edge $\{u, w\}$; if no such vertex u exists, then Builder claims an arbitrary edge incident with w .

In the following proposition we prove the easy part of Theorem 1.3.

PROPOSITION 3.3. For every $\varepsilon \in (0, 1)$ there exists an integer C for which the following holds. Let n and $\Delta = \Delta(n) \geq C \log n$ be positive integers and let H be an n -vertex graph of maximum degree Δ . Then, in the online version of the semi-random process on n vertices, Builder has a strategy guaranteeing that w.h.p., after $(1 + \varepsilon)\frac{\Delta n}{2}$ rounds of the process, his graph will contain a copy of H .

Proof. Builder executes Strategy 3.2 for $\ell := (1 + \varepsilon)\frac{\Delta n}{2}$ rounds. In the notation of Strategy 3.2, it is evident that if for every $x \in V(H)$, the vertex $\varphi(x)$ is offered at least $d_O^+(x)$ times, then Builder is successful in building a copy of H . Therefore, in order to complete the proof of the proposition, it suffices to show that w.h.p., for every $1 \leq i \leq n$, the vertex i is offered at least $\lfloor \Delta/2 \rfloor + 1$ times in the course of these ℓ rounds of the process.

Fix any $1 \leq i \leq n$, and let Z_i be the random variable counting the number of times i is offered during the ℓ rounds. Then $Z_i \sim \text{Bin}(\ell, 1/n)$, implying that $\mathbb{E}[Z_i] = \ell/n = (1 + \varepsilon)\Delta/2$. Applying Lemma 2.1 with $\lambda = \varepsilon\Delta/2$, we obtain

$$\begin{aligned} \mathbb{P}[Z \leq \Delta/2] &= \mathbb{P}[Z \leq \mathbb{E}[Z] - \varepsilon\Delta/2] \leq \exp\left(-\frac{(\varepsilon\Delta/2)^2}{(1 + \varepsilon)\Delta}\right) \\ &\leq e^{-\Delta\varepsilon^2/8} \leq n^{-C\varepsilon^2/8} \leq 1/n^2 \end{aligned}$$

where the penultimate inequality holds since $\Delta \geq C \log n$, and the last inequality holds for $C \geq 16\varepsilon^{-2}$. A union bound then implies that with probability at least $1 - \frac{1}{n} = 1 - o(1)$, every $1 \leq i \leq n$ was offered at least $\lfloor \Delta/2 \rfloor + 1$ times, as required. \square

We now return to the main case of Theorem 1.3, assuming henceforth that $\Delta = O(\log n)$. In the following lemma, we analyze the aforementioned first stage of Builder's strategy.

LEMMA 3.4. Let n, Δ be positive integers, let $\alpha \in (0, 0.1)$, and suppose that $n \gg \Delta/\alpha$ and $\Delta \gg \log(1/\alpha)$. Let H be an n -vertex graph of maximum degree Δ and let $\varphi : V(H) \rightarrow [n]$ be a bijection. Then, Builder has a strategy guaranteeing that after $(\Delta/2 + o(\Delta))n$

rounds of the process, w.h.p. there will be a φ -good set $A \subseteq V(H)$ of size at least $(1 - \alpha)n$.

Proof. Builder executes Strategy 3.2 for

$$\ell := (\Delta/2 + \sqrt{3\Delta \cdot \log(2\Delta/\alpha)})n = (\Delta/2 + o(\Delta))n$$

rounds. Let A' be the set of all vertices $x \in V(H)$ such that $\varphi(x)$ was offered at least $\lfloor \Delta/2 \rfloor + 1$ times. Apply Lemma 3.1 with $d = \lfloor \Delta/2 \rfloor$ and with $\frac{\alpha}{2\Delta}$ in place of α , to conclude that w.h.p. we have $|A'| \geq (1 - \frac{\alpha}{2\Delta})n$. Here we use the assumptions that $n \gg \Delta/\alpha$ and $\Delta \gg \log(1/\alpha)$, which are necessary in order to apply Lemma 3.1 with the above parameters.

From now on we assume that $|A'| \geq (1 - \frac{\alpha}{2\Delta})n$ (which happens with high probability). Now, let $B = \{u \in V(H) : N_O^-(u, V(H) \setminus A') \neq \emptyset\}$, where O is the orientation from Strategy 3.2. It readily follows from the description of Builder's strategy (namely, Strategy 3.2) that if $x \in A' \setminus B$, then after ℓ rounds of the process, $\varphi(x)$ is adjacent in Builder's graph to every vertex of $\varphi(N_H(x))$. Thus, $A := A' \setminus B$ is φ -good. Since the maximum degree of H is Δ , it follows that $|B| \leq \Delta \cdot |V(H) \setminus A'| \leq \frac{\alpha n}{2}$. We conclude that w.h.p. $|A| \geq (1 - \alpha)n$, completing the proof. \square

3.2 Improving the Embedding In this section we introduce and analyze Builder's strategy for the second stage (see Strategy 3.6 below). Our starting point is the state of Builder's graph immediately after applying the strategy given by Lemma 3.4. (The value of the parameter α , with which this lemma is applied, will be chosen later; the bijection φ can be chosen arbitrarily.) Our goal is to iteratively update φ , so as to maintain a φ -good set which gradually increases in size until it equals $V(H)$.

Before delving into the details, let us illustrate the idea behind Strategy 3.6 by considering the following "toy" example: suppose that at some point during the process, Builder has already managed to obtain a bijection $\varphi : V(H) \rightarrow [n]$ for which there is a φ -good set A of size $n - 1$. Let b denote the unique element of $V(H) \setminus A$. The fact that A is φ -good means that $\varphi(A)$ spans a copy of $H[A]$ in Builder's graph. So in order to make φ an embedding of H into Builder's graph, it remains to connect $\varphi(b)$ to all of the vertices in $\varphi(N_H(b))$. A naive way of doing this would be for Builder to wait until $\varphi(b)$ will have been offered $d_H(b)$ times, and at each such time, to connect $\varphi(b)$ to a new vertex in $\varphi(N_H(b))$. This, however, will not work, since the probability that $\varphi(b)$ is offered (even once) in the course of $O(n)$ rounds does not tend to 1. So instead, Builder will try to find another vertex in $[n]$ to "play" the role $\varphi(b)$, and to have $\varphi(b)$ play the role which was

previously played by that other vertex. To this end, Builder fixes (a large number of) vertices $a_1, \dots, a_m \in A$ which are not adjacent to b . (We note that in order to make the strategy work, we need the additional assumption that the neighbourhoods of a_1, \dots, a_m are pairwise-disjoint, but the reader may ignore this issue at the moment.) Now Builder acts as follows: each time a vertex of $\{\varphi(a_1), \dots, \varphi(a_m)\}$ is offered, Builder connects it to some new vertex of $\varphi(N_H(b))$; and each time a vertex of $\bigcup_{i=1}^m \varphi(N_H(a_i))$ is offered, Builder connects it to $\varphi(b)$. Now, if at some point there is an index $1 \leq i \leq m$ such that $\varphi(a_i)$ has already been offered at least $\Delta(H)$ times and every vertex in $\varphi(N_H(a_i))$ has already been offered at least once, then at this point $\varphi(a_i)$ is adjacent in Builder's graph to every vertex of $\varphi(N_H(b))$, and $\varphi(b)$ is adjacent in Builder's graph to every vertex of $\varphi(N_H(a_i))$. Hence, Builder can now safely "switch" the roles of $\varphi(a_i)$ and $\varphi(b)$. Formally, Builder defines a new bijection $\varphi' : V(H) \rightarrow [n]$ by setting $\varphi'(b) = \varphi(a_i)$, $\varphi'(a_i) = \varphi(b)$, and $\varphi'(x) = \varphi(x)$ for every $x \in V(H) \setminus \{b, a_i\}$. Then φ' is an embedding of H into Builder's graph. A key point of this method is that, since there are many "candidates" for the role of b (i.e. the vertices a_1, \dots, a_m), it is very likely that one of them will indeed be chosen to be switched with b .

We now give a precise definition of the setting in which we will apply our "role-switching" strategy.

SETTING 3.5. *We are given a graph⁶ G on the vertex-set $[n]$, a bijection $\varphi : V(H) \rightarrow [n]$, and a φ -good set $A \subseteq V(H)$. We set $B = V(H) \setminus A$ and write $B = \{b_1, \dots, b_r\}$. We are also given an integer $m > 0$ and distinct vertices $a_{i,k} \in A$, where $1 \leq i \leq r$ and $1 \leq k \leq m$. Finally, we are given an integer⁷ d such that $d_H(x) \leq d$ for every $x \in \bigcup_{i=1}^r \{b_i, a_{i,1}, \dots, a_{i,m}\}$. We assume that the following two properties are satisfied.*

1. *There are no edges in H between $\{a_{i,k} : (i,k) \in [r] \times [m]\}$ and B .*
2. *The sets $\{a_{i,k}\} \cup N_H(a_{i,k})$ are pairwise-disjoint, where (i,k) run over all pairs in $[r] \times [m]$.*

Throughout the second stage of his strategy, Builder maintains and updates sets $A_t \subseteq V(H)$ and bijections $\varphi_t : V(H) \rightarrow [n]$. Initially, $A_0 = A$ and $\varphi_0 = \varphi$. For

⁶We think of G as Builder's graph immediately after employing the strategy whose existence is guaranteed by Lemma 3.4 for the number of rounds specified in that lemma.

