Edge-Disjoint Paths in Expander Graphs

Alan M. Frieze
Department of Mathematical Sciences,
Carnegie Mellon University,
Pittsburgh, PA15213. *

Abstract

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 an r-regular expander graph G. We show that if G has sufficiently strong expansion properties and r is sufficiently large then *all* sets of $\kappa = \Omega(n/\log n)$ pairs of vertices can be joined. This is within a constant factor of best possible.

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 [17]. Peleg and Upfal [16] presented a polynomial time algorithm for the case where G is a (sufficiently strong) bounded degree expander graph, and $\kappa \leq n^\epsilon$ for a small constant ϵ that depends on the expansion property of the graph. This result has been improved and extended by Broder, Frieze, and Upfal [2, 3], Frieze [5], Leighton and Rao [12] and Leighton, Rao and Srinivasan [13, 14]: In these papers G has to be a (sufficiently strong) bounded degree expander and κ can grow as fast as $n/(\log n)^\theta$, where θ depends only on the expansion properties of the input graph, but is at least 2.

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 choice would require more edges than all the edges available. In the case of an r-regular expander, this absolute upper bound on κ is $O(n/\log n)$ (assuming r

*Supported in part by NSF grant CCR9530974. E-mail: `alan@random.math.cmu.edu`.

is independent of n). In this paper, we show that if G has sufficiently strong expansion properties and r is sufficiently large then *all* sets of $\kappa = \Omega(n/\log n)$ pairs of vertices can be joined. This therefore, is within a constant factor of the optimum. The precise definition of "sufficiently strong" is given after the theorem.

Theorem 1 *Let $G = (V, E)$ be an n-vertex, r-regular graph. Suppose that G is a sufficiently strong expander. Then there exist $\epsilon_1, \epsilon_2 > 0$ such that G has the following property: For all sets of pairs of vertices $\{(a_i, b_i) \mid i = 1, ..., \kappa\}$ satisfying:*

(i) $\kappa = \lceil \epsilon_1 rn/\log n \rceil$.

(ii) For each vertex v, $|\{i : a_i = v\}| + |\{i : b_i = v\}| \le \epsilon_2 r$.

There exist edge-disjoint paths in G, each of length $O(\log n)$, joining a_i to b_i, for each $i = 1, 2, \ldots, \kappa$. Furthermore, there is a polynomial time randomized algorithm for constructing these paths.
ϵ_1, ϵ_2 depend only on certain expansion parameters α, β, γ defined below. They do not depend on n or r.

The algorithm we use is based on the one used in Frieze and Zhao [7] which dealt with random r-regular graphs. In [7] we can take $\kappa = \lceil \epsilon_1 (r \log r) n/\log n \rceil$.

1.1 Preliminaries

We define expanders in terms of edge expansion (a weaker property than vertex expansion).

Let $G = (V, E)$ be a graph and let $n = |V|$. For $S \subset V$ let $\text{out}(S) = \text{out}_G(S)$ be the number of edges with one end-point in S and one end-point in $V \setminus S$, that is
$$\text{out}(S) = \left|\left\{\{u,v\} \mid \{u,v\} \in E, u \in S, v \notin S\right\}\right|.$$
Similarly,
$$\text{in}(S) = \left|\left\{\{u,v\} \mid \{u,v\} \in E, u, v \in S\right\}\right|.$$

A graph $G = (V, E)$ is a θ-expander, if for every set $S \subset V$, $|S| \le n/2$, we have $\text{out}(S) \ge \theta |S|$.

An r-regular graph $G = (V, E)$ is called an (α, β, γ)-expander if for every set $S \subset V$
$$\text{out}(S) \ge \begin{cases} (1-\alpha) r |S| & \text{if } |S| \le \gamma n \\ \beta r |S| & \text{if } \gamma n < |S| \le n/2 \end{cases}$$

We naturally assume that $\beta < 1 - \alpha$.

By "sufficiently strong" in Theorem 1, we mean that β, γ are arbitrary and α is sufficiently small. Then everything will work provided r is sufficiently large.

Since $2\text{in}(S) + \text{out}(S) = r|S|$ we see that in an (α, β, γ)-expander
$$\text{in}(S) \le \alpha r |S|/2 \qquad\qquad |S| \le \gamma n. \tag{1}$$

In particular random regular graphs and the (explicitly constructible) Ramanujan graphs of Lubotsky, Phillips and Sarnak [15] are (α, β, γ)-expanders. (See discussion in [2].)

The paper contains a few unspecified absolute constants. 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 $C_0, \ldots$, sometimes without further 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 neighbor set, $N_G(S)$, as

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

For $v \in V$ and $S \subseteq V$ we let $d_G(v, S) = |N_G(v) \cap S|$.

