Chapter 30

An Efficient Algorithm for the Vertex-Disjoint Paths Problem in Random Graphs

Andrei Z. Broder* Alan M. Frieze† Stephen Suen‡ Eli Upfal§

Abstract

Given a graph $G = (V, E)$ and a set of 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 paths so found is vertex-disjoint. (The problem is $\mathcal{NP}$-complete for general graphs as well as for planar graphs. It is in $\mathcal{P}$ if the number of pairs is fixed.) Our model is that the graph is chosen first, *then* an adversary chooses the pairs of endpoints, subject only to obvious feasibility constraints, namely, all pairs must be disjoint, no more than a constant fraction of the vertices could be required for the paths, and not "too many" neighbors of a vertex can be endpoints.

We present a randomized polynomial time algorithm that works for almost all graphs; more precisely in the $G_{n,m}$ or $G_{n,p}$ models, the algorithm succeeds with high probability for all edge densities above the connectivity threshold. The set of pairs that can be accommodated is optimal up to constant factors. Although the analysis is intricate, the algorithm itself is quite simple and suggests a practical heuristic.

We include two applications of the main result, one in the context of circuit switching communication, the other in the context of topological embeddings of graphs.

1 Introduction

Given a graph $G = (V, E)$ with n vertices, and m edges, 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 paths so found is vertex-disjoint.

Finding vertex-disjoint paths in graphs is a basic computational question with a variety of algorithmic applications. For arbitrary graphs the related decision problem is $\mathcal{NP}$-complete, and it remains in $\mathcal{NP}$ even when the input is restricted to planar graphs [3]. The problem is in $\mathcal{P}$ if κ is fixed – Robertson and Seymour [8], through the graph minors technique. A more efficient algorithm is known only for the case $\kappa = 2$ [10]. For large k it is natural to look for sub-optimal algorithms, that is, algorithms that can solve the problem provided that k is within a certain factor of an obvious upper bound or within a fraction of the maximum achievable, but few results have been obtained so far.

The situation is similar for the edge-disjoint problem. However the problem is solvable in polynomial time on strong expanders (up to a polynomial-log factor of a trivial upper bound [7, 2]), and in random graphs (up to a constant factor of a trivial upper bound [1]). For a certain class of planar graphs, Kleinberg and Tardos [5] developed an algorithm that can find $O(1/\log n)$ fraction of the number of achievable paths. See references therein for other special cases.

Returning to the vertex-disjoint case, Hochbaum [4], and Shamir and Upfal [9], have studied the vertex-disjoint paths problem in the random graph models $G_{n,p}$ and $G_{n,m}$. Both papers show (using different techniques) that there exists a constant $C > 1$ such that for $p > \frac{C \log n}{n}$ (or $m > \frac{C}{2} n \log n$) a set of $O(\sqrt{n})$ disjoint pairs of vertices can be connected **whp**[1] by vertex disjoint paths. In the model used in these two papers the pairs of vertices are fixed before the edges of the random graphs are chosen. These results are relatively weak in two respects: (1) The number of paths is far from optimal: a random graph with $\Omega(n \log n)$ random edges has **whp** diameter $O(\log n)$, thus $O(\sqrt{n})$ paths use only a vanishing fraction of the vertices. (2) Since the set of pairs is fixed before the random edges are chosen, the result does not model the typical communication problem where the underlying graph is fixed.

Here we obtain a significantly stronger result: we show that with high probability a random graph is such

*Digital Systems Research Center, 130 Lytton Avenue, Palo Alto, CA 94301, USA. E-mail: broder@src.dec.com.

†Department of Mathematics, Carnegie-Mellon University, Pittsburgh, PA 15213. A portion of this work was done while the author was visiting IBM Almaden Research Center. Supported in part by NSF grants CCR9024935 and CCR9225008. E-mail: af1p@euler.math.cmu.edu.

‡Department of Mathematics, Carnegie-Mellon University, Pittsburgh, PA 15213.

§IBM Almaden Research Center, San Jose, CA 95120, USA, and Department of Applied Mathematics, The Weizmann Institute of Science, Rehovot, Israel. Work at the Weizmann Institute supported in part by the Norman D. Cohen Professorial Chair of Computer Science, by a MINERVA grant, and by the Israeli Academy of Science. E-mail: eli@wisdom.weizmann.ac.il.

[1] A sequence of events $\mathcal{E}_n$ is said to occur *with high probability*, abbreviated **whp**, if $\lim_{n\to\infty} \mathbf{Pr}(\mathcal{E}_n) = 1$

that our algorithm will be able to find vertex-disjoint paths for *any* set of pairs of endpoints in the graph, subject to constraints that are optimal up to constant factors. In other words in our model the graph is chosen first, then an adversary chooses the pairs of endpoints to be connected.

Our main result is formulated in the following theorem.