⁷The reason for allowing flexibility in the choice of d (as opposed to simply letting d be the maximum degree of H), is that in one application (namely, Theorem 1.7), we will be able to make sure that the degrees of the vertices $\{b_i, a_{i,1}, \dots, a_{i,m} : 1 \leq i \leq r\}$ are much smaller than $\Delta(H)$, and this will be crucial for obtaining the desired bound.

every positive integer t , the pair (A_t, φ_t) will be defined immediately after round t of the second stage. We also set $B_t = V(H) \setminus A_t$ (so in particular, $B_0 = B$). Finally, we let G_t denote Builder's graph after exactly t rounds of the second stage (so in particular, $G_0 = G$). We are now ready to describe Builder's strategy for round t of the second stage (for any integer $t \geq 1$).

STRATEGY 3.6. *Let $w_t \in [n]$ be the random vertex Builder is offered at round t of the second stage.*

1. *If some pair $(i, k) \in [r] \times [m]$ is such that $w_t \in \varphi(\{a_{i,k}\} \cup N_H(a_{i,k}))$ and $b_i \in B_{t-1}$, then do:*

(a) *If $w_t \in \varphi(N_H(a_{i,k}))$, then claim the edge $\{w_t, \varphi(b_i)\}$.*

(b) *If $w_t = \varphi(a_{i,k})$, then choose an arbitrary vertex $u \in \varphi_{t-1}(N_H(b_i) \cap A_{t-1})$ which is not adjacent to $\varphi(a_{i,k})$ in G_{t-1} , and claim the edge $\{\varphi(a_{i,k}), u\}$.*

(c) *Check whether $\varphi(b_i)$ is adjacent in G_t to every vertex of $\varphi(N_H(a_{i,k}))$ and, moreover, $\varphi(a_{i,k})$ is adjacent in G_t to every vertex of $\varphi_{t-1}(N_H(b_i) \cap A_{t-1})$. If so, then set $A_t = A_{t-1} \cup \{b_i\}$ (and hence $B_t = V(H) \setminus A_t = B_{t-1} \setminus \{b_i\}$), and*

$$\varphi_t(x) = \begin{cases} \varphi_{t-1}(b_i) & x = a_{i,k}, \\ \varphi_{t-1}(a_{i,k}) & x = b_i, \\ \varphi_{t-1}(x) & x \in V(H) \setminus \{a_{i,k}, b_i\}. \end{cases}$$

(d) *Otherwise (i.e., if the condition in Item 1(c) does not hold), set $A_t = A_{t-1}$ and $\varphi_t = \varphi_{t-1}$.*

2. *Else (i.e., if there is no pair $(i, k) \in [r] \times [m]$ which satisfies the condition in Item 1), claim an arbitrary edge which is incident with w_t ; this edge will not be considered as part of Builder's graph in our analysis. Set $A_t = A_{t-1}$ and $\varphi_t = \varphi_{t-1}$.*

The operation of defining A_t and φ_t as done in Item 1(c), is referred to as *switching $a_{i,k}$ and b_i* . This name stems from the fact that we swap the vertices which play the roles of b_i and $a_{i,k}$ in our current partial embedding φ_t of H into G_t . Evidently, switching $a_{i,k}$ and b_i does not change the role of any other vertex (i.e., except for the two vertices playing the roles of $a_{i,k}$ and b_i at the current moment). Switching $a_{i,k}$ and b_i is only done if, roughly speaking, the vertex currently playing the role of b_i can play the role of $a_{i,k}$, and the vertex currently playing the role of $a_{i,k}$ can play the role of b_i ; the exact condition for switching is stated in Item 1(c) above. Note that the only pairs of vertices which can be switched, are of the form $(a_{i,k}, b_i)$ for some $1 \leq i \leq r$

and $1 \leq k \leq m$. In the following lemma we collect several simple facts regarding Strategy 3.6.

LEMMA 3.7. *Consider the execution of Strategy 3.6 for ℓ consecutive rounds, where ℓ is an arbitrary positive integer. Then the following statements hold.*

1. *If $a_{i,k}$ and b_i were switched in round t , then none of the vertices $b_i, a_{i,1}, \dots, a_{i,m}$ was switched at any other round.*

2. *If $x \in V(H)$ was not switched at any round, then $\varphi_s(x) = \varphi(x)$ for every $0 \leq s \leq \ell$. If x was switched in round t , then $\varphi_s(x) = \varphi(x)$ for every $0 \leq s \leq t-1$ and $\varphi_s(x) = \varphi_t(x)$ for every $t \leq s \leq \ell$.*

3. *For every $x \in V(H) \setminus (\bigcup_{i=1}^r \{b_i, a_{i,1}, \dots, a_{i,m}\})$, we have $\varphi_s(x) = \varphi(x)$ for every $0 \leq s \leq \ell$.*

4. *Fix $1 \leq i \leq r$, and define $N_t := \varphi_{t-1}(N_H(b_i) \cap A_{t-1})$ for every $1 \leq t \leq \ell$. Then, $N_t \subseteq N_{t'}$ holds for every $1 \leq t < t' \leq \ell$.*

Proof. We start with Item 1. Switching $a_{i,k}$ and b_i in round t forces $b_i \in A_t$. It then follows by the description of Strategy 3.6 that $b_i \in A_s$ for every $t < s \leq \ell$, making the condition in Item 1 of Strategy 3.6 false for b_i in each of the rounds $t+1, \dots, \ell$. Hence, if $a_{i,k}$ and b_i were switched in round t , then none of the vertices $b_i, a_{i,1}, \dots, a_{i,m}$ could have been switched in any subsequent round. Moreover, none of these vertices could have been switched in any round prior to round t as this would have made b_i ineligible for switching in round t . This proves Item 1. Item 2 can be easily proved by induction, using Item 1 and the definition of the functions $(\varphi_s : 0 \leq s \leq \ell)$ in Strategy 3.6. Item 3 follows from Item 2 and the fact that only vertices in $\bigcup_{i=1}^r \{b_i, a_{i,1}, \dots, a_{i,m}\}$ can be switched.

Let us prove Item 4. Fix $1 \leq i \leq r$ and $1 \leq t < t' \leq \ell$, and let $v \in N_t$ be an arbitrary vertex; we will prove that $v \in N_{t'}$. Set $x = \varphi_{t-1}^{-1}(v)$, and note that $x \in N_H(b_i) \cap A_{t-1} \subseteq N_H(b_i) \cap A_{t'-1}$. We will show that $\varphi_{t'-1}(x) = \varphi_{t-1}(x) = v$, which would imply that $v \in \varphi_{t'-1}(N_H(b_i) \cap A_{t'-1}) = N_{t'}$, as required. Assume first that $x \in A$. It then follows by Item 1 of Setting 3.5 that $x \notin \bigcup_{j=1}^r \{b_j, a_{j,1}, \dots, a_{j,m}\}$. By Item 3 of Lemma 3.7 we then have $\varphi_s(x) = \varphi(x)$ for every $0 \leq s \leq \ell$; in particular, $\varphi_{t'-1}(x) = \varphi_{t-1}(x) = v$, as claimed. Suppose now that $x \in B$, that is, $x = b_j$ for some $1 \leq j \neq i \leq r$. Since $b_j = x \in A_{t-1}$, the vertex b_j must have been switched with some $a_{j,k}$ prior to round t . Now, Item 2 of Lemma 3.7 implies that $\varphi_{t'-1}(x) = \varphi_{t-1}(x) = v$ in this case as well. \square

The following lemma can be thought of as a proof of the "correctness" of Strategy 3.6.

LEMMA 3.8. *For every non-negative integer t , the set A_t is (G_t, φ_t) -good.*

Proof. The proof is by induction on t . The base case $t = 0$ is immediate from our assumption that A is φ -good (see Setting 3.5), and the fact that $A_0 = A$, $\varphi_0 = \varphi$ and $G_0 = G$. For the induction step, fix some $t \geq 1$ and suppose that the assertion of the lemma holds for $t - 1$. Consider the execution of Strategy 3.6 in round t . If either the condition in Item 1 or the condition in Item 1(c) does not hold, then there is nothing to prove, since in that case $A_t = A_{t-1}$ and $\varphi_t = \varphi_{t-1}$. Suppose then that both of these conditions hold, and let $(i, k) \in [r] \times [m]$ be the pair satisfying the condition in Item 1 of Strategy 3.6. That is, we assume that $a_{i,k}$ and b_i were switched in round t .

