Combinatorics, Probability and Computing (2000) 9, 241–263. Printed in the United Kingdom
© 2000 Cambridge University Press

Optimal Construction of Edge-Disjoint Paths in Random Regular Graphs

ALAN M. FRIEZE† and LEI ZHAO†

Department of Mathematical Sciences, Carnegie-Mellon University, Pittsburgh, PA 15213, USA
(e-mail: alan@random.math.cmu.edu, lzhao@andrew.cmu.edu)

Received 29 September 1998; revised 21 May 1999

Given a graph $G = (V, E)$ and a set of κ pairs of vertices in V, we are interested in finding, for each pair (a_i, b_i), a path connecting a_i to b_i such that the set of κ paths so found is edge-disjoint. (For arbitrary graphs the problem is $\mathcal{NP}$-complete, although it is in $\mathcal{P}$ if κ is fixed.)

We present a polynomial time randomized algorithm for finding edge-disjoint paths in the random regular graph $G_{n,r}$, for sufficiently large r. (The graph is chosen first, then an adversary chooses the pairs of end-points.) We show that almost every $G_{n,r}$ is such that *all* sets of $\kappa = \Omega(n/\log n)$ pairs of vertices can be joined. This is within a constant factor of the optimum.

1. Introduction

Given a graph $G = (V, E)$ with n vertices, and a set of κ pairs of vertices in V, we are interested in finding, for each pair (a_i, b_i), a path connecting a_i to b_i such that the set of κ paths so found is edge-disjoint.

For arbitrary graphs the related decision problem is $\mathcal{NP}$-complete, although it is in $\mathcal{P}$ if κ is fixed (Robertson and Seymour [18]). Nevertheless, this negative result can be circumvented for certain classes of graphs. Peleg and Upfal [17] presented a polynomial time algorithm for the case where G is a (sufficiently strong) bounded degree expander graph, and $\kappa \leqslant n^{\epsilon}$ for a small constant ϵ that depends on the expansion property of the graph. (A precise upper bound for ϵ was not computed, but it is clearly less than $1/3$.) This result has been improved and extended by Broder, Frieze and Upfal [7, 8], Frieze [9], Leighton and Rao [13] and Leighton, Rao and Srinivasan [14, 15]: G still has to be a (sufficiently strong) bounded degree expander but κ can now grow as fast as $n/(\ln n)^{\theta}$, where θ depends only on the expansion properties of the input graph, but is at least 2.

In random graphs Shamir and Upfal have shown in [19] that any set of up to

† Supported in part by NSF grant CCR9530974.

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

$O(\sqrt{n})$ pairs can be connected via vertex-disjoint paths; similar results using efficient flow techniques were also obtained by Hochbaum [11]. These results were proved for graphs with $m \geqslant Kn \log n$ random edges, where K is a sufficiently large constant. In these two papers the pairs are chosen *first* and then the graph is randomly generated. Thus they do not deal with the problem of satisfying *all* sets of κ pairs.

Let D be the median distance between pairs of vertices in G. Clearly it is not possible to connect more than $O(m/D)$ pairs of vertices by edge-disjoint paths, for all choices of pairs, since some choices would require more edges than all the edges available. In the case of bounded degree expanders, this absolute upper bound on κ is $O(n/\log n)$. The results mentioned so far use only a vanishing fraction of the set of edges of the graph, thus are far from reaching this upper bound. Broder, Frieze, Suen and Upfal [5] have proved that, for the model of random graphs, $G_{n,m}$, the absolute upper bound is achievable to within a constant factor, but only if the average degree is at least $\ln n$. In this work, we show that this result holds when the minimum degree is a sufficiently large constant. Without significant loss of generality, we consider random regular graphs and present an algorithm that constructs the required paths in polynomial time.

As usual, let $G_{n,p}$ denote a random graph with vertex set $\{1, 2, \ldots, n\} = [n]$ in which each possible edge is included independently with probability p, and let $G_{n,m}$ denote a random graph also with vertex set $[n]$ and exactly m edges, all sets of m edges having equal probability. Let $G_{n,r}$ denote a random graph again with vertex set $[n]$ which is chosen uniformly at random from all possible r-regular graphs.

Our main result is formulated in the following theorem.

Theorem 1.1. *Let r be a sufficiently large constant. Then, as $n \longrightarrow \infty$, the graph $G_{n,r}$ has the following property, with high probability: there exist positive absolute constants α, β such that, for all sets of pairs of vertices $\{(a_i, b_i) \mid i = 1, \ldots, \kappa\}$ satisfying*

(i) $\kappa = \lceil \alpha rn/\log_r n \rceil$,

(ii) *for each vertex v, $|\{i : a_i = v\}| + |\{i : b_i = v\}| \leqslant \beta r$,*

there exist edge-disjoint paths in G, joining a_i to b_i, for each $i = 1, 2, \ldots, \kappa$. Furthermore, there is an $O(n^3)$ time randomized algorithm for constructing these paths.

This result is the best possible up to constant factors. The median distance between pairs of vertices in G is $\Omega(\log_r n)$. The need for $\beta < 1$ in this paper is a product of the proof. On the other hand, if one allows r to grow as n^γ for a small positive constant γ, then an argument in [5] shows that it is actually necessary.

The analogous problem of finding vertex-disjoint paths in random graphs is dealt with in [6].

2. Preliminaries

The paper contains a few unspecified absolute constants of which α above is the first. Exact values could be given but it is easier for us *and the reader* if we simply give the relations between them. New constants will be introduced as $C_0, \ldots$ without further

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

comment. Furthermore, specific constants have been chosen for convenience. We made no attempt to optimize them, and, in general, we only claim that inequalities dependent on n or r hold for n or r sufficiently large.

For a graph $G = (V, E)$ and $v \in V$ we let $d_G(v)$ denote the degree of v in G. We use $\delta(G)$ and $\Delta(G)$ to denote the smallest and largest degrees, respectively. For a set $S \subseteq V$ we let $\bar{S} = V \setminus S$ and define its neighbour set, $N_G(S)$, as

$$N_G(S) = \{v \in \bar{S} : \exists w \in S \text{ such that } \{v, w\} \in E\}.$$

For $S, T \subseteq V$ we let $e_G(S, T)$ denote the number of edges with one end in S and one end in T. For $S \subseteq V$, we use $G[S]$ to denote the subgraph of G induced by S.

The *Chernoff bounds* on the tails of the binomial random variable $\mathrm{Bin}(n, \theta)$ that we use are

$$\mathbf{Pr}(\mathrm{Bin}(n, \theta) \leqslant (1 - \epsilon)n\theta) \leqslant e^{-\epsilon^2 n\theta/2}, \tag{2.1}$$

$$\mathbf{Pr}(\mathrm{Bin}(n, \theta) \geqslant (1 + \epsilon)n\theta) \leqslant e^{-\epsilon^2 n\theta/3}, \tag{2.2}$$

valid for $0 \leqslant \epsilon \leqslant 1$.

3. Overview of the algorithm

Our algorithm divides naturally into the four phases sketched below.

Phase 0. Partition G into six edge-disjoint graphs $G_i = (V_i, E_i)$, $1 \leqslant i \leqslant 6$. Phase 1 will use only the graph G_1; Phase 2 will use only the graphs G_2, G_3, G_4 and G_5; and Phase 3 will use only the graph G_6. The partition is such that $V_1 = V$ but $V_2 = \cdots = V_6 \subseteq V$ with $|V_2| \geqslant n - \epsilon n$, where $\epsilon = \epsilon(r)$ is a small constant.

Phase 1. Choose a random set $Z = \{z_1, \ldots, z_{2\kappa}\}$ of 2κ points in V_2. Connect the end-points $\{(a_i, b_i) \mid i = 1, \ldots, \kappa\}$ to the newly chosen points in an arbitrary manner via edge-disjoint paths in G_1 using a flow algorithm. Let $\tilde{a}_i$ (resp. $\tilde{b}_i$) be the vertex connected to a_i (resp. b_i). The original problem is now reduced to finding edge-disjoint paths from $\tilde{a}_i$ to $\tilde{b}_i$ for each i. (This randomization was used in [7] and has its roots in Valiant's routing algorithm [21].)

Phase 2. We split this into parts (a), (b).

(a) For each z_i, $1 \leqslant i \leqslant 2\kappa$, we do a random walk of length

$$\tau_0 = \lceil 10 \log_r n \rceil \tag{3.1}$$

starting at z_i in G_2. The terminating end-point of this walk will be denoted by $\hat{a}_j$ if $z_i = \tilde{a}_j$ and by $\hat{b}_j$ if $z_i = \tilde{b}_j$. After we complete a walk we remove the edges of the walk from the corresponding graph.

(b) For each i in turn, we first do a random walk starting at $\hat{b}_i$ and terminating at b_i^* in G_4. Then we repeatedly do a certain type of random walk in G_3 starting from $\hat{a}_i$ until one of these walks ends at b_i^*. We keep the last walk as our path from $\hat{a}_i$ to b_i^* and remove all edges seen in these walks from G_3 and G_4, respectively.

Most pairs $(\tilde{a}_i, \tilde{b}_i)$ will be successfully connected in this phase. For such a pair, the final

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

path from a_i to b_i is the concatenation of the paths (after removing cycles if necessary) from a_i to $\tilde{a}_i$ and from b_i to $\tilde{b}_i$ found in Phase 1, the paths from $\tilde{a}_i$ to $\hat{a}_i$, from $\hat{a}_i$ to b_i^*, from b_i^* to $\hat{b}_i$ and from $\tilde{b}_i$ to $\hat{b}_i$ which are found in this stage. It is important in our analysis to ensure that random walks are done on graphs of high minimum degree. We use G_5 as a *backup* for ensuring that this is done.