THEOREM 1.1. *Suppose that $G = G_{n,m}$ and $m \geq \frac{n}{2}(\ln n + \omega)$, where $\omega(n) \to \infty$. Let $d = 2m/n$. Then there exist two positive constants α, β such that* **whp** *there are vertex-disjoint paths connecting a_i to b_i for any set of pairs*

$$F = \{(a_i, b_i) \mid a_i, b_i \in V,\ i = 1, \ldots, \kappa\}$$

satisfying:

A1. *The pairs (a_i, b_i) for $i = 1, ..., \ldots, \kappa$, are disjoint;*

A2. *The total number of pairs, κ, is not greater than $\alpha n \ln d / \ln n$.*

A3. *For every vertex $v \in V$, no more than a β-fraction of its set of neighbors, $N(v)$, are prescribed endpoints, that is $|N(v) \cap (A \cup B)| \leq \beta |N(v)|$, where $A = \{a_i\}$ and $B = \{b_i\}$.*

Furthermore, these paths can be constructed by an (explicit) randomized algorithm in polynomial time.

Note that the three conditions above are implied, up to constant factors, by the following obvious constraints:
(A1.) A vertex can be the end-point of no more than one vertex-disjoint path.
(A2.) Most pairs of vertices in G are, **whp**, at distance $\Omega(\log n / \log d)$; hence in general, with n vertices we can connect at most $O(n \log d / \log n)$ pairs.
(A3.) If vertices are allowed to have all of their neighbors is $A \cup B$, then an adversary can choose a vertex u and all vertices at distance 2 from u as endpoints. Then, clearly u cannot be connected to a vertex at distance 3. However a weaker, more "global" condition might be possible here – this is an open problem. (On the other hand such a global condition might not be checkable in polynomial time.)

Thus, we get a tight (up to constant factors) characterization of the sets of pairs of vertices that can be connected by vertex disjoint paths in a random graph.

The techniques used in the proofs build on our results in [1], where we obtained an optimal construction of edge-disjoint paths in random graphs. Our construction was based on the analysis of random walks on certain subgraphs of a random graphs, through tight bounds on the eigenvalues of these subgraphs. Here we push this technique further to allow the analysis of vertex-disjoint paths construction (i.e., vertices rather then edges are eliminated from the graph during the construction of the paths). But the algorithm remains very simple, in essence:

1. From all endpoints in parallel go to a random vertex using flow techniques so that the set of pairs to be connected becomes a random set.

2. Remove from the graph all vertices used in the flow phase.

3. Connect each random pair in turn by a random path removing vertices as you go.

This of course suggests a very simple heuristic for practical problems.

One application of the disjoint paths problem is in the context of circuit switching communication, where the pairs of stations that request connection have to be assigned disjoint paths. Depending on the particular communication model, the circuit switching problem is translated to either an edge-disjoint or a vertex-disjoint paths problem. Communication algorithm are usually measured in terms of routing an arbitrary permutation request (that is, each node is the source or destination of one communication request). In the context of circuit switching the goal is to minimize the number of rounds required to realize an arbitrary permutation.

Our technique leads to an optimal (up to constant factors) algorithm for this question on random graphs. Let $\kappa = n/2$, and assume that n is even. A partition of a set S into $A_1, A_2, \ldots, A_k$ is an *equipartition* if $|A_i - A_j| \leq 1$ for all $i \neq j$.

COROLLARY 1.1. *Suppose that $G = G_{n,m}$, n is even, and $m = \frac{nd}{2} \geq \frac{n}{2}(\ln n + \omega)$, where $\omega(n) \to \infty$. Then there exists an absolute constant $\gamma > 0$ such that* **whp** *the following holds: For every partition $(a_1, b_1), (a_2, b_2), \ldots, (a_{n/2}, b_{n/2})$ of $[n]$ into $n/2$ pairs, a random equipartition of $[n/2]$ into $r = \lceil \gamma \ln n / \ln d \rceil$ sets $X_1, X_2, \ldots, X_r$ is such that for each X_i there exists in G a set of vertex disjoint paths $\mathcal{P}_i$ joining the pairs in X_i. Furthermore these paths can be constructed by a (explicit) randomized algorithm in polynomial time.*

This means that the number of rounds needed for routing an arbitrary permutation is $O(\ln n / \ln d)$ on almost all graphs.

Our second corollary concerns topological embeddings of graphs, a subject that has been extensively studied in the context of communication networks for parallel computers [6]. A graph $G = (V, E)$ contains a topological copy of $H = (W, F)$ if there exists an injection $f : W \mapsto V$ and paths $\mathcal{P} = \{P_e : e \in E\}$ such that (i) if $e = \{x, y\}$ then P_e is a path from x to y and (ii) the paths $\mathcal{P}$ are internally vertex disjoint.

COROLLARY 1.2. *Suppose that $G = G_{n,m}$, and $m = \frac{dn}{2} \geq \frac{n}{2}(\ln n + \omega)$, where $\omega(n) \to \infty$. Then there exist positive constants ρ and σ such that* **whp** *G contains a topological copy of any graph $H = (W, F)$ provided that*
(i) The size of the vertex set of H satisfies $|W| \leq n/2$.
(ii) The size of the edge set of H satisfies $|F| \leq \rho n \ln d / \ln n$.
(iii) The maximum degree in H is bounded by σd.