We need to show that for every $x, y \in A_t$, if $\{x, y\} \in E(H)$, then $\{\varphi_t(x), \varphi_t(y)\} \in E(G_t)$. Hence, let $x, y \in A_t$ be such that $\{x, y\} \in E(H)$. Note that $\{x, y\} \neq \{a_{i,k}, b_i\}$, as $\{a_{i,k}, b_i\} \notin E(H)$ by Item 1 in Setting 3.5. If $x, y \notin \{a_{i,k}, b_i\}$ then we have $x, y \in A_{t-1}$, $\varphi_t(x) = \varphi_{t-1}(x)$ and $\varphi_t(y) = \varphi_{t-1}(y)$; so our assertion that $\{\varphi_t(x), \varphi_t(y)\} \in E(G_t)$ follows from the induction hypothesis for $t - 1$. Therefore, without loss of generality, we may assume that $x \in \{a_{i,k}, b_i\}$ and $y \notin \{a_{i,k}, b_i\}$. This assumption implies that $\varphi_t(y) = \varphi_{t-1}(y)$ and that $y \in A_{t-1}$.

Suppose first that $x = a_{i,k}$. Since $\{x, y\} \in E(H)$, it follows that $y \in N_H(x) = N_H(a_{i,k})$. Now Items 1 and 2 of Setting 3.5 imply that $y \notin \bigcup_{i=1}^r \{b_i, a_{i,1}, \dots, a_{i,m}\}$, which in turn implies that $\varphi_t(y) = \varphi(y) \in \varphi(N_H(a_{i,k}))$, see Item 3 of Lemma 3.7. As for x , it follows from the definition of φ_t in Item 1(c) of Strategy 3.6 that $\varphi_t(x) = \varphi_t(a_{i,k}) = \varphi_{t-1}(b_i) = \varphi(b_i)$, where the last equality holds by Item 2 of Lemma 3.7. Since $a_{i,k}$ and b_i were switched in round t , it follows by Item 1(c) of Strategy 3.6 that $\varphi_t(x) = \varphi(b_i)$ is adjacent in G_t to all vertices of $\varphi(N_H(a_{i,k}))$. In particular, $\{\varphi_t(x), \varphi_t(y)\} \in E(G_t)$ as required.

Suppose now that $x = b_i$. Since $\{x, y\} \in E(H)$, it follows that $y \in N_H(b_i)$. Therefore, $\varphi_t(y) = \varphi_{t-1}(y) \in \varphi_{t-1}(N_H(b_i) \cap A_{t-1})$. Observe that $\varphi_t(x) = \varphi_t(b_i) = \varphi_{t-1}(a_{i,k}) = \varphi(a_{i,k})$, where the last equality holds by Item 2 of Lemma 3.7. Since $a_{i,k}$ and b_i were switched in round t , it follows by Item 1(c) of Strategy 3.6 that $\varphi(a_{i,k})$ is adjacent in G_t to all vertices of $\varphi_{t-1}(N_H(b_i) \cap A_{t-1})$. In particular, $\{\varphi_t(x), \varphi_t(y)\} \in E(G_t)$. This concludes the proof of the lemma. \square

In the following three lemmas, we consider the execution of Strategy 3.6 for ℓ rounds for some positive integer ℓ . For every $1 \leq i \leq r$ and $1 \leq k \leq m$, we denote by $\mathcal{A}_{i,k}$ the event: “ $\varphi(a_{i,k})$ was offered at least d times **after** each of the vertices in $\varphi(N_H(a_{i,k}))$ had

already been offered”. In other words, $\mathcal{A}_{i,k}$ is the event that there are indices $1 \leq t_1 < \dots < t_q < s_1 < \dots < s_d \leq \ell$, where $q = |N_H(a_{i,k})|$, such that each element of $\varphi(N_H(a_{i,k}))$ was offered in one of the rounds t_1, \dots, t_q , and $\varphi(a_{i,k})$ was offered in each of the rounds s_1, \dots, s_d .

LEMMA 3.9. *Let $1 \leq i \leq r$. If there exists some $1 \leq k \leq m$ for which $\mathcal{A}_{i,k}$ occurred, then $b_i \in A_\ell$.*

Proof. Suppose for a contradiction that $\mathcal{A}_{i,k}$ occurred for some $1 \leq i \leq r$ and $1 \leq k \leq m$, but $b_i \notin A_\ell$, i.e., $b_i \in B_\ell$. This means that b_i was not switched at any of the ℓ rounds for which we execute Strategy 3.6. Set $q = |N_H(a_{i,k})|$ and let $1 \leq t_1 < \dots < t_q < s_1 < \dots < s_d \leq \ell$ be the round numbers appearing in the definition of $\mathcal{A}_{i,k}$. Item 1(a) of Strategy 3.6 dictates that whenever a vertex from $\varphi(N_H(a_{i,k}))$ is sampled, Builder connects it to $\varphi(b_i)$. This implies that, for every $t_q \leq t \leq \ell$, every vertex of $\varphi(N_H(a_{i,k}))$ is adjacent in G_t to $\varphi(b_i)$.

As in Lemma 3.7, we let $N_t = \varphi_{t-1}(N_H(b_i) \cap A_{t-1})$ for each $1 \leq t \leq \ell$. Suppose first that there exists some $1 \leq j \leq d$ such that $\varphi(a_{i,k})$ is adjacent in G_{s_j} to every vertex of N_{s_j} . Then by Item 1(c) of Strategy 3.6, Builder would have switched $a_{i,k}$ and b_i in round s_j , which would contradict our assumption that b_i was never switched. Hence, for every $1 \leq j \leq d$ there exists a vertex of N_{s_j} which is not adjacent in G_{s_j} to $\varphi(a_{i,k})$. It follows by Item 1(b) of Strategy 3.6 that, in round s_j , Builder claims an edge $\{\varphi(a_{i,k}), u_j\}$ for some $u_j \in N_{s_j}$ which is not adjacent to $\varphi(a_{i,k})$ in G_{s_j-1} . Note that u_1, \dots, u_d are distinct. It follows by Item 4 of Lemma 3.7 that $u_1, \dots, u_d \in N_{s_d}$. On the other hand, $|N_{s_d}| \leq |N_H(b_i)| \leq d$, implying that $N_{s_d} = \{u_1, \dots, u_d\}$. But this means that in the graph G_{s_d} , the vertex $\varphi(a_{i,k})$ is adjacent to every vertex of N_{s_d} , contrary to the above. \square

The following technical lemma provides lower bounds on the probability of the events $\mathcal{A}_{i,k}$.

LEMMA 3.10. *Fix any $1 \leq i \leq r$ and $1 \leq k \leq m$.*

(a) *If $\ell \geq 2d$, then*

$$\mathbb{P}[\mathcal{A}_{i,k}] \geq \left(\frac{\ell^2}{12dn^2} \right)^d \cdot e^{-\frac{(d+1)(\ell-2d)}{n-d-1}}.$$

(b) *If $\ell \geq (\log(2d) + d + 3\sqrt{d})n$, then $\mathbb{P}[\mathcal{A}_{i,k}] \geq \frac{1}{4}$.*

Proof. We start with Item (a). Since $|N_H(a_{i,k})| \leq d$, it

follows that

$$\begin{aligned} \mathbb{P}[\mathcal{A}_{i,k}] &\geq \binom{\ell}{2d} \cdot d! \cdot \left(\frac{1}{n}\right)^{2d} \cdot \left(1 - \frac{d+1}{n}\right)^{\ell-2d} \\ &\geq \left(\frac{\ell}{2dn}\right)^{2d} \cdot \left(\frac{d}{e}\right)^d \cdot e^{-\frac{(d+1)(\ell-2d)}{n-d-1}} \\ &\geq \left(\frac{\ell^2}{12dn^2}\right)^d \cdot e^{-\frac{(d+1)(\ell-2d)}{n-d-1}}, \end{aligned}$$

where in the second inequality we used the estimates $1-x \geq e^{-\frac{x}{1-x}}$ (which holds for every $0 < x < 1$) and $d! \geq \left(\frac{d}{e}\right)^d$ (which holds for every $d \geq 1$).