Phase 3. At the end of Phase 2, w.h.p., there will be at most $n/(\ln n)^{C_0}$ pairs $(\tilde{a}_i, \tilde{b}_i)$ that have not been joined by paths. We use the algorithm of [7] to join them by edge-disjoint paths, using only the edges of G_6, and then construct the final paths from a_i to b_i as above.

To prove Theorem 1.1 it suffices to show that, for almost every $G_{n,r}$:

- Phases 0 and 1 will succeed for *all* choices of $a_1, \dots, b_\kappa$ and *almost every* choice of $z_1, \dots, z_{2\kappa}$;
- Phases 2 and 3 are successful for *almost every* choice of $z_1, \dots, z_{2\kappa}$ and *any* bijection $\{\tilde{a}_1, \dots, \tilde{a}_\kappa, \tilde{b}_1, \dots, \tilde{b}_k\} \longrightarrow \{z_1, \dots, z_{2\kappa}\}$.

Note that to prove these facts we have to consider only one experiment, namely choose $G_{n,r}$ at random, choose $a_1, \dots, b_\kappa$ arbitrarily and then $z_1, \dots, z_{2\kappa}$ at random. From this we can deduce that almost every $G_{n,r}$ is such that, for *all* choices of $a_1, \dots, b_\kappa$ and almost every choice of $z_1, \dots, z_{2\kappa}$, we can find edge-disjoint paths $a_i - \tilde{a}_i - \hat{a}_i - b_i^* - \hat{b}_i - \tilde{b}_i - b_i$ for $1 \leqslant i \leqslant \kappa$.

4. Detailed description of the algorithm

The input to our algorithm is a random graph $G_{n,r}$ and a set of pairs of vertices $\{(a_i, b_i) \mid i = 1, \dots, \kappa\}$ satisfying the premises of Theorem 1.1. The output is a set of κ edge-disjoint paths, $P_1, \dots, P_\kappa$, such that P_i connects a_i to b_i.

4.1. Phase 0

We start by partitioning G into six edge-disjoint graphs $G_i = (V_i, E_i)$, for $1 \leqslant i \leqslant 6$. Phase 1 will use only G_1; Phase 2 will use only G_2, G_3, G_4 and G_5; Phase 3 will use only G_6. The partition is such that $V_1 = V$ but $V_2 = \cdots = V_6 \subseteq V$ with $|V_2| \geqslant n - \epsilon n$, where $\epsilon = \epsilon(r)$ is a small constant.

In this construction, we use the notion of a k-core. The k-core of a collection of graphs $H_1, H_2, \dots, H_l$ on the same vertex set V is the largest subset of V that induces a subgraph of minimum degree at least k in each of $H_1, H_2, \dots, H_l$. It is unique and can be found by repeatedly removing vertices that have degree less than k in one of the l induced graphs. This continues until there are no vertices left or minimum degree k is achieved.

The algorithm SPLIT, shown opposite, starts by constructing preliminary versions of these graphs, denoted G_i' for $1 \leqslant i \leqslant 6$. Then edges and vertices are deleted from $G_2', \dots, G_6'$ in order to achieve certain minimum degree properties.

We will show later (Lemma 6.2) that w.h.p. this algorithm terminates with $|K| \geqslant n - \epsilon n$ where ϵ is small as a function of r. Note that SPLIT ensures that:

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

(1) **Algorithm** SPLIT
(2) **begin**
(3) Divide E into E'_i, $1 \leqslant i \leqslant 6$ by placing each edge of E independently with probability $1/6$ into each of E'_i for $1 \leqslant i \leqslant 6$.
(4) **for** $1 \leqslant i \leqslant 6$ set $G'_i \leftarrow (V, E'_i)$
(5) $K \leftarrow \lceil r/7 \rceil$-core of G'_1
(6) **for** $2 \leqslant i \leqslant 6$ set $G_i \leftarrow G'_i[K]$
(7) $K \leftarrow \lceil r/7 \rceil$-core of $G_2, \ldots, G_6$
(8) **for** $2 \leqslant i \leqslant 6$ $G_i = (V_i, E_i) \leftarrow G_i[K]$
(9) $E_1 \leftarrow E \setminus \bigcup_{i=2}^{6} E_i$; $G_1 \leftarrow (V, E_1)$
(10) **end** SPLIT

- the final graphs G_i, $i = 2, \ldots, 6$ have the same vertex set K;
- every $v \in K$ has degree at least $\lceil r/7 \rceil$ in each G_i, $1 \leqslant i \leqslant 6$;
- every $v \in K$ has degree at most $2r/7$ in each G_i, $2 \leqslant i \leqslant 6$;
- if $v \in V \setminus K$, then $d_{G_1}(v) = d_G(v) = r$.

4.2. Phase 1

Choose $z_1, z_2, \ldots, z_{2\kappa} \in V_2$ uniformly and randomly *without* replacement. Let Z denote the set $\{z_1, z_2, \ldots, z_{2\kappa}\}$. We are going to replace the problem of finding paths from a_i to b_i by that of finding paths from $\tilde{a}_i$ to $\tilde{b}_i$, where $\{\tilde{a}_1, \tilde{b}_1, \tilde{a}_2, \tilde{b}_2, \ldots, \tilde{a}_\kappa, \tilde{b}_\kappa\} = Z$. Let A denote the set $\{a_1, b_1, a_2, b_2, \ldots, a_\kappa, b_\kappa\}$.

We connect A to Z via edge-disjoint paths in the graph G_1 using network flow techniques. We construct a network as follows:

- each undirected edge of G_1 gets capacity 1;
- each $v \in V$ becomes a source of capacity $|\{i : a_i = v \text{ or } b_i = v\}|$ and each member of Z becomes a sink of capacity 1.

Then we find a flow from A to Z that satisfies all demands. Since the maximum flow has integer values, it decomposes naturally into $|A|$ edge-disjoint paths (together perhaps with some cycles). If a path joins a_i to $z \in Z$, then we let $\tilde{a}_i = z$. Similarly, if a path joins b_i to $z \in Z$, then we let $\tilde{b}_i = z$.

Thus Phase 1 finds edge-disjoint paths $W_i^{(1)}$ from a_i to $\tilde{a}_i$ and $W_i^{(6)}$ from $\tilde{b}_i$ to b_i, $1 \leqslant i \leqslant \kappa$, where the vertices $\tilde{a}_1, \tilde{b}_1, \tilde{a}_2, \tilde{b}_2, \ldots, \tilde{a}_\kappa, \tilde{b}_\kappa \in V_2$ are chosen uniformly at random without replacement. On the other hand there may be some difficult conditioning involved in the pairing of $\tilde{a}_i$ with $\tilde{b}_i$, $1 \leqslant i \leqslant \kappa$. We deal with this in Phase 2(a).

4.3. Phase 2

4.3.1. Algorithm GENPATHS**.** We construct edge-disjoint paths connecting $\tilde{a}_i, \tilde{b}_i$ for $1 \leqslant i \leqslant \kappa$ in this phase using random walks. A *random walk* on an undirected graph (or multigraph) $G = (V, E)$ is a Markov chain $\{X_t\}$ on V associated with a particle that moves from vertex to vertex according to the following rule. The probability of a transition from vertex v, of

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

degree d_v to a vertex w is $1/d_v$ if $\{v,w\} \in E$, and 0 otherwise. (For multigraphs, each edge out of a vertex is an equally likely exit; loops are counted as two exits.) Its stationary distribution, denoted by π or $\pi(G)$, is given by

$$\pi_v = \frac{d_v}{2|E|}.$$

A trajectory W of length τ is a sequence of vertices $[w_0, w_1, \dots, w_\tau]$ such that $\{w_t, w_{t+1}\} \in E$ for $1 \leqslant t < \tau$. The Markov chain induces a probability distribution on trajectories in the usual way. We use $\mathbf{Pr}_G^{(\tau)}(a,b)$ to denote the probability that a random walk in G of length τ starting at a terminates at b.

There is some possible conditioning involved in the pairing of $\tilde{a}_i$ with $\tilde{b}_i$. To break this we first do random walks $W_i^{(2)}, W_i^{(5)}$ starting from $\tilde{a}_i, \tilde{b}_i, 1 \leqslant i \leqslant \kappa$ in graph Γ_2. Here $\Gamma_j = (K_j, E_j), j = 2, \dots, 5$ denotes G_j after the deletion of some vertices and edges. We do these in the order $z_1, z_2, \dots, z_{2\kappa}$. The walks are sufficiently long that their end-points $\hat{a}_i, \hat{b}_i$

```
(1)  Algorithm GenPaths
(2)  begin
(3)      for i = 1 to 2κ do
(4)          K_2 ← ⌈r/10⌉-core of G_2, Γ_2 ← G_2[K_2].
(5)          Execute ConnectBack(K_2, z_i, z_i', φ(z_i), W_CB) fi
(6)          if j = φ(z_i) ∉ L then
(7)              Let θ = 2 if z_i = ã_j and θ = 5 if z_i = b̃_j. Construct
                   random walk W_j^(θ) of length τ_0 starting at z_i' in Γ_2.
                   The terminal end-point is â_j if z_i = ã_j and b̂_j
                   otherwise.
(8)              W_j^(θ) ← (W_CB, W_j^(θ)), G_2 ← G_2 \ E(W_j^(θ))
(9)          fi
(10)     od
(11)     for i = 1 to κ do
(12)         K_3 = K_4 ← ⌈r/10⌉-core of G_3, G_4, Γ_j ← G_j[K_j],  j = 3, 4.
(13)         Execute ConnectBack (K_3, â_i, â_i', i, W_CB^(3)) fi
(14)         Execute ConnectBack (K_4, b̂_i, b̂_i', i, W_CB^(4)) fi
(15)         if i ∉ L then
(16)             Construct random walk W_i^(4) of length τ_0 starting at
                   b̂_i' in Γ_4, the end-point is denoted by b_i^*.
(17)             Execute Walk(â_i', b̂_i', b_i^*, Γ_3, Γ_4), W_i^(3) ← Ŵ_l
(18)             G_3 ← G_3 \ E(Ŵ_j), j = 1, ..., l;  G_4 ← G_4 \ E(W_i^(4))
(19)             W_i^(3) ← (W_CB^(3), W_i^(3)),  W_i^(4) ← (W_CB^(4), W_i^(4))
(20)         fi
(21)     od
(22) end GenPaths
```

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

are essentially independent of $\tilde{a}_i$ and $\tilde{b}_i$. The edges of each such walk are deleted before the next walk starts. This keeps the walks edge-disjoint. This constitutes Phase 2(a).