In this abstract we will present only an outline proof of Theorem 1.1 for the case $m \geq Cn \ln n$, where $C > 0$ is some large constant. This will allow us to give the main ideas of the proof without dealing with the technicalities caused by vertices of low degree. The proofs of the corollaries are given in Section 5.

2 Notations

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. The degree of a vertex v is denoted by $d_G(v)$.

For a set of vertices $S \subseteq V$ we denote its set of neighbors in T by $N(S : T) = N_G(S : T) = \{v \in T \setminus S : \exists w \in S \text{ with } (v, w) \in E\}$. We let $N(S) = N(S : V)$ and $N(v) = N(\{v\})$. Finally, the subgraph of G induced by $S \subseteq V$ is denoted $G[S] = (S, E_S)$ where E_S is the set of edges in E that have both endpoints in S.

3 The algorithm

We present a procedure, PATHS, that under the premises of the theorem, constructs the required paths in polynomial time. Our algorithm divides naturally into two phases: In Phase 1 (steps S1–S3) we aim to replace the given sets of endpoints A and B by two random sets $\tilde{A}$ and $\tilde{B}$; and in Phase 2 (steps S4 and S5) we connect these random endpoints using certain random walks.

We will view the paths notationally as sequences of vertices (as opposed to sequences of edges).

Algorithm PATHS

S1. Choose a random subset X_1 of $[n]$ by placing each $v \in V$ independently in X_1 with probability $1/3$. (Thus **whp** $|X_1| \approx n/3$.)
Let $X = X_1 \cup A \cup B$.

S2. Choose a random 2κ-subset $K = \{w_1, w_2, \ldots, w_{2\kappa}\}$ of X_1 by choosing w_i uniformly at random from $X_1 \setminus \{w_1, w_2, \ldots, w_{i-1}\}$.

S3. Find, using a network flow algorithm, 2κ vertex disjoint paths from $A \cup B$ to K in the graph $\Gamma = G[X]$.
For $1 \leq i \leq \kappa$ let $\tilde{a}_i$ (resp. $\tilde{b}_i$) denote the other endpoint of the path with one endpoint a_i (resp. b_i). Denote the path from a_i to $\tilde{a}_i$ by $W_i^{(1)}$ and the path from $\tilde{b}_i$ to b_i by $W_i^{(5)}$ for $1 \leq i \leq \kappa$.
Note that if $a_i \in K$ (resp. $b_i \in K$) the flow construction is simply $\tilde{a}_i = a_i$ (resp. $\tilde{b}_i = b_i$).

S4. Randomly partition $Y = [n] \setminus X$ into two sets Z_1, Z_2 by placing $v \in Y$ into Z_1 with probability $1/2$.
For $j = 1, 2, \ldots, 2\kappa$ construct a random walk $\hat{W}_j$ of length $\tau = \lceil 4 \ln n / \ln d \rceil$ starting at a random Z_1-neighbour $\tilde{w}_j$ of w_j in the graph $\Gamma_j = G[Y_j]$, where
$$Y_j = Z_1 \setminus \bigcup_{t=1}^{j-1} \hat{W}_t.$$
Let $\hat{w}_j$ be the endpoint of the walk $\hat{W}_j$. Now for $1 \leq i \leq \kappa$ if $\tilde{a}_i = w_j$ we let $\hat{a}_i = \hat{w}_j$ and $W_i^{(2)} = w_j + \hat{W}_j$ and if $\tilde{b}_i = w_k$ we let $\hat{b}_i = \hat{w}_k$ and $W_i^{(4)} = \hat{W}_k(\text{reversed}) + w_k$

S5. For $1 \leq i \leq \kappa$ construct, using the subroutine WALK described in section 4.2.2, a random walk $W_i^{(3)}$ of length τ, from a random Z_2-neighbour a_i^* of $\hat{a}_i$ to a random Z_2-neighbour b_i^* of $\hat{b}_i$ in the graph $\hat{\Gamma}_i = G[\hat{Y}_i]$ where
$$\hat{Y}_i = Z_2 \setminus \bigcup_{t=1}^{i-1} W_t^{(3)}.$$

S6. Output the paths
$$W_i = (W_i^{(1)}, W_i^{(2)}, W_i^{(3)}, W_i^{(4)}, W_i^{(5)}), \quad 1 \leq i \leq \kappa,$$
after cycles (if any) have been removed. (Cycles are possible only within a walk – the walks do not intersect.)
By construction W_i goes from a_i to b_i and these paths are vertex disjoint.

Further details of Steps S3, S4, and S5 are given in the next section.