Next, we prove (b). Let \mathcal{E}_k be the event that every vertex of $\varphi(N_H(a_{i,k}))$ was offered in the course of the first $\log(2d)n$ rounds, and let \mathcal{F}_k be the event that $\varphi(a_{i,k})$ was offered at least d times in the course of the last $(d+3\sqrt{d})n$ rounds. Since $\ell \geq (\log(2d) + d + 3\sqrt{d})n$ by assumption, the events \mathcal{E}_k and \mathcal{F}_k are independent. Note that $\mathcal{E}_k \cap \mathcal{F}_k \subseteq \mathcal{A}_{i,k}$; that is, if both \mathcal{E}_k and \mathcal{F}_k occur, then so does $\mathcal{A}_{i,k}$. Therefore, $\mathbb{P}[\mathcal{A}_{i,k}] \geq \mathbb{P}[\mathcal{E}_k \cap \mathcal{F}_k] = \mathbb{P}[\mathcal{E}_k] \cdot \mathbb{P}[\mathcal{F}_k]$. The probability that \mathcal{E}_k did not occur is at most

$$d \cdot \left(1 - \frac{1}{n}\right)^{\log(2d)n} \leq d \cdot e^{-\log(2d)} = \frac{1}{2},$$

and the probability that \mathcal{F}_k did not occur equals the probability that $\text{Bin}\left(\left(d+3\sqrt{d}\right)n, 1/n\right)$ is smaller than d , which is at most

$$\begin{aligned} \mathbb{P}\left[\text{Bin}\left(\left(d+3\sqrt{d}\right)n, 1/n\right) < d\right] &\leq e^{-\frac{9d}{2(d+3\sqrt{d})}} \\ &\leq e^{-\frac{9d}{8d}} \leq \frac{1}{2}. \end{aligned}$$

Here in the first inequality we used Lemma 2.1 with $\lambda = 3\sqrt{d}$. We thus conclude that $\mathbb{P}[\mathcal{A}_{i,k}] \geq 1/2 \cdot 1/2 = 1/4$ as claimed. \square

The following lemma forms the main result of Section 3.2, and plays a key role in the proofs of Theorems 1.3 and 1.7. Roughly speaking, this lemma states that if $\varphi : V(H) \rightarrow [n]$ is a bijection admitting a φ -good set that misses only a small fraction of $V(H)$, then by following Strategy 3.6 for a suitable (and not too large) number of rounds, Builder can obtain a bijection $\varphi' : V(H) \rightarrow [n]$ which admits a φ' -good set that misses significantly fewer vertices. The actual statement is somewhat technical. The proof of Lemma 3.11 utilizes Lemmas 3.8, 3.9 and 3.10, as well as some of the concentration inequalities from Section 2.

LEMMA 3.11. *Let H be an n -vertex graph, let $\varphi : V(H) \rightarrow [n]$ be a bijection, let $V(H) =$*

$A \cup B$ be a partition, and consider a moment in the semi-random process at which A is φ -good with respect to Builder's graph. Write $B = \{b_1, \dots, b_r\}$. Let m be a positive integer, and let $\{a_{i,k} \in A : 1 \leq i \leq r \text{ and } 1 \leq k \leq m\}$ be vertices which satisfy Items 1 and 2 of Setting 3.5. Let $d \geq 100$ be such that $d_H(x) \leq d$ for every $x \in \bigcup_{i=1}^r \{b_i, a_{i,1}, \dots, a_{i,m}\}$. Let $\ell_1 = (\log(2d) + d + 3\sqrt{d})n$, $\ell_2 = \lceil n \cdot m^{-1/4d} \rceil$, $q_1 = \frac{1}{4}$, and $q_2 = d^{-2d}m^{-1/2}$. Fix any $j \in \{1, 2\}$, and assume that $mq_j \geq 10^6d$. Let

$$p = \begin{cases} o(1), & \frac{mq_j}{64d} \geq \log n, \\ e^{-\frac{\sqrt{n}}{\delta(d)}}, & \frac{mq_j}{64d} < \log n. \end{cases}$$

Suppose that Builder executes Strategy 3.6 for ℓ_j additional rounds. Then, with probability at least $1-p$, after ℓ_j rounds of the process Builder's graph G will satisfy the following: there will be a bijection $\varphi' : V(H) \rightarrow V(G) = [n]$ and a partition $V(H) = A' \cup B'$ such that A' is (G, φ') -good, $A \subseteq A'$, and

$$(3.1) \quad |B'| \leq \begin{cases} 0, & \frac{mq_j}{64d} \geq \log n, \\ 5n \cdot e^{-\frac{mq_j}{256d}}, & \frac{mq_j}{64d} < \log n. \end{cases}$$

Proof. We will show that the assertion of the lemma holds with $\varphi' = \varphi_{\ell_j}$, $A' = A_{\ell_j}$, and $B' = B_{\ell_j}$. The fact that A_{ℓ_j} is φ_{ℓ_j} -good follows immediately from Lemma 3.8. It thus remains to prove that (3.1) holds for $B' = B_{\ell_j}$ with probability at least $1-p$.

Fix arbitrary indices $1 \leq i \leq r$ and $1 \leq k \leq m$. For $j \in \{1, 2\}$, let us denote by $\mathbb{P}_j(\mathcal{A}_{i,k})$ the probability that $\mathcal{A}_{i,k}$ occurred in the course of the first ℓ_j rounds of the process. We claim that $\mathbb{P}_j[\mathcal{A}_{i,k}] \geq q_j$. The fact that $\mathbb{P}_1[\mathcal{A}_{i,k}] \geq \frac{1}{4} = q_1$ follows immediately from Item (b) of Lemma 3.10 and our choice of ℓ_1 . As for $j = 2$, recall that $\ell_2 = \lceil n \cdot m^{-1/4d} \rceil$, which implies that $\ell_2 \geq 2d$ holds for n which is sufficiently large with respect to d (since we trivially have $m \leq n$, and since $d \geq 100$, something like $n \geq (2d)^{1.01}$ would suffice). Therefore, Item (a) of Lemma 3.10 yields

$$\begin{aligned} \mathbb{P}_2[\mathcal{A}_{i,k}] &\geq \left(\frac{\ell_2^2}{12dn^2}\right)^d \cdot e^{-\frac{(d+1)(\ell_2-2d)}{n-d-1}} \\ &\geq \left(\frac{1}{12d}\right)^d \cdot m^{-1/2} \cdot e^{-d-1} \geq d^{-2d}m^{-1/2} = q_2, \end{aligned}$$

where the last inequality holds for sufficiently large d (our assumption that $d \geq 100$ suffices).

We have thus proved our assertion that $\mathbb{P}_j[\mathcal{A}_{i,k}] \geq q_j$ holds for every $j \in \{1, 2\}$. From now on, let us fix an arbitrary $j \in \{1, 2\}$ and assume that $mq_j \geq 10^6d$. For convenience, we put $\ell := \ell_j$ and $q := q_j$. For every $1 \leq i \leq r$, let X_i be the random variable counting the

number of indices $1 \leq k \leq m$ for which $\mathcal{A}_{i,k}$ occurred in the course of the first ℓ rounds. It follows by linearity of expectation that $\mathbb{E}[X_i] \geq mq$.

Now, consider the sequence (w_1, \dots, w_ℓ) of random vertices offered to Builder, and observe that changing any one coordinate in this sequence can change the value of X_i by at most 1 (here we use Item 2 in Setting 3.5). Furthermore, for every s , if $X_i \geq s$, then there is a set of at most $2ds$ coordinates in the sequence (w_1, \dots, w_ℓ) which certify that $X_i \geq s$ (indeed, each event $\mathcal{A}_{i,k}$ that occurred is certified by a set of at most $2d$ coordinates). It thus follows by Lemma 2.3 with $c = 1$, $g = 2d$, and $\lambda = \frac{\mathbb{E}[X_i]}{2}$, that

$$\begin{aligned} \mathbb{P}\left[X_i < \frac{\mathbb{E}[X_i]}{2} - 60\sqrt{2d\mathbb{E}[X_i]}\right] &\leq 4e^{-\frac{(\mathbb{E}[X_i]/2)^2}{16d\mathbb{E}[X_i]}} \\ &= 4e^{-\frac{\mathbb{E}[X_i]}{64d}} \leq 4e^{-\frac{mq}{64d}}. \end{aligned}$$

Therefore, with probability at least $1 - 4e^{-\frac{mq}{64d}}$, it holds that

$$\begin{aligned} X_i &\geq \frac{\mathbb{E}[X_i]}{2} - 60\sqrt{2d\mathbb{E}[X_i]} \\ &= \sqrt{\mathbb{E}[X_i]} \cdot \left(\frac{\sqrt{\mathbb{E}[X_i]}}{2} - 60\sqrt{2d} \right) \\ &\geq \sqrt{\mathbb{E}[X_i]} \cdot \left(\frac{\sqrt{mq}}{2} - 60\sqrt{2d} \right) \\ &> 0, \end{aligned}$$

where the last inequality follows from our assumption that $mq \geq 10^6d$.