Then for $i = 1, \ldots, \kappa$ we do the following. We construct a random walk $W_i^{(4)}$ starting at $\hat{b}_i$ in Γ_4 of length τ_0; the end-point is denoted by b_i^*. (This might seem unnecessary and we explain later why we do it.) Then, in step (20), we execute the subroutine WALK to generate a set of random walks $\hat{W}_1, \ldots, \hat{W}_l$ starting at $\hat{a}_i$ in Γ_3. The last walk $\hat{W}_l$ will end at b_i^* and be accepted as $W_i^{(3)}$. Finally, we delete $\hat{W}_1, \ldots, \hat{W}_l$ from G_3 and delete $W_i^{(4)}$ from G_2.

It is important for the above analysis to ensure that our random walks take place on graphs with sufficiently high minimum degree. The obvious reason for this is that we do not wish to isolate any vertex that needs to be an end-point of a walk. We also require that the walks take place on graphs that are strong expanders. For us, this only requires a high minimum degree. In general we ensure that graphs supporting the random walks have minimum degree at least $r/10$. We manage this by the simple artifact of removing vertices of low degree until the graph has none left. (We could empty a graph this way, but we show later that this is unlikely to happen.)

If the proposed start vertex v of a walk is of low degree then we try to *connect it back* to vertices of large degree (those in K_j) by a path in G_5. The terminal end-point of this walk is denoted by v'. (Note that this path is not of length τ_0: we expect it to be much shorter.) We use a subroutine CONNECTBACK for this purpose. We do not expect to succeed all the time and our failures are kept in a set L for later consideration.

The walk from a_i to b_i is then the concatenation of walks $W_i^{(t)}$, $t = 1, \ldots, 6$. These walks may each include a short walk W_{CB} at the beginning provided by CONNECTBACK.

For $z \in Z$ we use the notation $\phi(z) = i$ to indicate $z \in \{\tilde{a}_i, \tilde{b}_i\}$.

4.3.2. Subroutine CONNECTBACK**.** The purpose of CONNECTBACK is to connect a vertex x to vertices of large degree. (If x already has large degree then CONNECTBACK does nothing except to relabel x as x'.) All walks are done on vertices of large degree and in steps (3)–(6) we check that the start-point x has large degree in Γ_5. (Γ_5 is initially G_5 but

```
(1)   subroutine ConnectBack(K, x, x', i, W_CB)
(2)   begin
(3)       Γ_5 ← ⌈r/10⌉-core of G_5
(4)       if x ∉ Γ_5 then
(5)           L ← L ∪ {i}
(6)           exit
(7)       fi
(8)       Construct an edge-avoiding random walk W_CB starting at x in Γ_5
              until W_CB visits some vertex x' in K.
(9)       (This is a trivial walk if x ∈ K.)
(10)      G_5 ← G_5 \ W_CB
(11)  end ConnectBack
```

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

loses vertices and edges as the algorithm proceeds.) If not, we put it into L for later. We do an *edge-avoiding* random walk W_{CB} from x until a vertex x' of K is reached. Later we replace x by x'. By edge avoiding, we mean that one is not allowed to use the same edge more than once. We do this for technical reasons, but it does not seem unreasonable anyway. This is done in steps (7)–(10).

4.3.3. Subroutine WALK**.** To connect $\hat{a}_i$ and $\hat{b}_i$ after Phase 2(a) we first connect them (if necessary) to the $\lceil r/10 \rceil$-core of G_3, G_4. We then do a random walk $W_i^{(4)}$ in Γ_4 starting at $\hat{b}'_i$ and ending at b_i^*. We then try to join $\hat{a}'_i$ to b_i^*.

To join $\hat{a}'_i$ and b_i^*, the reader might expect us to choose a random walk from those with end-points $\hat{a}'_i, b_i^*$. The main problem with this is that the distribution of b_i^* may be significantly different from the steady state distribution of a walk from $\hat{a}'_i$ in Γ_3. If we choose a walk in this manner, then deleting it will condition the graph in a way that is complex to analyse, especially as we have to repeat the procedure κ times.

We overcome this by choosing a set of random walks and use rejection sampling to make the final walk have the correct distribution. There is still the complication that the b_i^* are chosen before we do the walks. This leads to the subroutine WALK described next. WALK$(\hat{a}'_i, \hat{b}'_i, b_i^*, \Gamma_3, \Gamma_4)$, shown below, generates a series of random walks of length τ_0 in Γ_3 starting from $\hat{a}'_i$. The last walk generated ends at b_i^*, which has the distribution

$$\hat{p}_v = P_{\Gamma_4}^{(\tau_0)}(\hat{b}'_i, v).$$

```
(1)  subroutine WALK(â'_i, b̂'_i, b*_i, Γ_3, Γ_4)
(2)  begin
(3)      p_v ← P^(τ0)_Γ3(â'_i, v) for v ∈ K_3
(4)      p̂_v ← P^(τ0)_Γ4(b̂'_i, v) for v ∈ K_4 (the distribution of b*_i)
(5)      p_min ← min{p_v : v ∈ K_3}
(6)      p̂_max ← max{p̂_v : v ∈ K_4}
(7)      Choose l from the geometric distribution with probability of success
             s = p_min/p̂_max.
(8)      for k from 1 to l − 1 do
(9)          Choose x_k from K_3(= K_4) according to
                 Pr(x_k = v) = (p_v − p̂_v p_min/p̂_max)/(1 − s).
(10)     od
(11)     x_l ← b*_i
(12)     for k from 1 to l do
(13)         Pick a walk Ŵ_k of length τ_0 in Γ_3 according to the
                 distribution on trajectories, conditioned on
                 start-point = â'_i and end point = x_k.
(14)     od
(15)     output Ŵ_1, Ŵ_2, ..., Ŵ_l
(16) end WALK
```

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

(There is the question as to why we do not try to construct a walk from $\hat{a}_i$ to $\hat{b}_i$. We need in WALK to generate an end-point from the same distribution as the target. This will not be so easy if $\hat{b}_i$ has dropped out of the $\lceil r/10 \rceil$-core.)

The somewhat strange method used to generate these walks will be further explained in Section 8.

The distributions p_v and $\hat{p}_v$ can be computed in $O(nm\tau_0)$ time by computing powers of the transition matrix, after which a random walk can be found in $O(n\tau_0)$ time. (For details see [7].) The analysis will show that, in the range of interest, w.h.p., s is bounded away from zero by a constant; hence the expected total running time of WALK is $O(nm\tau_0)$.

4.4. Phase 3

There is still the set L of pairs which have not been connected by paths. We will show later that w.h.p. $|L|$ is at most $n/(\log n)^{C_0}$. As such, these pairs can be dealt with by the algorithm of [7], using graph G_6.

5. Configuration model

5.1. Model

We first show (Lemma 5.2 below) that the graphs G_i, $i = 2, \ldots, 6$ are random given their degree sequences.

The simplest model for graphs with a fixed degree sequence is the *configuration model* of Bollobás [2], which is a probabilistic interpretation of the counting formula of Bender and Canfield [1]. Let $\mathbf{d} = \{d_1, d_2, \ldots, d_v\}$ denote a degree sequence, *block* $D_i = \{1, \ldots, d_i\} \times \{i\}$, for $1 \leqslant i \leqslant v$ and $D = \cup_{i=1}^{v} D_i$ be the set of *points*. Let $\Omega = \Omega(D)$ be the set of partitions of D into pairs. Such partitions are called *configurations*. If $F \in \Omega$ then the multigraph $M(F)$ is defined as follows: $V(M) = [v]$ and there is an edge $\{i, j\}$ for every pair in F of the form $\{(x, i), (y, j)\}$ (for some x and y). Assume F is chosen randomly from Ω. The salient properties of $M = M(F)$ (see [2]) are as follows.

Lemma 5.1.

(a) *If M is simple, then it is equally likely to be any simple graph with degree sequence* $\mathbf{d}$.

(b) $\mathbf{Pr}(M \text{ is simple}) = \exp\{-O(\mu^2/v^2)\}$ *where $\mu = |D|/2$ is the number of edges in M; hence $2\mu/v$ is the average degree of M.*

It therefore suffices to replace $G_{n,r}$ by a random configuration multigraph $G = M(F)$ where $\mathbf{d} = (r, r, \ldots, r)$. The corresponding random partition is denoted by F_0. Any property that holds with high probability in this model holds with high probability in $G_{n,r}$, since $\exp\{-O(\mu^2/v^2)\} = e^{-O(1)}$. In what follows the graphs $G_2, \ldots, G_6$ are multigraphs associated with configurations $F_2, \ldots, F_6$.

5.2. Random walks on configurations

To analyse random walks on the graphs G_i, we need to be sure they have the correct distribution.