4 Proof of Theorem 1.1 in the case $m \geq Cn \ln n$

4.1 Analysis of the flow phase: S1 - S3. Let $\mathrm{Bin}(n,p)$ be a binomial random variable with parameters n and p. We will use the well known bound

$$\mathbf{Pr}(|\mathrm{Bin}(n,p) - np| \geq \epsilon np) \leq 2e^{-\epsilon^2 np/3}, \qquad 0 \leq \epsilon \leq 1. \tag{4.1}$$

Let $\delta(G) = \min_{v \in V} d(v)$ and $\Delta(G) = \max_{v \in V} d(v)$. It is easy to show that if C is sufficiently large then **whp** $G = G_{n,m}$ has the following properties:

P1. $d/2 \leq \delta(G) \leq \Delta(G) \leq 2d$.

P2. For any set of vertices $S \subseteq V$ with $|S| \leq 20n/d$ the number of its neighbors satisfies $|N(S)| \geq d|S|/40$.

P3. For any two sets of vertices $S_1, S_2 \subseteq V$ with $S_1 \cap S_2 = \emptyset$ and $|S_1| \times |S_2| \geq 2n^2/d$ there is at least one $S_1 : S_2$ edge.

From now on we assume then that our graph G has these three properties.

Let $S = (A \cup B) \setminus K$ and $T = K \setminus (A \cup B)$. Suppose that S cannot be joined to T by $s = |S|$ vertex disjoint paths. Then (by Menger's theorem) there exists a set W, with $|W| = s - 1$ such that no component of $\Gamma \setminus W$ contains a vertex of both S and T. Let $L = W \setminus (S \cup T)$, $a = |S \cap W|$, and $b = |T \cap W|$, so that $|L| = s - a - b - 1$.

Let the components of $\Gamma \setminus W$ be $C_1, C_2, \ldots, C_r$ and suppose that $S \setminus W \subseteq D_1 = C_1 \cup C_2 \cup \cdots \cup C_\ell$ and $T \setminus W \subseteq D_2 = C_{\ell+1} \cup C_{\ell+2} \cup \cdots \cup C_r$. (We must have $\ell \geq 1$, otherwise $W \supseteq S$ which implies $|W| \geq s$. Similarly $r > \ell$.) Let $n_i = |D_i|$ for $i = 1, 2$, and let $\nu = \min\{n_1, n_2\}$. Condition P3 implies that

$$\frac{2n^2}{d} > \nu(n_1 + n_2 - \nu) \geq \nu(n_1 + n_2)/2 \geq \nu n/10.$$

Thus $\nu \leq 20n/d$. Assuming without loss of generality that $\nu = n_1$ we see from P2 that $|N(D_1)| \geq d\nu/40$.

Now for any set $D \subseteq V$, the size of its neighborhood $|N(D : X)|$ is distributed as $\mathrm{Bin}(|N(D)|, 1/3)$. Let $\mathcal{E}_1$ denote the event

$$\{\exists D \subseteq V : 1 \leq |D| \leq 20n/d \text{ and } |N(D : X)| \leq |N(D)|/6\}.$$

Then by P2 and (4.1),

$$\mathbf{Pr}(\mathcal{E}_1) \leq \sum_{k=1}^{20n/d} \binom{n}{k} \exp\left(-\frac{1}{8}\frac{dk}{60}\right) \leq \sum_{k=1}^{20n/d} \left(\frac{ne}{k} \exp\left(-\frac{d}{480}\right)\right)^k = o(1),$$

for large enough C. So we can assume $|N(D_1 : X)| \geq d\nu/240$. On the other hand $|N(D_1 : A \cup B)| \leq 2\beta d\nu$, by assumption A3 of Theorem 1.1 and P1. So D_1 must have $((1/240) - 2\beta)d\nu$ distinct neighbours in L. But $|L| < \nu$ and we have a contradiction for β sufficiently small.

Thus Steps S1-S3 will be successful **whp** on any graph with properties P1-P3.

4.2 Analysis of the random walks phase. A *random walk* on an undirected graph $G = (V, E)$ 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 degree d_v to a vertex w is $1/d_v$ if $\{v, w\} \in E$ and 0 otherwise. Its stationary distribution, denoted by π or $\pi(G)$, is given by $\pi_v = d_v/(2|E|)$. A trajectory W of length τ is a sequence of vertices $[w_0, w_1, \ldots, w_\tau]$ such that $\{w_t, w_{t+1}\} \in E$ for $1 \leq t < \tau$. The Markov chain induces a probability distribution on trajectories in the usual way. We use $P_G^{(\tau)}(u, v)$ to denote the probability that a random walk of length τ starting at u terminates at v.