Now let \mathcal{I} be the set of all $1 \leq i \leq r$ such that $X_i = 0$. It follows by Lemma 3.9 that if some $1 \leq i \leq r$ satisfies $X_i > 0$, then $b_i \notin B_\ell$. Hence, we have $B_\ell \subseteq \{b_i : i \in \mathcal{I}\}$. So to complete the proof it is enough to show that the bounds in (3.1) hold for the set \mathcal{I} . We have seen that $\mathbb{P}[i \in \mathcal{I}] \leq 4e^{-\frac{mq}{64d}}$ holds for every $1 \leq i \leq r$. Therefore,

$$\mathbb{E}[|\mathcal{I}|] \leq r \cdot 4e^{-\frac{mq}{64d}}.$$

Suppose first that $\frac{mq}{64d} \geq \log n$, and note that we have $r \leq \frac{n}{m} \leq \frac{n}{\log n}$. It follows that

$$\mathbb{E}[|\mathcal{I}|] \leq r \cdot 4e^{-\frac{mq}{64d}} \leq \frac{n}{\log n} \cdot \frac{4}{n} = o(1).$$

So by Markov's inequality, we have $|\mathcal{I}| = 0$ w.h.p., as required.

Suppose then that $\frac{mq}{64d} < \log n$. Observe that changing any one coordinate in the sequence (w_1, \dots, w_ℓ) of random vertices, can change the value of $|\mathcal{I}|$ by at most

1. Hence, it follows by Lemma 2.2 with $c = 1$ and $\lambda = n \cdot e^{-\frac{mq}{256d}} \geq n^{3/4}$, that

$$\mathbb{P}[|\mathcal{I}| \geq \mathbb{E}[|\mathcal{I}|] + \lambda] \leq e^{-\frac{\lambda^2}{2\ell}} \leq e^{-\frac{n^{3/2}}{2\ell}} \leq e^{-\frac{\sqrt{n}}{\sigma(\bar{d})}},$$

where the last inequality holds since $\ell_1, \ell_2 = O(dn)$. We conclude that with probability at least $1 - e^{-\frac{\sqrt{n}}{\sigma(\bar{d})}}$, we have $|\mathcal{I}| \leq \mathbb{E}[|\mathcal{I}|] + \lambda \leq (4r + n) \cdot e^{-\frac{mq}{256d}} \leq 5n \cdot e^{-\frac{mq}{256d}}$, as required. \square

3.3 Putting it All Together

In this section we iterate Lemma 3.11 to prove Lemma 3.13, from which Theorem 1.3 then easily follows. Lemma 3.13 roughly states that given a bijection $\varphi : V(H) \rightarrow [n]$ which admits a φ -good set covering almost all of $V(H)$, Builder can use φ as a basis for constructing a copy of H in his graph, and he can achieve this objective w.h.p. fairly quickly. In the proof we will need the following simple claim, which asserts that we can satisfy the conditions listed in Setting 3.5 with a relatively large choice of m .

CLAIM 3.12. *Let H be an n -vertex D -degenerate graph of maximum degree Δ . Let $A \cup B$ be a partition of $V(H)$ and suppose that $|A| \geq 4\Delta|B|$. Set $m = \lceil \frac{|A|}{8\Delta^2|B|} \rceil$. Then, there are vertices $\{a_{i,k} \in A : 1 \leq i \leq r := |B| \text{ and } 1 \leq k \leq m\}$ such that $d_H(a_{i,k}) \leq 2D$ for every $(i,k) \in [r] \times [m]$, and such that Conditions 1 and 2 of Setting 3.5 are satisfied.*

Proof. Let

$$A' := \{a \in A : |N_H(a) \cap A| \leq 2D \text{ and } N_H(a) \cap B = \emptyset\}.$$

Note that every $a \in A'$ satisfies $d_H(a) = |N_H(a) \cap A| \leq 2D$. Since H is D -degenerate, there are less than $|A|/2$ vertices $a \in A$ which satisfy $|N_H(a) \cap A| > 2D$. As $\Delta(H) = \Delta$, there are at most $\Delta|B| \leq |A|/4$ vertices $a \in A$ which have a neighbour in B . Altogether, we get $|A'| \geq |A|/4$. Using again the assumption $\Delta(H) = \Delta$, we infer that there exists an integer

$$M \geq \frac{|A'|}{\Delta^2 + 1} \geq \frac{|A|}{8\Delta^2}$$

and vertices $a_1, \dots, a_M \in A'$ such that $\text{dist}_H(a_i, a_j) \geq 3$ for every $1 \leq i < j \leq M$. Recalling our choice of m , we index (a subset of) the vertices a_1, \dots, a_M by pairs $(i,k) \in [r] \times [m]$. For every $(i,k) \in [r] \times [m]$, let $a_{i,k}$ be the vertex in $\{a_1, \dots, a_M\}$ which is indexed by the pair (i,k) . Now Condition 1 of Setting 3.5 is satisfied due to our choice of A' , and Condition 2 of Setting 3.5 is satisfied because $\text{dist}_H(a_{i,k}, a_{i',k'}) \geq 3$ for every choice of distinct pairs $(i,k), (i',k') \in [r] \times [m]$. \square

LEMMA 3.13. *Let H be an n -vertex D -degenerate graph of maximum degree Δ . Set $d = \min\{2D, \Delta\}$. Suppose that at some point in the semi-random process, there is a bijection $\varphi : V(H) \rightarrow [n]$ and a partition $V(H) = A \cup B$ such that A is φ -good with respect to Builder's graph; such that $|B| \leq 10^{-8}\Delta^{-5}n$; and such that $d_H(b) \leq d$ for every $b \in B$. Then Builder has a strategy guaranteeing that w.h.p., after $(d+o(d))n$ additional rounds his graph will contain a copy of H .*

Proof. We may and will assume that d is large enough, say $d \geq 100$. Builder's strategy consists of two phases, which correspond to the two cases (i.e. $j = 1$ and $j = 2$) in Lemma 3.11.

Phase 1: Let $m_0 := \left\lceil \frac{n}{8\Delta^2|B|} \right\rceil$. Find vertices $\{a_{i,k} \in A : 1 \leq i \leq |B| \text{ and } 1 \leq k \leq m_0\}$ such that $d_H(a_{i,k}) \leq d$ for every $(i,k) \in [r] \times [m_0]$, and such that Conditions 1 and 2 of Setting 3.5 are satisfied (with $m = m_0$). Apply the strategy whose existence is guaranteed by Lemma 3.11 with parameter $j = 1$ and with $m = m_0$. Lemma 3.11 (with $j = 1$) ensures that after $\ell_1 = (\log(2d) + d + 3\sqrt{d})n = (d + o(d))n$ rounds, there will be a bijection $\varphi_0 : V(H) \rightarrow [n]$ and a partition $V(H) = A_0 \cup B_0$ such that A_0 is φ_0 -good with respect to Builder's graph, $A \subseteq A_0$ (and hence $B_0 \subseteq B$), and

$$(3.2) \quad |B_0| \leq \begin{cases} 0, & \frac{m_0}{256d} \geq \log n, \\ 5n \cdot e^{-\frac{m_0}{1024d}}, & \frac{m_0}{256d} < \log n. \end{cases}$$

If $B_0 = \emptyset$ then $A_0 = V(H)$ is φ_0 -good, implying that Builder has successfully embedded H into his graph, and so Builder is done. Otherwise, proceed to Phase 2.

Phase 2: Define a sequence of bijections $\varphi_1, \varphi_2, \dots$ from $V(H)$ to $[n]$, and a sequence of partitions $A_1 \cup B_1, A_2 \cup B_2, \dots$ of $V(H)$, by performing the following steps for every integer $t \geq 1$ for which $B_{t-1} \neq \emptyset$.

(a) Find vertices $\{a_{i,k} \in A_{t-1} : 1 \leq i \leq |B_{t-1}| \text{ and } 1 \leq k \leq m_t := \left\lceil \frac{n}{8\Delta^2|B_{t-1}|} \right\rceil\}$ such that $d_H(a_{i,k}) \leq d$ for every $(i,k) \in [r] \times [m_t]$, and such that Conditions 1 and 2 of Setting 3.5 are satisfied for the partition $A_{t-1} \cup B_{t-1}$ with $m = m_t$.

(b) Invoke the strategy whose existence is guaranteed by Lemma 3.11 with $j = 2$, with $m = m_t$, and with input φ_{t-1} and $A_{t-1} \cup B_{t-1}$. Using this strategy, Builder obtains a bijection $\varphi_t : V(H) \rightarrow [n]$ and a partition $A_t \cup B_t$ of $V(H)$ such that A_t is φ_t -good with respect to Builder's graph, $A_{t-1} \subseteq A_t$ (and hence $B_t \subseteq B_{t-1}$), and

$$(3.3) \quad |B_t| \leq \begin{cases} 0, & m'_t \geq \log n, \\ 5n \cdot e^{-m'_t/4}, & m'_t < \log n, \end{cases}$$

where

$$m'_t := \frac{m_t \cdot q_2}{64d} = \frac{\sqrt{m_t}}{64d^{2d+1}},$$

and q_2 is as in Lemma 3.11. (Here and later on we slightly abuse notation by hiding the fact that $q_2 = d^{-2d}/\sqrt{m_t}$ depends on t .)