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

Lemma 5.2. *For each $i = 2, \dots, 6$, the configuration F_i produced by the algorithm* SPLIT *is uniformly random given its degree sequence, that is, conditional on having degree sequence* $\mathbf{d}$*; F_i is equally likely to be any pairing of the corresponding $\sum d_i$ points.*

Proof. Let F' be obtained from F by including each pair of F with probability $1/6$. We first show that F' is a random configuration given its degree sequence. Notice that

$$\mathbf{Pr}(F' \mid \mathbf{d}) = \frac{\mathbf{Pr}(F')}{\mathbf{Pr}(\mathbf{d})} = \frac{\sum_F \mathbf{Pr}(F' \mid F)}{|\Omega|\,\mathbf{Pr}(\mathbf{d})}$$

and $\mathbf{Pr}(F' \mid F) = (1/6)^{(\sum d_i)/2}(5/6)^{(rn-\sum d_i)/2}$ if $F' \subset F$ and 0 otherwise. Suppose F_1 and F_2 are two configurations with the same degree sequence $\mathbf{d}$, and let $\mathscr{F}_1 = \{F \mid F_1 \subset F\}$ and $\mathscr{F}_2 = \{F \mid F_2 \subset F\}$. It suffices to show that $|\mathscr{F}_1| = |\mathscr{F}_2|$. Let E_0 be the set of pairs $F_1 \oplus F_2 = (F_1 \setminus F_2) \cup (F_2 \setminus F_1)$. Define mapping $f : \mathscr{F}_1 \to \mathscr{F}_2$ as follows. Given $F \in \mathscr{F}_1$, let $f(F) = [F \setminus (E_0 \cap F_1)] \cup (E_0 \cap F_2)$. It is easy to see that $f(F) \in \mathscr{F}_2$ and $f^{-1} : \mathscr{F}_2 \to \mathscr{F}_1$ is well defined as $f^{-1}(F) = [F \setminus (E_0 \cap F_2)] \cup (E_0 \cap F_1)$. Hence f is a bijection and $|\mathscr{F}_1| = |\mathscr{F}_2|$.

Thus (in an obvious notation) $F_1', \dots, F_6'$ are all random configurations. Deleting a set D_j and all t pairs containing a member of D_j from F_i' preserves randomness since every derived configuration arises from the same number of possible configurations F_i'. So the final configurations $F_2, \dots, F_6$ are random, given their degree sequences. □

We need to show that removing the edges of a random walk does not condition the pairings of the remaining points. We now consider the construction of a random configuration F. It is useful to think of F as being constructed sequentially.

It is important to observe that, for any $t > 0$, $F \setminus F_t$ is a random member of $\Omega(R_t)$. An important consequence of this observation is that, if we start with $M = M(F)$, then the multigraph obtained by removing from M the edges of a random walk W remains unconditioned. Indeed, we may imagine CONSTRUCT as performed in parallel with our walk W. Suppose our walk makes a transition from a vertex x and the current value of R_t in CONSTRUCT is R. The transition from x is equivalent to choosing a random member $u = u_t$ of D_x. If $u \in R$, then we perform one step of CONSTRUCT and pair u with a point $v = v_t \in R \setminus \{u\}$. If $v \in D_y$ for some y, then the walk makes a transition from x to y.

```
(1)  Algorithm CONSTRUCT
(2)  begin
(3)      F_0 ← ∅; R_0 ← W
(4)      for t = 1 to |W|/2 do
(5)          Choose u_t ∈ R_{t−1} arbitrarily.
(6)          Choose v_t randomly from R_{t−1} \ {u_t}.
(7)          F_t ← F_{t−1} ∪ {{u_t, v_t}}; R_t ← R_{t−1} \ {u_t, v_t}
(8)      od
(9)  output F
(10) end CONSTRUCT
```

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

If $u \notin R$ then v is the point already paired with u. Thus, since $F \setminus F_t$ is random, we see that removing from M the edges of a random walk results in a multigraph from a *random* configuration. The same argument can be applied to the edge-avoiding walk of ConnectBack. In the latter case we do not need to consider the case $u \notin R_t$.

In summary, we can conduct our analysis of Phase 3 always assuming that the graphs G_2–G_5 are random configuration multigraphs, given their degree sequences.

6. Analysis of Phase 0

We begin with some discussion of G.

Lemma 6.1. *With high probability, every subgraph H of G that is induced by a set of vertices S, $|S| \leqslant n_0 = 27n/(8e^6r^3)$, has at most $3|S|/2$ edges.*

Proof. Let $D_S = \bigcup_{i \in S} D_i = \{w_1, w_2, \ldots, w_{r|S|}\}$. Let us randomly pair up $w_1, w_2, \ldots$. At any stage, if w_i has not been paired with any of $w_1, w_2, \ldots, w_{i-1}$, and given the pairings of these points, the probability that w_i is paired with something in D_S is at most $|S|/n$. Hence

$$\begin{aligned}
\mathbf{Pr}(\exists S : |S| = s \leqslant n_0,\ S \text{ contains } 3s/2 \text{ edges}) &\leqslant \sum_{s=4}^{n_0} \binom{n}{s}\binom{rs}{3s/2}\left(\frac{s}{n}\right)^{3s/2} \\
&\leqslant \sum_{s=4}^{n_0} \left(\frac{ne}{s}\right)^{s}\left(\frac{2ers}{3n}\right)^{3s/2} \\
&= \sum_{s=4}^{n_0} \left(\frac{8e^5r^3s}{27n}\right)^{s/2} \\
&= O(n^{-2}). \qquad \square
\end{aligned}$$

The next lemma concerns the size of K at the end of Split.

Lemma 6.2. *If K is as defined in step (7) of algorithm* Split*, then w.h.p.*

$$|K| \geqslant (1 - e^{-r/800})n.$$

Proof. Fix $1 \leqslant i \leqslant 6$ and consider G_i' immediately after step (4). If $v \in V$ then $d_{G_i'}(v)$ has distribution $\mathrm{Bin}(r, 1/6)$, and hence at this point (2.1) implies

$$\mathbf{Pr}(d_{G_i'}(v) \leqslant \lfloor r/7 \rfloor) \leqslant \epsilon_1 = e^{-r/600}.$$

So, if $V_i = \{v \in V : d_{G_i'}(v) > \lfloor r/7 \rfloor\}$ then $\mathbf{E}(|V_i|) \geqslant (1-\epsilon_1)n$. Now the random variable $|V_i|$ is determined by the outcome of $rn/2$ random choices. Also, changing one such choice results in changing $|V_i|$ by at most 2. Applying the Azuma–Hoeffding martingale tail inequality [3, 16], we obtain

$$\mathbf{Pr}(|V_i| \leqslant \mathbf{E}(|V_i|) - t) \leqslant e^{-t^2/(rn)}.$$

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

Putting $t = \epsilon_1 n$ we see that w.h.p.

$$|V_i| \geqslant (1 - 2\epsilon_1)n, \qquad 1 \leqslant i \leqslant 6. \tag{6.1}$$

Now consider K at the end of step (5). Let $W_0 = V \setminus V_1$ and define a sequence $w_1, w_2, \ldots, w_k$ as follows. Suppose $W_j = W_0 \cup \{w_1, w_2, \ldots, w_j\}$ and w_{j+1} is an arbitrary vertex not in W_j but which has at least 2 neighbours in W_j. The sequence stops at index k when every vertex not in W_k has at most one neighbour in W_k. Now every vertex not in W_k has $d_{G_1'}(v) > \lfloor r/7 \rfloor$ and hence has degree at least $\lfloor r/7 \rfloor$ in $V \setminus W_k$. Thus $|K| \geqslant n - |W_k|$. We now show that w.h.p. $k < 6\epsilon_1 n$. If not, then

- $W_{6\epsilon_1 n}$ has at most $8\epsilon_1 n$ vertices,
- $W_{6\epsilon_1 n}$ contains at least $12\epsilon_1 n$ edges.

For sufficiently large r, this contradicts the conclusion of Lemma 6.1. Thus w.h.p., at the end of step (5),

$$|K| \geqslant (1 - 8\epsilon_1)n. \tag{6.2}$$

Now consider the execution of step (7). One can see that the value for K at the termination of this step is independent of the order in which vertices are deleted by the k-core algorithm. So assume that we start by deleting $\bigcup_{i=2}^{6}(V \setminus V_i)$. Inequalities (6.1) and (6.2) imply that w.h.p. after this $|K| \geqslant (1 - 18\epsilon_1)n$. We then argue, as above, that no more than $72\epsilon_1 n$ more vertices will be deleted by the k-core algorithm. The lemma follows for sufficiently large r. □

7. Analysis of Phase 1

In this section we show that if our input graph $G = (V, E)$ is $G_{n,r}$ then w.h.p., after we run SPLIT, we can find in G_1 edge-disjoint paths from a_i to $\tilde{a}_i$, and b_i to $\tilde{b}_i$, for $1 \leqslant i \leqslant \kappa$, for *any* choice of $a_1, \ldots, b_\kappa$ consistent with the premises of Theorem 1.1, and every choice for $\tilde{a}_1, \ldots, \tilde{b}_\kappa$.

Let A and Z be as defined in Section 4.2. For $S \subseteq V$, let

$$\alpha(S) = |\{i : a_i \text{ or } b_i \in S\}| \text{ and } \xi(S) = |S \cap Z|.$$

For sets $S, T \subseteq V$, let $e_{G_1}(S, T)$ denote the number of edges of G_1 with an end-point in S and the other end-point in T. It suffices to prove that

$$e_{G_1}(S, \bar{S}) \geqslant \xi(\bar{S}) - \alpha(\bar{S}), \qquad \forall S \subseteq V. \tag{7.1}$$

We can then apply a theorem of Gale [10] (see Bondy and Murty [4], Theorem 11.8) to deduce the existence of the required flow in G_1 for the successful run of Phase 2.