It is well known that the second eigenvalue λ of the transition matrix determines the rate of convergence of a Markov chain to its steady state. A useful form of this result was obtained by Sinclair and Jerrum [11]:

$$|P_G^{(\tau)}(u, v) - \pi_v| \leq \lambda^\tau \sqrt{\frac{\pi_v}{\pi_u}}. \tag{4.2}$$

We will need the following result from [1]:

THEOREM 4.1. *Let* $\mathbf{d} = d_1, d_2, \ldots, d_n$ *be a degree sequence with maximum degree* $\Delta = o(n^{1/2})$ *and minimum degree* δ *such that* $\Delta/\delta < \theta$ *for some constant* $\theta > 0$. *Let* G *be chosen randomly from the set of* simple *graphs with degree sequence* $\mathbf{d}$. *Let* $0 < c < 1$ *be an arbitrary constant and* $\mathcal{G}$ *be the set of vertex induced subgraphs* H *of* G *which have degree at least* $c\delta$. *Let* $K > 0$ *be an arbitrary constant. Then with probability* $1 - O(n^{-K})$ every *graph* H *in* $\mathcal{G}$ *has second eigenvalue at most* $\gamma/\sqrt{\Delta}$ *where* $\gamma = \gamma(\theta, c, K)$.

To apply this theorem and (4.2) we will first deal with the condition $\Delta(G) = o(n^{1/2})$, then show that **whp** we can restrict our random walks to vertex induced sub-graphs H of $G_{n,m}$ with

$$\delta(H) > d/10. \tag{4.3}$$

Note that although Theorem 4.1 is couched in terms of random graphs with a fixed degree sequence, it applies to $G_{n,m}$ since given its degree sequence, $G_{n,m}$ is still randomly chosen.

We handle $\Delta(G) = o(n^{1/2})$ by assuming from now on that $d \le n^{1/10}$. If $d > n^{1/10}$ we simply choose a random set of $\lfloor n^{11/10}/2 \rfloor$ edges from the m previously chosen and delete the rest. Since for any initial d we have $\ln n^{1/10} \ge (\ln d)/10$ this assumption only affects the constants α, β in the statement of Theorem 1.1. We must now show that (4.3) holds during steps S4 and S5.

4.2.1 Analysis of Step S4. If C is large and α, β are small then **whp** for $i = 1, 2$ and any $v \in [n]$

$$d/4 \le |N(v : Z_i)| \le d/2. \tag{4.4}$$

Thus **whp** (4.3) holds for $H = \Gamma_1$. Consider the j'th walk of Step S4. For $v \in Y_j \cup X_1$, let $Z_{j,v}$ denote the number of vertices in $N(v : Y_j)$ that are visited by the j'th walk. Let $q_t = \mathbf{Pr}(Z_{j,v} = k)$. We claim that independent of previous walks,

$$q_k \le \frac{ab^{k-1}\log n}{d^{k-2} n \log d}, \tag{4.5}$$

for some constants $a, b > 0$.

To prove (4.5) for $k = 1$, let $h_v(t)$ be the probability that the walk is at a neighbour of $v \in Z_1$ at time t. We claim that

$$h_v(0) = O(d/n). \tag{4.6}$$

Indeed for $v \in Z_1$,

$$\begin{aligned}
&\mathbf{Pr}(\tilde{w}_j \in N(v : Z_1)) \\
&= \sum_{w' \in N(v:Z_1)} \mathbf{Pr}(\tilde{w}_j = w') \\
&= \sum_{w' \in N(v:Z_1)} \sum_{\xi \in N(w':X_1)} \mathbf{Pr}(\tilde{w}_j = w', w_j = \xi) \\
&= \frac{1}{|X_1|} \sum_{w' \in N(v:Z_1)} \sum_{\xi \in N(w':X_1)} \mathbf{Pr}(\tilde{w}_j = w' \mid w_j = \xi) \\
&= O(d/n),
\end{aligned}$$

assuming that for $1 \le j \le 2\kappa$

$$|N(w_j : Y_j)| \ge d/10.$$

This is true for $j = 1$ by (4.4) and for the remaining j's it is part of the induction, based on the use of (4.5) and (4.10).

Now given (4.6) we show inductively that for all $v \in V$, we have $h_v(t) = O(d\hat{\pi}_v^{(j)})$ where $\hat{\pi}^{(j)}$ is the stationary distribution of a random walk on Y_j. This follows from the stationarity equations: suppose that $h_v(t) \le cd\hat{\pi}_v^{(j)}$, then

$$\begin{aligned}
h_v(t+1) &= \sum_{w \in N(v:Y_j)} \frac{h_w(t)}{d_w^{(j)}} \\
&\le cd \sum_{w \in N(v:Y_j)} \frac{\hat{\pi}_w^{(j)}}{d_w^{(j)}} = cd\hat{\pi}_v^{(j)} = O(d/n).
\end{aligned}$$