Having described Builder's strategy, we now turn to prove that w.h.p. Builder can follow it. First, note that by Claim 3.12, there are vertices $\{a_{i,k} \in A : 1 \leq i \leq |B| \text{ and } 1 \leq k \leq m_0\}$ such that $d_H(a_{i,k}) \leq d$ for every $(i,k) \in [r] \times [m_0]$, and such that Conditions 1 and 2 of Setting 3.5 hold. Moreover, the conditions required for the application of Lemma 3.11 with $j = 1$ are satisfied as $m_0 q_1 = \frac{m_0}{4} \geq \frac{n}{32\Delta^2|B|} > 10^6\Delta \geq 10^6d$, where the second inequality follows from our assumption that $|B| \leq 10^{-8}\Delta^{-5}n$. This shows that Builder can follow Phase 1 of his strategy. It remains to show that w.h.p. Builder can follow Phase 2 of his strategy. Similarly to Phase 1, the existence of the vertices $\{a_{i,k} \in A_{t-1} : 1 \leq i \leq |B_{t-1}| \text{ and } 1 \leq k \leq m_t\}$ for every positive integer t , follows from Claim 3.12 with input $A_{t-1} \cup B_{t-1}$. It remains to prove that the conditions of Lemma 3.11 are met whenever Builder wishes to apply it (with $j = 2$). The fact that A_{t-1} is φ_{t-1} -good for every positive integer t is guaranteed by the previous applications of Lemma 3.11. The fact that $d_H(b) \leq d$ for every $b \in B_{t-1}$ follows from our assumption that the same holds for every $b \in B$, and the fact that $B_0 \subseteq B$ and $B_i \subseteq B_{i-1}$ for every $i \geq 1$. We now show that $\sqrt{m_t} \cdot d^{-2d} = m_t \cdot q_2 \geq 10^6d$ holds for every positive integer t for which $B_{t-1} \neq \emptyset$. To this end, first note that

$$(3.4) \quad |B_0| \leq 5n \cdot e^{-\frac{m_0}{1024d}} \leq 5n \cdot \exp\left(-\frac{n}{2^{13}\Delta^3|B|}\right) e \leq n \cdot e^{-\Delta^2},$$

Here, the first inequality follows from (3.2), and the third from our assumption that $|B| \leq 10^{-8}\Delta^{-5}n$. Since $B_i \subseteq B_{i-1}$ for every $i \geq 1$, we have $|B_i| \leq n \cdot e^{-\Delta^2}$ for each $i \geq 0$. Now we obtain

$$(3.5) \quad \begin{aligned} m_t \cdot q_2 &= \sqrt{m_t} \cdot d^{-2d} \geq \frac{\sqrt{n}}{3\Delta^{2\Delta+1}\sqrt{|B_{t-1}|}} \\ &\geq \frac{e^{\Delta^2/2}}{3\Delta^{2\Delta+1}} \geq 10^6\Delta \geq 10^6d, \end{aligned}$$

where the penultimate inequality holds for sufficiently large Δ (say, $\Delta \geq 100$). This shows that we can indeed apply Lemma 3.11 with $j = 2$ and with input $A_{t-1} \cup B_{t-1}$ for every positive integer t for which $B_{t-1} \neq \emptyset$. We conclude that Builder can follow Phase 2 of his strategy.

Before moving on to prove the correctness of Builder's strategy, we first prove the following claim.

CLAIM 3.14. Let $\zeta = e^{-\Delta^2}$ and, for every non-negative integer t , let $\beta_t = |B_t|/n$. If $t = 0$, or if $t \geq 1$ and $m'_t < \log n$, then $\beta_t \leq \zeta^{t+1}$.

Proof. Our proof proceeds by induction on t . The base case $t = 0$ follows from (3.4). Let then $t \geq 1$ and suppose that $m'_t < \log n$. Observe that the sequence m'_s is monotone non-decreasing in s (this follows from the fact that $B_i \subseteq B_{i-1}$ for every $i \geq 1$). So either $t - 1 = 0$, or $m'_{t-1} < \log n$. In either case we can apply the induction hypothesis to get $\beta_{t-1} \leq \zeta^t$. Note that

$$(3.6) \quad m'_t = \frac{\sqrt{m_t}}{64d^{2d+1}} \geq \frac{\sqrt{n}}{200\Delta^{2\Delta+2}\sqrt{|B_{t-1}|}}.$$

Now we get

$$\begin{aligned} \beta_t &\leq 5 \cdot e^{-m'_t/4} \leq 5 \cdot \exp\left(-\frac{1}{800\Delta^{2\Delta+2}\sqrt{\beta_{t-1}}}\right) \\ &\leq 5 \cdot \exp\left(-\frac{1}{800\Delta^{2\Delta+2}\zeta^{t/2}}\right) \\ &= 5 \cdot \exp\left(-\frac{e^{\Delta^2 t/2}}{800\Delta^{2\Delta+2}}\right) \leq e^{-\Delta^2(t+1)} = \zeta^{t+1}, \end{aligned}$$

where the first inequality holds by (3.3), the second inequality holds by (3.6), the third inequality holds by the induction hypothesis for $t - 1$, and the last inequality holds for every $t \geq 1$, provided that Δ is larger than some suitable absolute constant (again, $\Delta \geq 100$ suffices). This proves the claim. \square

Returning to the proof of the lemma, we now prove the correctness of Builder's strategy. For the time being, we will assume that all applications of Lemma 3.11 throughout Builder's strategy are successful; later we will show that w.h.p. this is indeed the case. It follows from (3.6) and from Claim 3.14 that $m'_t \geq e^{\Delta^2 t/2} \cdot \frac{1}{200\Delta^{2\Delta+2}}$ holds for every $t \geq 1$. Hence, $m'_t \geq \log n$ must hold for some $t \leq \log \log n$ (and in fact much earlier, but we will not need this). Now, if $m'_t \geq \log n$ then by (3.3) we have $B_t = \emptyset$, which in turn implies that Builder has successfully embedded H into the graph he is constructing.

Next, we estimate the probability that Builder's strategy fails. Recall that Lemma 3.11 is only applied once with parameter $j = 1$, and that this application is w.h.p. successful. Let us now consider the applications of Lemma 3.11 with $j = 2$ (in Phase 2). As previously noted, there is at most one such application with $m'_t \geq \log n$, and at most $\log \log n$ such applications with $m'_t < \log n$. The failure probability of the former application is $o(1)$, and the failure probability of each of the latter applications is at most $e^{-\frac{\sqrt{n}}{o(d)}}$. We thus

conclude that w.h.p. all of the above applications of Lemma 3.11 are successful, as required. This completes the proof of correctness of Builder's strategy.

It remains to estimate the overall number of rounds required for implementing Builder's strategy. Recall that the sole application of Lemma 3.11 with $j = 1$ requires $(d + o(d))n$ rounds. It thus remains to bound from above the number of rounds required for Phase 2 of Builder's strategy. To this end, let t^* denote the smallest integer t satisfying $m'_t \geq \log n$, and note that $t^* \leq \log \log n$. Then in Phase 2, Lemma 3.11 was invoked at most t^* times. Moreover, for each $1 \leq t \leq t^*$, invoking Lemma 3.11 with input $A_{t-1} \cup B_{t-1}$ (and with $j = 2$) required at most

$$\begin{aligned} \lceil n \cdot m_t^{-\frac{1}{4d}} \rceil &\leq n \cdot m_t^{-\frac{1}{4d}} + 1 \leq n \cdot \left(\frac{n}{8\Delta^2|B_{t-1}|}\right)^{-\frac{1}{4d}} + 1 \\ &\leq O(n) \cdot \left(\frac{1}{\beta_{t-1}}\right)^{-\frac{1}{4d}} + 1 \\ &\leq O(n) \cdot e^{-\Delta t/4} + 1 \end{aligned}$$

rounds, where in the last inequality we used Claim 3.14. Therefore, the overall number of rounds required for the (at most) t^* applications of Lemma 3.11 in Phase 2 is no more than

$$\begin{aligned} \sum_{t=1}^{t^*} \left(O(n) \cdot e^{-\Delta t/4} + 1\right) &\leq O(n) \cdot \sum_{t=1}^{\infty} e^{-\Delta t/4} + \log \log n \\ &= O(n). \end{aligned}$$

We conclude that the overall number of rounds required for implementing Builder's strategy is at most $(d + o(d))n$, thus completing the proof. \square

Equipped with Lemma 3.13, we can finally prove Theorem 1.3.