Lemma 7.1. *With high probability, $e_{G_1}(S, \bar{S}) \geqslant r|S|/288$ for all $S \subseteq V$, $|S| \leqslant n/2$.*

Proof. Assume first that $|S| \leqslant n_0$. From Lemma 6.1, we obtain

$$e_G(S, \bar{S}) \geqslant (\delta(G_1) - 3)|S| \geqslant (\lceil r/7 \rceil - 3)|S|.$$

So now assume $|S| > n_0$ and first consider $e_G(S, \bar{S})$. We claim that in the configuration

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

model this stochastically dominates Bin($r|S|/4$, $1/3$). Apply algorithm CONSTRUCT, always choosing $u_t \in D_v$, $v \in S$ whenever possible. This is definitely possible in the first $r|S|/4$ choices. For each such choice, the probability that $v_t \in D_w$, $w \in \bar{S}$ is at least $1/3$, since $|R_{t-1} \cap S| \leqslant r|S| \leqslant rn/2$ and $|R_{t-1} \cap \bar{S}| \geqslant rn/4$. But by (2.1) with $\epsilon = 1/2$,

$$\mathbf{Pr}\left(\mathrm{Bin}\left(\frac{r|S|}{4}, \frac{1}{3}\right) \leqslant \frac{r|S|}{24}\right) \leqslant e^{-r|S|/96}.$$

Thus

$$\begin{aligned}\mathbf{Pr}\left(\exists S : n_0 \leqslant |S| \leqslant n/2 \text{ and } e_G(S,\bar{S}) \leqslant \frac{r|S|}{24}\right) &\leqslant \sum_{s=n_0}^{n/2} \binom{n}{s} e^{-rs/96} \\ &\leqslant \sum_{s=n_0}^{n/2} \left(\frac{ne}{s} e^{-r/96}\right)^s \\ &= o(1)\end{aligned}$$

for sufficiently large r.

Assume then that $e_G(S,\bar{S}) \geqslant r|S|/24$ for $n_0 \leqslant |S| \leqslant n/2$. Given this, $e_{G_1}(S,\bar{S})$ stochastically dominates Bin($r|S|/24, 1/6$). Applying (2.1) again with $\epsilon = 1/2$,

$$\begin{aligned}\mathbf{Pr}\left(\exists S : n_0 \leqslant |S| \leqslant n/2 \text{ and } e_G(S,\bar{S}) \leqslant \frac{r|S|}{288}\right) &\leqslant \sum_{s=n_0}^{n/2} \binom{n}{s} e^{-rs/1152} \\ &\leqslant \sum_{s=n_0}^{n/2} \left(\frac{ne}{s} e^{-r/1152}\right)^s \\ &= o(1)\end{aligned}$$

for sufficiently large r. □

Now

$$\alpha(\bar{S}) = 2\kappa - \alpha(S) \geqslant 2\kappa - \beta r|S|,$$

and so

$$\xi(\bar{S}) - \alpha(\bar{S}) \leqslant |Z \cap \bar{S}| - 2\kappa + \beta r|S| \leqslant \beta r|S|.$$

Thus Lemma 7.1 verifies (7.1) for $|S| \leqslant n/2$ provided we choose $\beta < 1/288$. For $|S| > n/2$ we take $r \geqslant 288$ and use

$$e_{G_1}(S,\bar{S}) = e_{G_1}(\bar{S},S) \geqslant r|\bar{S}|/288 \geqslant |Z \cap \bar{S}| \geqslant |Z \cap \bar{S}| - 2\kappa + \alpha(S) = \xi(\bar{S}) - \alpha(\bar{S}),$$

and so Phase 2 succeeds w.h.p.

8. Analysis of Phase 2

8.1. Construction of $W_i^{(2)}, W_i^{(5)}$

For $v \in K$, let $d_{j,v}$ denote its degree in Γ_j, $j = 2,3,4$ and let $\pi_{j,v} = d_{j,v} / \sum_{w \in K} d_{j,w}$ be the steady state distribution of a random walk on Γ_j. These values change as the algorithm progresses, but because our walks always take place on an $\lceil \frac{r}{10} \rceil$-core, immediately prior

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

to any random walk in a Γ_j we have

$$\frac{r}{10} \leqslant d_{j,v} \leqslant \frac{2r}{7}. \tag{8.1}$$

Hence $v \in K$ implies

$$\frac{1}{3|K|} \leqslant \pi_{j,v} \leqslant \frac{3}{|K|}. \tag{8.2}$$

Let $\Gamma^{(i)}$ denote Γ_2 immediately before constructing the random walk starting at z_i, and let $K^{(i)}$ be its vertex set, the current value of K. Let $P^{(i)}$ denote the transition probability matrix of a random walk on $\Gamma^{(i)}$. Let $\lambda^{(i)}$ be the second largest eigenvalue of $P^{(i)}$. It is well known that the second eigenvalue determines the rate of convergence of a Markov chain to its steady state. An explicit form of this result was obtained by Sinclair and Jerrum [20]: if $P^{(t)}_{\Gamma^{(i)}}(u,v)$ denotes the probability that a random walk of length t in $\Gamma^{(i)}$ that starts at u will end at v, then

$$|P^{(t)}_{\Gamma^{(i)}}(u,v) - \pi^{(i)}_v| \leqslant \left(\lambda^{(i)}\right)^t \sqrt{\frac{\pi^{(i)}_v}{\pi^{(i)}_u}}, \tag{8.3}$$

where $\pi^{(i)}$ is the steady state distribution of a random walk on $\Gamma^{(i)}$.

We will argue later that w.h.p. throughout the algorithm

$$|K| \geqslant (1-\epsilon_K)n, \tag{8.4}$$

where

$$\epsilon_K = 8e^{-r/250000}.$$

We will argue that, as long as K is this large, vertices are being removed at a sufficiently slow rate that the overall loss from K is less than $\epsilon_K n$ w.h.p. Thus the specific assumption at each stage is that (8.4) holds at the present time. We do not in this way 'condition on the future'.

We argued in Section 5.2 that $\Gamma^{(i)}$ is random, given its degree sequence. According to Theorem 2 of [5] (which is adapted from [12]), the second largest eigenvalue $\lambda^{(i)}$ is with probability $1 - O(|K|^{-a}) = 1 - O(n^{-a})$ bounded by $\gamma/\sqrt{r}$, where γ depends only on a and a bound on the ratio of the maximum and the minimum degrees of $\Gamma^{(i)}$, which we can take to be 3 by (8.1). Taking $a = 4$ we have that γ is a constant independent of r. Using this in (8.3) we obtain that, conditional on an event of probability $1 - O(n^{-4})$,

$$|P^{(t)}_{\Gamma^{(i)}}(u,v) - \pi^{(i)}_v| \leqslant \frac{3\gamma^t}{r^{t/2}}. \tag{8.5}$$

So from (3.1), we see that for large r we have

$$|P^{(\tau_0)}_{\Gamma^{(i)}}(u,v) - \pi^{(i)}_v| \leqslant n^{-4}. \tag{8.6}$$

Let $\mathscr{E}^{(i)}$ be the intersection of the high probability events described in Lemmas 6.1, 6.2 and 7.1 plus (8.4), (8.6) for $\Gamma^{(1)}, \ldots, \Gamma^{(i)}$. We prove by induction on i that

$$\mathbf{Pr}(\mathscr{E}^{(i+1)} \mid \mathscr{E}^{(i)}) \geqslant 1 - O(n^{-3}). \tag{8.7}$$

This is clearly true for $i = 0$ and all we need to show is that (8.4) holds with the correct probability.

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

Fix $1 \leqslant i \leqslant \kappa$, let $s = 100 \ln n$ and $\hat{V}_2$ be the set of vertices in G_2 which are incident with at least $r/30$ edges of $W_1, \ldots, W_i$, where W_j denotes $W_j^{(2)}$, $1 \leqslant j \leqslant i$. For $S \subseteq K^{(0)}$ of size s, let $\mathscr{B}_S$ be the event that $S \subseteq \hat{V}_2$. We first show that, for all such S,

$$\mathbf{Pr}(\mathscr{B}_S \mid \mathscr{E}^{(i)}) \leqslant 2e^{-rs/250000}. \tag{8.8}$$

Let $S = \{v_1, \ldots, v_s\} \subset K^{(0)}$. Let $\mathscr{P} = \mathscr{P}(S)$ be the event that there exist $w \in K^{(1)} \setminus S$ and $t = r/2000$ distinct paths $P_1, P_2, \ldots, P_t$ from w to S of length $\leqslant k_0 = (\log_r n)/10$. If such paths exist then the graph induced by $w, S, P_1, P_2, \ldots, P_t$ contains $s + u$ vertices and $\geqslant s + u + t - 1$ edges for some $1 \leqslant u \leqslant (k_0 - 1)t + 1$. This is because the addition of each new path to S, w adds at least one more new edge than vertex. Thus,

$$\begin{aligned}
\mathbf{Pr}(\mathscr{P}) &\leqslant \sum_{u=1}^{(k_0-1)t+1} \binom{n}{s+u} \binom{2r(s+u)/7}{s+u+t-1} \left(\frac{3(s+u)}{n}\right)^{s+u+t-1} && (8.9)\\
&\leqslant \sum_{u=1}^{(k_0-1)t+1} \left(\frac{ne}{s+u}\right)^{s+u} \left(\frac{r(s+u)e}{n}\right)^{s+u+t-1}\\
&\leqslant \sum_{u=1}^{(k_0-1)t+1} \frac{(re^2)^{s+u+t}(s+u)^{t-1}}{n^{t-1}}\\
&\leqslant \frac{2(re^2)^{s+k_0t+1}(s+k_0t)^{t-1}}{n^{t-1}}\\
&\leqslant n^{-r/2500}\\
&= e^{-rs/250000}. && (8.10)
\end{aligned}$$

Explanation of (8.9). Working within the configuration model we choose the $s+u$ vertices in at most $\binom{n}{s+u}$ ways. The corresponding blocks contain at most $2r(s+u)/7$ points X. We then choose the set of lowest indices I of the $s+u+t-1$ pairs in at most $\binom{2r(s+u)/7}{s+u+t-1}$ ways. The last factor is an upper bound on the probability that each point in I chooses a partner in X.