Hence

$$q_1 \le \sum_{t=1}^{\tau} h_v(t) = O\left(\frac{d \log n}{n \log d}\right).$$

Fix Γ_j and for vertex v let ρ_v be the probability that a random walk of length τ from $N(v : Y_j)$ ever returns to $N(v : Y_j)$. We claim that

$$\rho_v = O(d^{-1}). \tag{4.7}$$

This gives (4.5) since

$$q_k \le (\rho_v)^{k-1} \sum_{t=1}^{\tau} h_v(t). \tag{4.8}$$

Let $\mathcal{D}_{t',t}$ for $2 \le t' \le t-2$ be the event that the walk is at distance 2 from $N(v : Y_j)$ at time t', at a neighbour of $N(v : Y_j)$ at times $t'+1, \dots, t-1$, and at $N(v : Y_j)$ at time t. Let $\mathcal{D}_0$ be the event that the walk never gets further than one away from $N(v : Y_j)$ before its first return to $N(v : Y_j)$. Then

$$\rho_v \le \mathbf{Pr}(\mathcal{D}_0) + \sum_{t'=2}^{\tau-2} \sum_{t=t'+2}^{\tau} \mathbf{Pr}(\mathcal{D}_{t',t}). \tag{4.9}$$

Assuming $d \le n^{1/10}$ it is easy to show that in $G_{n,m}$, **whp**

- No vertex is in more than one triangle.
- No pair of vertices are joined by more than two distinct paths of length at most three.

This implies that

$$\mathbf{Pr}(\mathcal{D}_0) = O(d^{-1})$$

and

$$\mathbf{Pr}(\mathcal{D}_{t-k,t}) = O(d^{-k}).$$

This proves (4.5) and therefore for any constant $c > 0$, as $n \to \infty$ we have

$$\begin{aligned}
\mathbf{E}(\exp(cZ_{j,v})) &\le 1 + \sum_{k=1}^{\infty} \frac{ab^{k-1}\log n}{d^{k-2} n \log d} e^{ck} \\
&\le 1 + \frac{2ae^c d \log n}{n \log d} \\
&\le \exp\left(\frac{2ae^c d \log n}{n \log d}\right).
\end{aligned}$$

It follows that for $t > 0$,

$$\begin{aligned}\Pr(\sum_{j=1}^{2\kappa} Z_{j,v} \geq t) \\ \leq \exp\left(-ct + 2\kappa\frac{2ae^c d\log n}{n\log d}\right) \\ \leq \exp(-ct + 4\alpha a e^c d) = O(n^{-2}),\end{aligned} \tag{4.10}$$

if $t = d/10$, and c and α are suitably chosen. Since the minimum degree in Γ_1 is at least $d/4$ **whp**, this proves that **whp** $\delta(\Gamma_{2\kappa}) > d/10$ and so (4.3) holds for $H = \Gamma_j$.

Now if α, β are small then $|Y_j| \geq n/4$ and $\pi^{(j)}$, the stationary distribution on Γ_j, is *almost uniform* i.e. there exist constants $a, b > 0$ such that **whp**

$$a/n \leq \pi_v^{(j)} \leq b/n. \tag{4.11}$$

Next let $\hat{p}^{(j)} = P_{\Gamma_j}^{(\tau)}(w_j, \cdot)$. Since we have chosen $\tau = \lceil 4\ln n/\ln d\rceil$, Theorem 4.1, with $\theta = 5$, $c = 1/10$, and $K = 2$, and equations (4.2) and (4.11), imply that $\hat{p}$ is also nearly uniform, in other words the points $\hat{w}_j$, are nearly uniformly distributed.

4.2.2 Analysis of Step S5. The analysis of Step S5 is complicated by the fact that we specify *both* endpoints of the walk. It helps for us to think in the following terms. At the start of Step S5 we have $\hat{W} = \{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_{2\kappa}\} \subseteq Z_1$. Furthermore, $\hat{w}_j$ has been chosen randomly from $Z_1 \setminus \{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_{j-1}\}$ in such a way that if $v \notin Z_1 \setminus \{\hat{w}_1, \hat{w}_2, \ldots, \hat{w}_{j-1}\}$ then the conditional distribution $\hat{p}^{(j)}$ defined by $\hat{p}^{(j)}(v) = \mathbf{Pr}(v = \hat{w}_j \mid \hat{w}_1, \hat{w}_2, \ldots, \hat{w}_{j-1})$ is nearly uniform. The analysis of [1] shows that algorithm WALK depicted in Figure 1 generates a random walk from a_i^* to b_i^* in $\hat{\Gamma}_i$.

subroutine WALK$(a_i^*, b_i^*, \hat{\Gamma}_i, \Gamma_j, w_j)$
begin
/* By construction, $w_j = \tilde{b}_i$, and by definition $V(\hat{\Gamma}_i) = \hat{Y}_i$ and $V(\Gamma_j) = Y_i$. */
$p_v \leftarrow P_{\hat{\Gamma}_i}^{(\tau)}(a_i^*, v)$ for $v \in \hat{Y}_i$
$\hat{p}_v \leftarrow \displaystyle\sum_{u\in N(v:G)\cap Y_j} \frac{P_{\Gamma_j}^{(\tau)}(\tilde{w}_j, u)}{|N(u:G)\cap \hat{Y}_i|}$ for $v \in Y_j$
/* $\hat{p}_v$ = the distribution of b_i^* */
$p_{\min} \leftarrow \min\{p_v : v \in \hat{Y}_i\}$;
  $\hat{p}_{\max} \leftarrow \max\{\hat{p}_v : v \in Y_j\}$
Choose r from the geometric distribution with probability of success $s = p_{\min}/\hat{p}_{\max}$
for k **from** 1 **to** $r-1$ **do**
  Choose x_k according to $\mathbf{Pr}(x_k = v) = (p_v - \hat{p}_v p_{\min}/\hat{p}_{\max})/(1-s)$
od
$x_r \leftarrow b_i^*$
for k **from** 1 **to** r **do**
  Pick a walk $\hat{W}_k$ of length τ in $\hat{\Gamma}_i$ according to the distribution on trajectories, conditioned on start point $= a_i^*$ and end point $= x_k$
od
output $\hat{W}_r$
end WALK