Proof of Theorem 1.3. Let n , Δ , and H be as in the statement of the theorem. Due to Proposition 3.3, we only need to handle the case $\Delta = O(\log n)$. Set $\alpha = 10^{-8}\Delta^{-5}$. Builder's strategy for embedding H is as follows. Fix an arbitrary bijection $\varphi : V(H) \rightarrow [n]$. In the first $(\Delta/2 + o(\Delta))n$ rounds, Builder invokes the strategy whose existence is guaranteed by Lemma 3.4, and thus obtains w.h.p. a φ -good set $A \subseteq V(H)$ of size at least $(1 - \alpha)n$. Setting $B := V(H) \setminus A$, observe that the requirements of Lemma 3.13 are satisfied for $D := \Delta$. Builder now applies the strategy given by Lemma 3.13 for an additional $(\Delta + o(\Delta))n$ rounds, and by doing so successfully constructs a copy of H in his graph w.h.p. The overall number of rounds is then $(3\Delta/2 + o(\Delta))n$, as required. \square

4 Constructing Spanning Forests

In this section we prove Theorem 1.7 and Proposition 1.8. We start with the following simple lemma, whose proof demonstrates a strategy for greedily embedding an almost-spanning forest.

LEMMA 4.1. *Let n be a positive integer and let $\alpha \in (0, 1)$ be such that $n \gg \alpha^{-2} \log(1/\alpha)$. Let T' be a forest on $(1 - \alpha)n$ vertices and let $\ell = \log(2/\alpha) \cdot n$. Then, in the semi-random process on n vertices, Builder has a strategy which w.h.p. allows him to construct a copy of T' within ℓ rounds.*

Proof. Assume without loss of generality that T' is a tree (otherwise simply replace T' with a tree containing it). Let $t = (1 - \alpha)n$ and let v_1, \dots, v_t be an ordering of the vertices of T' such that $T'[\{v_1, \dots, v_i\}]$ is a tree for every $1 \leq i \leq t$. Throughout the process, Builder maintains a partial function φ which is initially empty. For every positive integer i , let w_i denote the vertex Builder is offered in the i th round. In the first round, Builder connects w_1 to an arbitrary vertex u ; he then sets $\varphi(v_1) = w_1$ and $\varphi(v_2) = u$. For every $i \geq 2$, Builder plays the i th round as follows. Let r denote the largest integer for which $\varphi(v_r)$ has already been defined. If $w_i \notin \{\varphi(v_1), \dots, \varphi(v_r)\}$, then Builder connects w_i to $\varphi(v_j)$, where $j \leq r$ is the unique integer for which $\{v_{r+1}, v_j\} \in E(T')$; he then sets $\varphi(v_{r+1}) = w_i$. Otherwise, Builder claims an arbitrary edge which is incident with w_i , but does not consider this edge to be part of the tree he is building (alternatively, Builder skips this round).

It is evident that, by following the proposed strategy, Builder's graph contains a copy of T' as soon as t different vertices are offered. Hence, it suffices to prove that w.h.p. at least t different vertices are offered during the first ℓ rounds. For every $1 \leq j \leq n$, let I_j be the indicator random variable for the event: "vertex j was not offered during the first ℓ rounds of the process". Let $X = \sum_{j=1}^n I_j$; then

$$\mathbb{E}(X) = \sum_{j=1}^n \mathbb{E}(I_j) = n(1 - 1/n)^\ell \leq n \cdot e^{-\ell/n} = \alpha n/2.$$

Observe that changing any one coordinate in the sequence (w_1, \dots, w_ℓ) of random vertices, can change the value of X by at most 1. Hence, it follows by Lemma 2.2 with $c = 1$ and $\lambda = \alpha n/2$, that

$$\begin{aligned} \mathbb{P}[X \geq \alpha n] &\leq \mathbb{P}[X \geq \mathbb{E}(X) + \alpha n/2] \leq e^{-\frac{(\alpha n/2)^2}{2t}} \\ &= e^{-\frac{\alpha^2 n^2}{8 \log(2/\alpha)^n}} = o(1), \end{aligned}$$

where the last equality holds by our assumption that $n \gg \alpha^{-2} \log(1/\alpha)$. \square

We are now in a position to prove Theorem 1.7.

Proof of Theorem 1.7. Let n , Δ , and T be as in the statement of the theorem. Assume first that $\Delta \geq n^{1/11}$. In this case Builder employs the strategy presented in the proof of Lemma 4.1. It is easy to see that as soon as each of the n vertices has been offered, Builder's graph contains a copy of T . It is well-known (and easy to prove) that this will happen w.h.p. in $(1 + o(1))n \log n = O(n \log \Delta)$ rounds. For the remainder of the proof we thus assume that $\Delta < n^{1/11}$.

Set $\alpha = 10^{-8} \Delta^{-5}$, and note that $n \gg \alpha^{-2} \log(1/\alpha)$, as required by Lemma 4.1. Since T is a forest, there are at least $n/2$ vertices of T whose degree is at most 2. Fix a set B of αn such vertices. Set $A := V(H) \setminus B$ and $T' := T[A]$. At the first stage of his strategy, Builder invokes Lemma 4.1, which enables him to construct a copy of T' w.h.p. in $\log(2/\alpha)n = O(\log \Delta)n$ rounds.

Let $\varphi : V(T) \rightarrow [n]$ be a bijection such that $\varphi|_A$ is an embedding of T' into Builder's graph. Note that the conditions of Lemma 3.13 are satisfied with $D = 1$, since forests are 1-degenerate, and since we made sure that all vertices in B have degree at most 2. In the second stage of his strategy, Builder applies (the strategy given by) Lemma 3.13 to construct a copy of T in his graph w.h.p. This requires $O(n)$ additional rounds. The total number of rounds is thus $O(\log \Delta) \cdot n$, as required. \square

4.1 Lower Bound : Proof of Proposition 1.8

We now turn to the proof of the lower bound for semi-random construction of trees.

Proof of Proposition 1.8. Let Δ and $n \geq n_0(\Delta)$ be as in the statement of the proposition. Since Builder clearly needs at least $n - 1$ rounds in order to build a tree on n vertices, we can assume that Δ is a sufficiently large constant. We prove the proposition for the n -vertex forest T consisting of $\lfloor \frac{n}{\Delta+1} \rfloor$ pairwise-disjoint $(\Delta + 1)$ -vertex stars, together with some (at most Δ) isolated vertices (if needed). Let us denote the center of the i th star by u_i , and its leaves by $x_{i,1}, \dots, x_{i,\Delta}$ ($1 \leq i \leq \lfloor \frac{n}{\Delta+1} \rfloor$).

Let w_1, w_2, \dots denote the sequence of random vertices offered to Builder, and let $m = 0.1n \log \Delta$. Suppose that Builder did manage to build a copy of T within m rounds, and let $\varphi : V \rightarrow [n]$ be a bijection such that $\{\varphi(u), \varphi(v)\}$ is an edge in Builder's graph for every $\{u, v\} \in E(T)$. It is then evident that, for every $1 \leq i \leq \lfloor \frac{n}{\Delta+1} \rfloor$, either $\varphi(u_i)$ appears at least $\sqrt{\Delta}$ times in (w_1, w_2, \dots, w_m) , or at least $\Delta - \sqrt{\Delta}$ of the elements of $\{\varphi(x_{i,j}) : 1 \leq j \leq \Delta\}$ appear at least once in (w_1, w_2, \dots, w_m) . A straightforward calculation then shows that, either at least $\frac{1}{\sqrt{\Delta}} \cdot \lfloor \frac{n}{\Delta+1} \rfloor \geq \frac{n}{2\Delta^{3/2}}$ of the vertices in $\{\varphi(u_i) : 1 \leq i \leq \lfloor \frac{n}{\Delta+1} \rfloor\}$ were offered at least

$\sqrt{\Delta}$ times each; or all but at most

$$\Delta + \Delta \cdot \frac{1}{\sqrt{\Delta}} \cdot \left\lfloor \frac{n}{\Delta+1} \right\rfloor + \left\lfloor \frac{n}{\Delta+1} \right\rfloor \cdot (\sqrt{\Delta} + 1) \leq \frac{3n}{\sqrt{\Delta}}$$

of the n vertices were offered at least once. In the inequality above we use the assumption that $n \geq n_0(\Delta)$ for some suitable $n_0(\Delta)$. So in order to prove that w.h.p. Builder needs more than m rounds to build T , it suffices to show that w.h.p. there are more than $3n/\sqrt{\Delta}$ vertices $1 \leq i \leq n$ that do not appear in (w_1, w_2, \dots, w_m) , and less than $\frac{n}{2\Delta^{3/2}}$ vertices $1 \leq i \leq n$ which appear in (w_1, w_2, \dots, w_m) at least $\sqrt{\Delta}$ times.