Assume $\mathscr{P}$ does not occur. Let $Z_{i,S}$ denote the number of edges incident with vertices in S that are used by W_i and let $N_{i,S}$ be the number of visits to S by W_i. Consider ρ_S, the probability that a random walk W from $u \in S$ returns to S within τ_0 steps.

Lemma 8.1. *If $\mathscr{P}$ does not occur then $\rho_S \leqslant 1/20$.*

Proof. Let $\mathrm{dist}(v, S)$ denote the length of a shortest path in $\Gamma^{(i)}$ from v to S. Define $S_j = \{v \in K^{(i)} \setminus S : \mathrm{dist}(v, S) = j\}$ for $j = 1, \ldots, k_0 - 1$ and $S_{k_0} = \{v \in K^{(i)} \setminus S : \mathrm{dist}(v, S) \geqslant k_0\}$.

We define a random walk $X_0, X_1, \ldots$, where $X_0 = 0$ and $X_t = j$ if and only if W is at a vertex $v \in S_j$ after t steps ($S_0 = S$). Then, given $\bar{\mathscr{P}}$ and that the minimum degree in $\Gamma^{(i)}$ is $r/10$, for a random walk W,

$$\mathbf{Pr}(X_t = j + 1 \mid X_t = j) \geqslant 199/200, \qquad j = 1, \ldots, k_0 - 1, \tag{8.11}$$

$$\mathbf{Pr}(X_{t+1} = k_0 \mid X_t = k_0) \geqslant 199/200. \tag{8.12}$$

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

Now define another random walk $\tilde{X}_0 = 0, \tilde{X}_1, \ldots$, where

$$\mathbf{Pr}(\tilde{X}_{t+1} = \tilde{X}_t + \delta) = \begin{cases} \frac{199}{200} & \delta = 1 \\ \frac{1}{200} & \delta = -1, \quad \tilde{X}_t > 0, \quad \text{or} \quad \delta = 0, \quad \tilde{X}_t = 0. \end{cases}$$

Now for $u > 0$ we have

$$\mathbf{Pr}(\tilde{X}_u = 0) \leqslant \binom{u}{\lceil u/2 \rceil} \left(\frac{1}{200}\right)^{\lceil u/2 \rceil} \leqslant 50^{-\lceil u/2 \rceil},$$

since, if $\tilde{X}_u = 0$, at least $\lceil u/2 \rceil$ of the first u moves must be to the left, *i.e.*, $\delta \leqslant 0$. So,

$$\mathbf{Pr}(\exists\, u > 0 : X_u = 0) \leqslant \sum_{u=1}^{\infty} 50^{-\lceil u/2 \rceil} \leqslant 2 \sum_{u=1}^{\infty} 50^{-u} = \frac{2}{49}.$$

It follows from (8.11) and (8.12) that $\tilde{X}_j \leqslant X_j$ in distribution up to the first time u_0 that $X_u = k_0$. So

$$\mathbf{Pr}(\exists\, 0 < u \leqslant u_0 :\ X_u = 0) \leqslant \frac{2}{49}. \tag{8.13}$$

Next observe that, for any $u > 0$,

$$\mathbf{Pr}\left(X_{u+\frac{k_0}{10}} < \frac{k_0}{2} \mid X_u \geqslant \frac{k_0}{2}\right) \leqslant 2^{k_0/10} 200^{-k_0/20} = 50^{-k_0/20}$$

since, in order that $X_{u+\frac{k_0}{10}} < \frac{k_0}{2}$, there must be at least $\frac{k_0}{20}$ moves to the left.

Putting $u_i = u_0 + i\frac{k_0}{10}$, we see that

$$\mathbf{Pr}\left(\exists\, i,\, 1 \leqslant i \leqslant 10 :\ X_{u_i} < \frac{k_0}{2}\right) \leqslant 10 \times 50^{-k_0/20}.$$

Then observe that, if $X_u = 0$ for some $u_0 < u \leqslant \tau_0$, then we must have $u_i < \frac{k_0}{2}$ for some $1 \leqslant i \leqslant 10$. This together with (8.13) completes the proof of the lemma. □

Let $q_k = \mathbf{Pr}(N_{i,S} = k \mid \mathscr{E}^{(i)})$ for $k \geqslant 1$. We claim that, given $W_1, W_2, \ldots, W_{i-1}$,

$$q_k < \left(\frac{1000s \log_r n}{n}\right) 20^{-(k-1)}. \tag{8.14}$$

To prove (8.14) for $k = 1$, fix $\Gamma^{(i)}$ and let $h_v(t)$ be the probability that the walk is at v after t steps. For $t = 0$, we have two cases.

Case 1. Subroutine ConnectBack is not executed. In this case W_i starts from $z_i' = z_i$, which is a vertex chosen uniformly at random from $V_2 \setminus \{z_1, z_2, \ldots, z_{i-1}\}$. Notice that the execution of Phase 2 does not depend on the paths created in Phase 1. Therefore we can analyse Phase 2 without reference to Phase 1 and it is legitimate to consider z_i as chosen randomly in this way, independent of the state of $\Gamma^{(i)}$. So

$$h_v(0) \leqslant \frac{2}{|V_2|} \leqslant 10\pi_v^{(i)}. \tag{8.15}$$

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

Case 2. Subroutine ConnectBack is executed. In this case W_i starts from z_i', which is the end-point of a random walk in Γ_3. The random walks in Γ_3 only use unexposed edges. The end-point is chosen randomly from $K^{(i)}$ with probability at most $2\pi_v^{(i)}$: see Section 8.3.

So in either case $h_v(0) \leqslant 10\pi_v^{(i)}$. We can now inductively show that $h_v(t) \leqslant 10\pi_v^{(i)}$ for all $t \geqslant 0$. Using the stationarity equations, we have

$$h_v(t+1) = \sum_{w \in N_{\Gamma^{(i)}}(v)} \frac{h_w(t)}{d_w^{(i)}} \leqslant 10 \sum_{w \in N_{\Gamma^{(i)}}(v)} \frac{\pi_w^{(i)}}{d_w^{(i)}} = 10\pi_v^{(i)} \leqslant \frac{100}{n}. \tag{8.16}$$

The last inequality follows from (8.2) and (8.4). Hence, if $\hat{q}_1$ denotes the probability that $N_{i,S} \geqslant 1$ then

$$q_1 \leqslant \hat{q}_1 \leqslant \sum_{v \in S} \sum_{t=0}^{\tau_0} h_v(t) \leqslant \frac{1000 s \log_r n}{n}.$$

For $k \geqslant 2$, (8.14) follows from

$$q_k \leqslant \rho_S^{k-1} \hat{q}_1.$$

We prove next that w.h.p. the size of $\hat{V}_2$ is small. First (8.14) implies that

$$\begin{aligned} \mathbf{E}(e^{N_{i,S}} \mid \mathscr{E}^{(i)}, \bar{\mathscr{P}}) &\leqslant 1 + \sum_{k \geqslant 1} e^k \left(\frac{1000 s \log_r n}{n} \right) \rho_S^{k-1} && (8.17) \\ &\leqslant 1 + \frac{C_2 s \log_r n}{n}. && (8.18) \end{aligned}$$

Clearly $Z_{i,S} \leqslant 2N_{i,S}$. Thus, for any $t > 0$,

$$\begin{aligned} &\mathbf{Pr}\left(\sum_{j=1}^{i} Z_{j,S} \geqslant st \;\middle|\; \mathscr{E}^{(i)}, \bar{\mathscr{P}} \right) && (8.19) \\ &\leqslant e^{-st/2} \mathbf{E}\left(\exp\left\{ \sum_{j=1}^{i} N_{j,S} \right\} \;\middle|\; \mathscr{E}^{(i)}, \bar{\mathscr{P}} \right) \\ &\leqslant e^{-st/2} \left(1 + \frac{C_2 s \log_r n}{n} \right) \mathbf{E}\left(\exp\left\{ \sum_{j=1}^{i-1} N_{j,S} \right\} \;\middle|\; \mathscr{E}^{(i)}, \bar{\mathscr{P}} \right) \\ &\leqslant e^{-st/2} \exp\left\{ \frac{C_2 s \log_r n}{n} \right\} \mathbf{E}\left(\exp\left\{ \sum_{j=1}^{i-1} N_{j,S} \right\} \;\middle|\; \mathscr{E}^{(i-1)}, \bar{\mathscr{P}} \right) \frac{\mathbf{Pr}(\mathscr{E}^{(i-1)}, \bar{\mathscr{P}})}{\mathbf{Pr}(\mathscr{E}^{(i)}, \bar{\mathscr{P}})} \\ &\leqslant \exp\left\{ -\frac{st}{2} + i \frac{C_2 s \log_r n}{n} \right\} \frac{1}{\mathbf{Pr}(\mathscr{E}^{(i)}, \bar{\mathscr{P}})} \\ &\leqslant 2 \exp\left\{ -\frac{st}{2} + 2\kappa \frac{C_2 s \log_r n}{n} \right\} \\ &\leqslant 2 \exp\{ s(-t/2 + 2\alpha r C_2) \}. && (8.20) \end{aligned}$$

Take $t = r/30$ and $\alpha \leqslant (240 C_2)^{-1}$ so that $-t + 2\alpha C_2 r < -r/120$. Therefore

$$\mathbf{Pr}(S \subseteq \hat{V}_2 \mid \mathscr{E}^{(i)}, \bar{\mathscr{P}}) \leqslant 2e^{-rs/120}.$$

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

```
(1)  subroutine WALK1(â'_i, b̂'_i, b*_i, Γ3, Γ4)
(2)  begin
(3)      p_v ← P^(τ0)_Γ3(â'_i, v) for v ∈ K3
(4)      p̂_v ← P^(τ0)_Γ4(b̂'_i, v) for v ∈ K4 (the distribution of b*_i)
(5)      p_min ← min{p_v : v ∈ K3}
(6)      p̂_max ← max{p̂_v : v ∈ K4}
(7)      ℓ̂ ← 0
(8)      forever do
(9)          ℓ̂ ← ℓ̂ + 1
(10)         Pick a walk W̄_ℓ̂ of length τ0 in K3 according to the
               distribution on trajectories, conditioned on
               start-point = â'_i.
(11)         Let x̂_ℓ̂ be the terminal vertex of W̄_ℓ̂.
(12)         With probability p̂_{x̂_ℓ̂} p_min/(p_{x̂_ℓ̂} p̂_max) accept W̄_ℓ̂ and exitloop.
(13)     od
(14)     output W̄_1, W̄_2, ..., W̄_ℓ̂
(15) end WALK1
```

Thus

$$\mathbf{Pr}(S \subseteq \hat{V}_2 \mid \mathscr{E}^{(i)}) \leqslant \mathbf{Pr}(\mathscr{P} \mid \mathscr{E}^{(i)}) + \mathbf{Pr}(S \subseteq \hat{V}_2 \mid \mathscr{E}^{(i)}, \bar{\mathscr{P}}) \leqslant 2e^{-rs/250000},$$

proving (8.8).