Figure 1: Algorithm WALK

Note that assuming (4.3) holds for $\hat{\Gamma}_i$ we have from Theorem 4.1 that p_v is nearly uniform and in fact s will be bounded below by some absolute constant $\sigma > 0$.

To complete the proof we have to show that (4.3) remains true also during S5. The proof is similar to the one in subsection 4.2.1 except that now there the complication that we start several walks from the same point. We have shown in [1] that the algorithm WALK1 depicted in Figure 2 generates a path with the same output distribution as WALK.

It is easier to analyse WALK1 rather than WALK because the former is just a random sequence of random walks. In order to placate some additional correlation we make the algorithm choose another random a_i^* for each of the r walks on $\hat{\Gamma}_i$.

Fix $v \in \hat{Y}_i$. Consider the r walks as a single walk of length $r\tau$, which restarts at a random $\hat{Y}_i$-neighbour a_i^* of $\hat{a}_i$ every τ steps. We distinguish between *start* visits to $N(v : \hat{Y}_i)$ when $a_i^* \in N(v : \hat{Y}_i)$ and *free* visits. The number of start visits, A_s, is binomial random variable with parameters r and $p_s \leq a/d$ for some constant $a > 0$. Furthermore one can show with a proof similar to that of (4.7) the following

LEMMA 4.1. *Assume $\hat{Y}_i$ satisfies (4.3) and consider a random walk W of length τ in $\hat{Y}_i$. If W starts at vertex in $N(v : \hat{Y}_i)$ (resp. not in $N(v : \hat{Y}_i) \cup \{v\}$) then the probability that the walk returns (resp. visits) $N(v : \hat{Y}_i)$ k times is $O(d^{-k})$.*

Now let A_f denote the number of free visits to $N(v : \hat{Y}_i)$ and let $\hat{q}_k = \mathbf{Pr}(A_f = k)$. Then $\hat{q}_k$ is at most

$$\sum_{t=1}^{\tau} h_v(t)\,\mathbf{Pr}(k-1 \text{ free visits to } v \text{ after } t \mid \hat{a}_i, X_t = v),$$

where $h_v(t) = O(d/n)$ is the unconditional probability

subroutine WALK1$(a_i^*, \hat\Gamma_i, \Gamma_j, w_j)$
begin
/* By construction, $w_j = \tilde{b}_i$, and by definition $V(\hat\Gamma_i) = \hat Y_i$ and $V(\Gamma_j) = Y_i$. */
$p_v \leftarrow P_{\hat\Gamma_i}^{(\tau)}(a_i^*, v)$ for $v \in \hat Y_i$

$$\hat p_v \leftarrow \sum_{u \in N(v:G)\cap Y_j} \frac{P_{\Gamma_j}^{(\tau)}(\tilde w_j, u)}{|N(u:G) \cap \hat Y_i|} \text{ for } v \in Y_j$$

/* $\hat p_v$ = the distribution of b_i^* */
$p_{\min} \leftarrow \min\{p_v : v \in \hat Y_i\}$;
  $\hat p_{\max} \leftarrow \max\{\hat p_v : v \in Y_j\}$
$r \leftarrow 0$;
forever do
  $r \leftarrow r + 1$;
  Pick a walk W_r^* of length τ according to the distribution on trajectories, conditioned on start point $= a_i^*$
  Let x_r^* be the endpoint vertex of W_r^*;
  With probability $\hat p_{x_r^*} p_{\min}/(p_{x_r^*} \hat p_{\max})$ accept W_r^* and **exitloop**
od
output W_r^*
end WALK1

Figure 2: Algorithm WALK1

that the t'th vertex of the walk is in $N(v : \hat Y_i)$.

Now given r, the $k-1$ free visits to $N(v : \hat Y_i)$ can be distributed among the r walks in at most $\binom{k+r-2}{r-1}$ ways. So in view of Lemma 4.1 we have

$$\mathbf{Pr}(k-1 \text{ free visits to } N(v:\hat Y_i) \text{ after } t \mid \hat a_i, X_t = v, r) \le \binom{k+r-2}{r-1}\left(\frac{c}{d}\right)^{k-1}\sigma(1-\sigma)^{r-1},$$

for some constant $c > 0$. Hence

$$\hat q_k \le \frac{c' d\tau}{n}\binom{k+r-2}{r-1}\left(\frac{c}{d}\right)^{k-1}\sigma(1-\sigma)^{r-1},$$

for some constant $c' > 0$.