Let X be the random variable which counts the number of vertices $1 \leq i \leq n$ that do not appear in (w_1, w_2, \dots, w_m) . Our goal is to show that w.h.p. $X > 3n/\sqrt{\Delta}$. We have

$$\begin{aligned} \mathbb{E}(X) &= n(1 - 1/n)^m \geq n \cdot e^{-m/(n-1)} \\ &= n \cdot e^{-0.1 \log \Delta \cdot \frac{n}{n-1}} = n \cdot \Delta^{-0.1 - o(1)} > 4n/\sqrt{\Delta}, \end{aligned}$$

where the first inequality follows from the fact that $(1 - 1/n)^{n-1} \geq 1/e$ for each $n \geq 1$, and the last inequality holds for sufficiently large n and Δ . We will use Lemma 2.2 to prove that w.h.p. X is not much smaller than its expected value. Observe that changing any single coordinate in the sequence of random vertices (w_1, w_2, \dots, w_m) can change the value of X by at most 1. Therefore, applying Lemma 2.2 with parameters $c = 1$ and $\lambda = n/\sqrt{\Delta}$ yields

$$\begin{aligned} \mathbb{P}\left[X \leq 3n/\sqrt{\Delta}\right] &\leq \mathbb{P}\left[X \leq \mathbb{E}(X) - n/\sqrt{\Delta}\right] \\ &\leq e^{-\frac{(n/\sqrt{\Delta})^2}{2m}} \leq e^{-\frac{(n/\sqrt{\Delta})^2}{n \log \Delta}} \\ &= e^{-\frac{n}{\Delta \log \Delta}} = o(1), \end{aligned}$$

where the last equality holds since n is sufficiently large with respect to Δ .

Let Z be the random variable which counts the number of vertices $1 \leq i \leq n$ that appear at least $\sqrt{\Delta}$ times in (w_1, w_2, \dots, w_m) . Our goal is to prove that w.h.p. $Z < \frac{n}{2\Delta^{3/2}}$. For every $1 \leq i \leq n$, let Z_i be the random variable counting the number of times i appears in (w_1, w_2, \dots, w_m) . Then $Z_i \sim \text{Bin}(m, 1/n)$, implying that $\mathbb{E}[Z_i] = m/n \leq 0.1 \log \Delta$. Applying Lemma 2.1 with parameter $\lambda = 2 \log \Delta \leq \sqrt{\Delta} - 0.1 \log \Delta$, we obtain

$$\begin{aligned} \mathbb{P}[Z_i \geq \sqrt{\Delta}] &\leq \mathbb{P}[Z_i \geq \mathbb{E}[Z_i] + \lambda] \\ (4.7) \quad &\leq \exp\left(-\frac{(2 \log \Delta)^2}{2(\mathbb{E}[Z_i] + \frac{2}{3} \log \Delta)}\right) \\ &\leq e^{-2 \log \Delta} \leq \frac{1}{4\Delta^{3/2}}, \end{aligned}$$

where the last inequality holds for sufficiently large Δ . For every $1 \leq i \leq n$, let I_i be the indicator

random variable for the event $Z_i \geq \sqrt{\Delta}$; note that $Z = \sum_{i=1}^n I_i$. It follows by (4.7) and by the linearity of expectation that $\mathbb{E}(Z) \leq \frac{n}{4\Delta^{3/2}}$. Since changing any single coordinate in the sequence of random vertices (w_1, w_2, \dots, w_m) can change the value of Z by at most 1, applying Lemma 2.2 with parameters $c = 1$ and $\lambda = \frac{n}{4\Delta^{3/2}}$ yields

$$\begin{aligned} \mathbb{P}\left[Z \geq \frac{n}{2\Delta^{3/2}}\right] &\leq \mathbb{P}\left[Z \geq \mathbb{E}(Z) + \frac{n}{4\Delta^{3/2}}\right] \\ &\leq e^{-\frac{n^2}{32\Delta^3 m}} \leq e^{-\frac{n^2}{32\Delta^3 n \log \Delta}} = o(1), \end{aligned}$$

where the equality holds since n is sufficiently large with respect to Δ . \square

References

- [1] Dimitris Achlioptas, Raissa M. D'Souza, and Joel Spencer. Explosive percolation in random networks. *Science*, 323(5920):1453–1455, 2009.
- [2] Yossi Azar, Andrei Z. Broder, Anna R. Karlin, and Eli Upfal. Balanced allocations. *SIAM Journal on Computing*, 29(1):180–200, 1999.
- [3] Omri Ben-Eliezer, Lior Gishboliner, Dan Hefetz, and Michael Krivelevich. Very fast construction of bounded-degree spanning graphs via the semi-random graph process. *arXiv:1907.08031*, 2019. Full version of this work.
- [4] Omri Ben-Eliezer, Dan Hefetz, Gal Kronenberg, Olaf Parczyk, Clara Shikhelman, and Miloš Stojaković. Semi-random graph process. *arXiv:1805.02259*, to appear in *Random Structures and Algorithms*.
- [5] Tom Bohman and Alan M. Frieze. Avoiding a giant component. *Random Structures and Algorithms*, pages 75–85, 2001.
- [6] Tom Bohman and David Kravitz. Creating a giant component. *Combinatorics, Probability and Computing*, 15:489–511, 2006.
- [7] Allan Borodin and Ran El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, New York, NY, USA, 1998.
- [8] Asaf Ferber, Gal Kronenberg, and Kyle Luh. Optimal threshold for a random graph to be 2-universal. *Transactions of the American Mathematical Society*, 372:4239–4262, 2019.
- [9] Michael J. Freedman, Kobbi Nissim, and Benny Pinkas. Efficient private matching and set intersection. In *Advances in Cryptology - EUROCRYPT 2004*, pages 1–19, 2004.
- [10] Alan Frieze and Michał Karoński. *Introduction to Random Graphs*. Cambridge University Press, 2016.
- [11] Wulfram Gerstner and Werner M. Kistler. *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, 2002.
- [12] Craig H. Bailey, Eric R. Kandel, and Kristen M. Harris. Structural components of synaptic plasticity and mem-

- ory consolidation. *Cold Spring Harbor perspectives in biology*, 7(7), 2015.
- [13] Svante Janson, Tomasz Łuczak, and Andrzej Ruciński. *Random Graphs*. Wiley, 2000.
 - [14] Anders Johansson, Jeff Kahn, and Van Vu. Factors in random graphs. *Random Structures and Algorithms*, 33(1):1–28, 2008.
 - [15] Jeff Kahn, Eyal Lubetzky, and Nicholas Wormald. The threshold for combs in random graphs. *Random Structures and Algorithms*, 48(4):794–802, 2016.
 - [16] Mihyun Kang, Will Perkins, and Joel Spencer. The Bohman-Frieze process near criticality. *Random Structures and Algorithms*, 43(2):221–250, 2013.
 - [17] Michael Krivelevich, Po-Shen Loh, and Benny Sudakov. Avoiding small subgraphs in Achlioptas processes. *Random Structures and Algorithms*, 34:165–195, 2009.
 - [18] Michael Krivelevich, Eyal Lubetzky, and Benny Sudakov. Hamiltonicity thresholds in Achlioptas processes. *Random Structures and Algorithms*, 37:1–24, 2010.
 - [19] John Lisman, Katherine Cooper, Megha Sehgal, and Alcino Silva. Memory formation depends on both synapse-specific modifications of synaptic strength and cell-specific increases in excitability. *Nature Neuroscience*, 21(3), 2018.
 - [20] Mark Mayford, Steven A Siegelbaum, and Eric R Kandel. Synapses and memory storage. *Cold Spring Harbor perspectives in biology*, 4(6), 2012.
 - [21] Michael Molloy and Bruce Reed. *Graph Colouring and the Probabilistic Method*. Springer, 2002.
 - [22] Richard Montgomery. Spanning trees in random graphs. *Advances in Mathematics*, 356:106793, 2019.
 - [23] Tobias Müller and Reto Spöhel. A geometric Achlioptas process. *The Annals of Applied Probability*, 25(6):3295–3337, 2015.
 - [24] Torsten Mütze, Reto Spöhel, and Henning Thomas. Small subgraphs in random graphs and the power of multiple choices. *J. Combin. Theory Ser. B*, 101(4):237–268, 2011.
 - [25] Rasmus Pagh and Flemming Friche Rodler. Cuckoo hashing. *Journal of Algorithms*, 51(2):122–144, 2004.
 - [26] Oliver Riordan and Lutz Warnke. Achlioptas process phase transitions are continuous. *The Annals of Applied Probability*, 22(4):1450–1464, 2012.
 - [27] Joel Spencer and Nicholas Wormald. Birth control for giants. *Combinatorica*, 27:587–628, 2007.
 - [28] Sasu Tarkoma, Christian Esteve Rothenberg, and Eemil Lagerstetz. Theory and practice of Bloom filters for distributed systems. *IEEE Communications Surveys and Tutorials*, 14(1):131–155, 2012.