So if $\epsilon_2 = e^{-r/250000}$ then it follows easily that w.h.p. $|\hat{V}_2| \leqslant 3\epsilon_2 n$. Indeed,

$$\begin{aligned}
\mathbf{Pr}(|\hat{V}_2| \geqslant 3\epsilon_2 n \mid \mathscr{E}^{(i)}) &= \mathbf{Pr}\left(\binom{|\hat{V}_2|}{s} \geqslant \binom{3\epsilon_2 n}{s} \,\middle|\, \mathscr{E}^{(i)}\right) \\
&\leqslant \frac{\mathbf{E}\left(\binom{|\hat{V}_2|}{s} \,\middle|\, \mathscr{E}^{(i)}\right)}{\binom{3\epsilon_2 n}{s}} \leqslant \frac{\binom{n}{s} 2\epsilon_2{}^s}{\binom{3\epsilon_2 n}{s}} \\
&\leqslant 2(e/3)^s = O(n^{-9}). \qquad (8.21)
\end{aligned}$$

Suppose that, besides $\hat{V}_2$, we also delete $v_1, v_2, \ldots, v_\ell$ from K_2 in step (4) of GENPATHS. We claim that $\ell \leqslant \epsilon_2 n$. To see this, note that v_i must have at least $\lceil r/7 \rceil - r/30 - r/10 \geqslant r/200$ neighbours in $\hat{V}_2 \cup \{v_1, \ldots, v_{i-1}\}$. Thus $\hat{V}_2 \cup \{v_1, \ldots, v_{i-1}\}$ has at least $\ell r/200$ edges. This contradicts Lemma 6.1 for $\ell > \epsilon_2 n$ and sufficiently large r, which proves that $|K| \geqslant n(1 - 4\epsilon_2)$ w.h.p. for the duration of Phase 2(a).

8.2. Construction of $W_i^{(4)}$ and analysis of WALK

The arguments we gave above apply to constructing and deleting $W_i^{(4)}$ in Γ_4 and we will focus on discussing the subroutine WALK. Consider a modification of WALK as shown above.

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

Lemma 8.2. *In* WALK1, $\hat{x}_{\hat{\ell}}$ *is chosen according to the distribution* $\hat{p}$.

Proof. The probability s that a walk is accepted at the last step in the loop is given by

$$s = \sum_{v \in K_3} p_v \frac{\hat{p}_v p_{\min}}{p_v \hat{p}_{\max}} = \frac{p_{\min}}{\hat{p}_{\max}}. \tag{8.22}$$

(Observe that $\hat{p}_{\max} \geqslant 1/|K_4| = 1/|K_3| \geqslant p_{\min}$.) Thus

$$\mathbf{Pr}(\hat{x}_{\hat{\ell}} = v) = \sum_{k=0}^{\infty} (1-s)^k p_v \frac{\hat{p}_v p_{\min}}{p_v \hat{p}_{\max}} = \hat{p}_v. \tag{8.23}$$

□

Hence $\bar{W}_{\hat{\ell}}$ is a random walk to a vertex chosen with distribution $\hat{p}$. Furthermore, by (8.2),

$$s \geqslant \sigma = \frac{1}{9}, \tag{8.24}$$

and therefore the expected number of walks generated is constant.

There is a minor problem in that we want to choose the end-points before we do the walks. This leads to the algorithm WALK described before. We now turn to its analysis.

Lemma 8.3. *Suppose that* b_i^* *is chosen from* K_3 *with distribution* $\hat{p}$. *Then the walks* $\bar{W}_1, \ldots, \bar{W}_{\hat{\ell}}$ *in* WALK1$(\hat{a}_i', \hat{b}_i', b_i^*, \Gamma_3, \Gamma_4)$, *and the walks* $\hat{W}_1, \ldots, \hat{W}_\ell$ *in* WALK$(\hat{a}_i', \hat{b}_i', b_i^*, \Gamma_3, \Gamma_4)$ *have the same distribution.*

Proof. Note first from the proof of Lemma 8.2 that $\hat{\ell}$ and ℓ have the same geometric distribution. Also, we have from Lemma 8.2 that $\hat{x}_{\hat{\ell}}$ and $x_\ell = b_i^*$ have the same distribution. Consider next that, for $v_1, v_2, \ldots, v_i \in K_3$,

$$\begin{aligned}
&\mathbf{Pr}(\hat{x}_1 = v_1, \ldots, \hat{x}_i = v_i \text{ and } \hat{\ell} > i) \\
&= \prod_{j=1}^{i} \left(\left(1 - \frac{\hat{p}_{v_j} p_{\min}}{p_{v_j} \hat{p}_{\max}} \right) p_{v_j} \right) = \prod_{j=1}^{i} \left(p_{v_j} - \frac{\hat{p}_{v_j} p_{\min}}{\hat{p}_{\max}} \right) \\
&= (1-s)^i \prod_{j=1}^{i} \left(\frac{p_{v_j} - \hat{p}_{v_j} p_{\min} / \hat{p}_{\max}}{1-s} \right) \\
&= \mathbf{Pr}(x_1 = v_1, \ldots, x_i = v_i \text{ and } \ell > i).
\end{aligned}$$

Thus $\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_{\hat{\ell}}$ and $x_1, x_2, \ldots, x_\ell$ have the same distribution. Finally, the lemma follows from the fact that the distribution of $\bar{W}_j$ conditional on $\hat{x}_j = v$ is clearly equal to that of $\hat{W}_j$ conditional on $x_j = v$. □

Now we need to show that there are not many vertices in G_3 having degrees less than $r/10$ after the ℓ walks are deleted. We shall refer to these walks as the ith *bundle*, B_i. We shall closely follow the line of proof used in the analysis in Section 8.1, with all the events now referring to Γ_3 rather than Γ_2. As before, the proof reduces to showing that, given

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

$\mathscr{E}^{(i)}$, after the removal of the bundles $B_1,\dots,B_i$ from G_3, the size of $\hat{V}_3$ (the set of vertices incident with at least $r/30$ edges of $B_1,\dots,B_i$) is small.

Again, let S be a fixed vertex set of size $100\ln n$, and q_k be the probability that S is visited by bundles $B_1,\dots,B_i$ exactly k times. We now evaluate the number of visits to S. Assume first that $\hat{a}_i \in S$. In this case this we will distinguish between *free* visits and *start* visits. S undergoes ℓ visits, as the start-point of all the walks in the bundle. All other visits to S are free visits. For $k \geqslant 2$, the $k-\ell$ free visits to S can be distributed among the ℓ walks in at most $\binom{k-1}{\ell-1}$ ways. We have

$$\mathbf{Pr}(k \text{ visits and } \hat{a}_i \in S) \leqslant \frac{100s}{n}\sum_{\ell=1}^{\infty}\binom{k-1}{\ell-1}\rho_S^{k-\ell}\sigma(1-\sigma)^{\ell-1} \leqslant \frac{C_4 s}{n}(1-\sigma+\rho_S)^{k-1}.$$

The term $\frac{100}{n}$ is arrived at as in (8.16).

Now consider the case $\hat{a}_1 \notin S$. Here we distinguish between *first* visits and free visits. A first visit is the first visit to S in a walk. Now we argue that there will be at least $k-\ell$ free visits and the probability of actually visiting S at least once in ℓ walks it at most $\frac{1000s\ell\log_r n}{n}$. We obtain

$$\begin{aligned}\mathbf{Pr}(k \text{ visits and } \hat{a}_i \notin S) &\leqslant \sum_{\ell=1}^{\infty}\frac{1000s\ell\log_r n}{n}\binom{k-1}{\ell-1}\rho_S^{k-\ell}\sigma(1-\sigma)^{\ell-1}\\ &\leqslant \frac{C_5 s\log_r n}{n}(1-\sigma+\rho_S)^{k-1}.\end{aligned}$$

So, putting $\xi = (\sigma-\rho_S)/2 > 1/20$, we see that

$$\begin{aligned}\mathbf{E}((1+\xi)^{N_{i,S}} &\leqslant 1+\sum_{k\geqslant 1}\frac{C_6 s\log_r n}{n}(1-\xi)^{k-1}\\ &\leqslant 1+\frac{C_6 s\log_r n}{n}.\end{aligned}$$

Arguing as from (8.19) to (8.20) we then obtain that

$$\mathbf{Pr}\left(\sum_{j=1}^{\kappa} Z_{j,S} \geqslant st \mid \mathscr{E}^{(I)},\bar{\mathscr{P}}\right) \leqslant 2(1+\xi)^{s(-t/2+2\alpha r C_6)}.$$

We then take $t = r/30$ as before and obtain

$$\mathbf{Pr}(S \subseteq \hat{V}_3 \mid \mathscr{E}^{(i)},\bar{\mathscr{P}}) \leqslant 2(1+\xi)^{-rs/120} < 2\epsilon_2.$$

We then argue, as before, that w.h.p. $|\hat{V}_3| \leqslant 3\epsilon_2 n$, and then use Lemma 6.1 to show that the $\lceil r/10\rceil$-core of G_3, G_4 is of size at least $n-\epsilon_K n$.