So if q_k is the probability of k visits to $N(v : \hat Y_i)$ then

$$q_k \le \frac{c' d\tau}{n}\sum_{r=1}^{\infty}\sum_{\ell=0}^{k}\binom{r}{\ell}\left(\frac{a}{d}\right)^{\ell} \times \binom{k+r-2}{r-1}\left(\frac{c}{d}\right)^{k-\ell-1}\sigma(1-\sigma)^{r-1} \le a'\left(\frac{b'}{d}\right)^{k-2}\frac{\ln n}{n\ln d},$$

for some $a', b' > 0$ and we can proceed as in subsection 4.2.1.

(Note that visits to v itself are not a problem as they cause v to be deleted and we don't have to worry anymore about the size of its neighbourhood.)

5 Proof of the Corollaries

5.1 Proof of Corollary 1.1. Partition $[n/2]$ randomly into r sets $X_1, X_2, \ldots X_r$, with $r = \lceil \gamma \ln n/(\ln d)\rceil$ and γ to be chosen. Then let $Y_i = \{(a_j, b_j) : j \in X_i\}$ and $Z_i = \bigcup_{j\in X_i}\{a_j, b_j\}$. Each set Y_i is of size $\approx s = \lceil n\ln d/(\gamma \ln n)\rceil$. Each set Z_i is the union of two random subsets $\{a_j : j \in X_i\}$ and $\{b_j : j \in X_i\}$. Then $|N(v : Z_i)|$ is approximately the sum of two (correlated) binomials. Thus if $|N(v)| = \nu$ then

$$\mathbf{Pr}(|N(v:Z_i)| \ge \theta\nu|) \le 2\binom{s}{\theta\nu}\left(\frac{2\nu}{n}\right)^{\theta\nu} \le 2\left(\frac{2se\nu}{\theta\nu n}\right)^{\theta\nu} \le \left(\frac{n\ln d}{\gamma \ln n}\frac{6\nu}{\theta\nu n}\right)^{\theta\nu} = o(1/n),$$

for large enough γ. So we can assume

$$(5.12)\qquad |N(v:Z_i)| < \theta|N(v)|, \qquad \forall v \in V.$$

Thus we can apply Algorithm PATHS to compute the paths joining the pairs in each Y_i. (There are $o(\ln n)$ sets Y_i and the failure probability for PATHS is certainly $o(1/\ln n)$.)

5.2 Proof of Corollary 1.2. Choose a set S of $|W|$ vertices of G of degree $\ge d/2$ and define a 1-1 mapping $f : W \mapsto S$. For each $e = \{x, y\} \in F$ randomly choose distinct neighbours x_e, y_e of $f(x), f(y)$ in G. By Theorem 1.1, if ρ, σ are sufficiently small, then for most choices of $\{x_e, y_e\}$ we can find vertex disjoint paths joining x_e to y_e for all $e \in F$.

References

[1] A. Z. Broder, A. Frieze, S. Suen, and E. Upfal. Optimal construction of edge-disjoint paths in random graphs.

In *Preliminary version in Proceedings of the 5th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 603–612, 1994.

[2] A. Z. Broder, A. Frieze, and E. Upfal. Existence and construction of edge disjoint paths on expander graphs. *SIAM Journal of Computing*, 23:976–989, 1994. Preliminary version appeared in Proceedings of the 24th Annual ACM Symposium on Theory of Computing (1992).

[3] M. R. Garey and D. S. Johnson. *Computers and Intractibility: A Guide to the Theory of NP-Completeness.* W. H. Freeman and Company, New York, 1979.

[4] D. S. Hochbaum. An exact sublinear algorithm for the max flow, vertex-disjoint paths and communication problems on random graphs. *Operations Research*, 40:923–935, 1992.

[5] J. Kleinberg and E. Tardos. Approximations for the disjoint paths problem in high diameter planar networks. In *Proceedings of the 27th Annual ACM Symposium on Theory of Computing*, pages 26–35, 1995.

[6] F. T. Leighton. *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes.* Morgan Kaufmann Publishers, San Mateo, CA 94403, 1992.

[7] D. Peleg and E. Upfal. Constructing disjoint paths on expander graphs. *Combinatorica*, 9:289–313, 1989.

[8] N. Robertson and P. D. Seymour. Graph minors-XIII: The disjoint paths problem. To appear.

[9] E. Shamir and E. Upfal. A fast construction of disjoint paths in networks. *Annals of Discrete Mathematics*, 24:141–154, 1985.

[10] Y. Shiloach. A polynomial solution in the undirected two paths problem. *Journal of the ACM*, 27:445–456, 1980.

[11] A. Sinclair and M. Jerrum. Approximate counting, uniform generation, and rapidly mixing Markov chains. *Information and Computation*, 82:93–133, 1989.