8.3. Analysis of ConnectBack

We use the subroutine ConnectBack to connect vertices that are not in K back to K. In this subroutine we construct random walks using only uncovered edges. This effectively removes the possibility of long walks that go back and forward over the same edges.

Recall the configuration model of Section 5.1. Assuming (8.4) and

$$|V(\Gamma_5)| \geqslant n - ne^{-r/800} - o(n) \tag{8.25}$$

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

at each step of one of the random walks in ConnectBack, the probability that the next vertex of the walk is not in K is at most $3\epsilon_K$. The degrees of Γ_5 satisfy (8.1) and these walks model Construct. Inequality (8.25) is justified by Lemma 6.2 and the fact that w.h.p. the $\lceil r/10\rceil$-core of Γ_5 contains all but $o(n)$ vertices of G_5: see the argument preceding (8.27) below. The main algorithm GenPaths will call the subroutine ConnectBack at most 4κ times, so that the total number of random walks executed by ConnectBack is at most 4κ. Let W_i be the ith walk executed by ConnectBack. As before, fix v and let $N_{i,v}$ be the number of visits to $v \in G_5$ by the walk W_i, $Z_{i,v}$ be the number of edges incident with v that are traversed by W_i, and let ℓ_i be the length of W_i. It is clear from the previous paragraph that $\mathbf{Pr}(\ell_i \geqslant (\ln n)^2) = O(n^{-A})$ for any constant $A > 0$. We can therefore consider a modified algorithm which terminates in failure if ever we find $\ell_i \geqslant (\ln n)^2$.

Note now that the probability that W_i starts at v is at most $10/n$, since v is either chosen uniformly from $K^{(0)}$ (step (6) in GenPaths) or it is the end-point of a long random walk (steps (15), (17)). Then

$$\begin{aligned}
\mathbf{Pr}(N_{i,v} = k) &= \sum_{\ell=k}^{(\ln n)^2} \mathbf{Pr}(N_{i,v} = k, \ell_i = \ell) \\
&\leqslant \sum_{\ell=k}^{(\ln n)^2} (3\epsilon_K)^{\ell-1}(1-3\epsilon_K)\binom{\ell}{k}\left(\frac{10}{n}\right)^k \qquad (8.26)\\
&\leqslant \frac{(3\epsilon_K)^{k-1}}{(1-3\epsilon_K)^k}\left(\frac{10}{n}\right)^k \\
&\leqslant \frac{10}{n^k}.
\end{aligned}$$

In (8.26) we have $(3\epsilon_K)^{\ell-1}(1-3\epsilon_K)$ as a bound for the probability that $\ell_i = \ell$. Then there are $\binom{\ell}{k}$ positions on the walk for the visits to v. Then $10/n$ bounds the probability that the walk makes the move to v at this point.

Note that if $v \in K$ then $N_{i,v} \leqslant 1$. Also, the upper bound we have calculated is valid, conditional on the previous history of the algorithm.

Suppose that there are $\kappa_1 \leqslant 4\kappa$ walks generated by ConnectBack in total. Then, for any $t > 0$,

$$\begin{aligned}
\mathbf{Pr}\left(\sum_{j=1}^{\kappa_1} N_{j,v} = t\right) &\leqslant \sum_{\ell_1+\cdots+\ell_{\kappa_1}=t} \prod_{j=1}^{\kappa_1}\left(\frac{10}{n}\right)^{\ell_j} \\
&\leqslant \binom{\kappa+t-1}{t}\left(\frac{10}{n}\right)^t \\
&\leqslant \binom{4\kappa+t-1}{t}\left(\frac{10}{n}\right)^t \\
&\leqslant \left(\frac{40e\alpha r}{\log_r n}\right)^t.
\end{aligned}$$

Let $\hat{V}_5$ be the set of vertices of G_5 that are incident with at least $r/30$ edges of $W_1, \ldots, W_i$.

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

Then

$$\begin{aligned}\mathbf{Pr}(v \in \hat{V}_5) &\leqslant \mathbf{Pr}\left(\sum_{j=1}^{\kappa} N_{j,v} \geqslant r/30\right)\\ &\leqslant \sum_{t=r/30}^{2r/7} \mathbf{Pr}\left(\sum_{j=1}^{\kappa} N_{j,v} = t\right) \leqslant 2\left(\frac{40e\alpha r}{\log_r n}\right)^{r/30}.\end{aligned}$$

So, if

$$\rho = n\left(\frac{40e\alpha r}{\log_r n}\right)^{r/40}$$

then, by the Markov inequality,

$$\mathbf{Pr}(|\hat{V}_5| \geqslant \rho) \leqslant \left(\frac{40e\alpha r}{\log_r n}\right)^{r/120} \to 0.$$

If $\hat{V}_5$ is this small then Lemma 6.1 implies that the $\lceil r/10\rceil$-core Γ_5 contains all but 2ρ vertices of G_5 (see the argument following (8.21)).

Thus

$$|L| \leqslant 2\rho \text{ w.h.p.} \tag{8.27}$$

as $L \subseteq G_5 \setminus \Gamma_5$.

9. Analysis of Phase 4

We join the pairs in L using the algorithm of [7]. The algorithm is capable of joining $\Omega(n/(\ln n)^c)$ distinct pairs for some constant $c > 0$, provided the graph has sufficient edge-expansion. Notice that $\tilde{a}_i, \tilde{b}_i$ are chosen as distinct vertices. For large r, G_6 will w.h.p. have ample expansion and 2ρ will be small enough for the algorithm.

References

[1] Bender, E. A. and Canfield, E. R. (1978) The asymptotic number of labelled graphs with given degree sequences. *J. Combin. Theory Ser. A* **24** 296–307.
[2] Bollobás, B. (1980) A probabilistic proof of an asymptotic formula for the number of labelled regular graphs. *Europ. J. Combinatorics* **1** 311–316.
[3] Bollobás, B. (1987) Martingales, isoperimetric inequalities and random graphs. *Coll. Math. Soc. J. Bolyai* **52** 113–139.
[4] Bondy, J. A. and Murty, U. S. R. (1976) *Graph Theory with Applications*, North-Holland.
[5] Broder, A. Z., Frieze, A. M., Suen, S. and Upfal, E. (1999) Optimal construction of edge-disjoint paths in random graphs. *SIAM J. Comput.* **28** 541–574.
[6] Broder, A. Z., Frieze, A. M., Suen, S. and Upfal, E. (1996) Optimal construction of vertex-disjoint paths in random graphs. In *Proc. 7th Annual ACM–SIAM Symposium on Discrete Algorithms*, pp. 261–268.
[7] Broder, A. Z., Frieze, A. M. and Upfal, E. (1994) Existence and construction of edge disjoint paths on expander graphs. *SIAM J. Comput.* **23** 976–989.
[8] Broder, A. Z., Frieze, A. M. and Upfal, E. (1999) Existence and construction of edge low congestion paths on expander graphs. *Random Struct. Alg.* **14** 87–109.

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press

[9] Frieze, A. M. (1998) Disjoint paths in expander graphs via random walks: a short survey. In *Proc. Random* '98, Vol. 1518 of *Lecture Notes in Computer Science*, Springer, pp. 1–14.
[10] Gale, D. (1957) A theorem on flows in networks. *Pacific J. Math.* **7** 1073–1082.
[11] Hochbaum, D. S. (1992) An exact sublinear algorithm for the max flow, vertex-disjoint paths, and communication problems on random graphs. *Operations Research* **40** 923–935.
[12] Kahn, J. and Szemerédi, E. (1989) On the second eigenvalue in random regular graphs: Section 2. In *Proc. 21st Annual ACM Symposium on Theory of Computing*, pp. 587–598.
[13] Leighton, T. and Rao, S. (1996) Circuit switching: a multicommodity flow based approach. In *Proceedings of a Workshop on Randomized Parallel Computing*.
[14] Leighton, T., Rao, S. and Srinivasan, A. (1998) Multi-commodity flow and circuit switching. In *Proc. Hawaii International Conference on System Sciences*.
[15] Leighton, T., Rao, S. and Srinivasan, A. (1999) New algorithmic aspects of the local lemma with applications to partitioning and routing. In *Proc. 10th Annual ACM–SIAM Symposium on Discrete Algorithms*, pp. 643–652.
[16] McDiarmid, C. J. H. (1989) On the method of bounded differences. In *Surveys in Combinatorics*, *Proc. 12th British Combinatorial Conference* (J. Siemons, ed.), Cambridge University Press, pp. 148–188.
[17] Peleg, D. and Upfal, E. (1989) Constructing disjoint paths on expander graphs. *Combinatorica* **9** 289–313.
[18] Robertson, N. and Seymour, P. D. (1995) Graph minors-XIII: The disjoint paths problem. *J. Combin. Theory Ser. B* **63** 65–110.
[19] Shamir, E. and Upfal, E. (1985) A fast construction of disjoint paths in networks. *Ann. Discrete Math.* **24** 141–154.
[20] Sinclair, A. and Jerrum, M. (1989) Approximate counting, uniform generation, and rapidly mixing Markov chains. *Inform. Comput.* **82** 93–133.
[21] Valiant, L. G. and Brebner, G. J. (1981) Universal schemes for parallel computation. In *Proc. 13th Annual ACM Symposium on Theory of Computing*, pp. 263–277.

https://doi.org/10.1017/S0963548300004284 Published online by Cambridge University Press