Let $\Phi_S = \text{out}(S)/|S|$ and let the (edge)-expansion $\Phi = \Phi(G)$ of G be defined by

$$\Phi = \min_{\substack{S \subseteq V \\ |S| \leq n/2}} \Phi_S.$$

We need an algorithm for splitting a strong expander into ten expander graphs. We could use the algorithm of [2] or [6]. The latter gives a better split and we arbitrarily choose to use it. $\epsilon > 0$ is a small constant. The expansion requirements for the algorithm are

$$\frac{r}{\log r} \geq 70\epsilon^{-2} \text{ and } \Phi \geq 40\epsilon^{-2} \log r, \tag{2}$$

which for us means

$$\beta \geq 40\epsilon^{-2} r^{-1} \log r. \tag{3}$$

The result we need from this paper (Theorem 2) is:

Theorem 2 *Suppose that (2), (3) hold and that G is an r-regular (α, β, γ)-expander. Then there is a randomised polynomial time algorithm ($O(n^2 \log \delta^{-1})$)[1] which with probability at least $1 - \delta$ constructs $E_1, E_2, \ldots, E_{10}$ such that the edge-expansion Φ_i of $G_i = (V, E_i)$ satisfies*

$$\Phi_i \geq (1 - \epsilon)\frac{\Phi}{10} - (\alpha + 2\epsilon)\, r,$$

for $i = 1, 2, \ldots, 10$.

2 Overview of the algorithm

Our algorithm divides naturally into the three phases sketched below.

[1] The paper only claims $n^{O(\log r)}$ expected time but changing the definition of X_0 in [6] to deal with smaller $|S|$ easily yields this improvement

Phase 0: Partition G into ten edge-disjoint graphs $G_i = (V, E_i)$, $1 \leq i \leq 10$. Phase 1 will use only the graphs G_1 and G_2; Phase 2 will use only the graphs G_3, G_4 and G_5; and Phase 3 will use only the graphs $G_6 - G_{10}$.

Phase 1: Choose two random sets $\tilde{A}, \tilde{B}$ of κ vertices in V. Connect the endpoints $A = \{a_i : i = 1, 2, \ldots, \kappa\}$ to the newly chosen points $\tilde{A}$ in an arbitrary manner via edge-disjoint paths in G_1 using a flow algorithm. Similarly, connect the endpoints $B = \{b_i : i = 1, 2, \ldots, \kappa\}$ to the newly chosen points $\tilde{B}$, this time using G_2. 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.

Phase 2: We split this into parts (a),(b),(c).
(a) At this point we want $\tilde{a}_i$, $i = 1, 2, \ldots, \kappa$ to be a random ordering of a random set of vertices and so we randomly re-order $a_1, a_2, \ldots, a_\kappa$ to ensure this. We do the same with $\tilde{b}_1, \tilde{b}_2, \ldots, \tilde{b}_\kappa$. We then randomly generate $x_1, x_2, \ldots, x_\kappa$ from V with replacement.

(b) For each i in turn, we connect $\tilde{a}_i$ to x_i by a path of minimum length in G_3. We remove the edges of this path from G_3.

(c) For each i in turn, we connect $\tilde{b}_i$ to x_i by a path of minimum length in G_4. We remove the edges of this path from G_4.

Most pairs $(\tilde{a}_i, \tilde{b}_i)$ will be successfully connected via x_i in this phase. For such a pair, the final path from a_i to b_i is the concatenation of the paths indicated as follows

$$a_i - \tilde{a}_i - x_i - \tilde{b}_i - b_i$$

It is important in our analysis to ensure that random walks are done on subgraphs which are expander graphs. We use G_5 as a *backup* for ensuring that this is done.

Phase 3: At the end of Phase 2, there will with probability $\geq 1/2$, be at most $n/(\log n)^4$ pairs $(\tilde{a}_i, \tilde{b}_i)$ which have not been joined by paths. We use the algorithm of [5] to join them by edge disjoint paths, using only the edges of $G_6 - G_{10}$, and then construct the final paths from a_i to b_i as above.

To prove Theorem 1 it suffices to show that:

- Phases 0 and 1 will succeed for *all* choices of $a_1, \ldots, b_\kappa$ and *almost every* choice of $\tilde{A}, \tilde{B}$.
- Phases 2 and 3 are successful for *almost every* choice of $\tilde{A}, \tilde{B}$ and *any* bijection $\tilde{A} \to \tilde{B}$

3 Detailed description of the algorithm

The input to our algorithm is a sufficiently strong (α, β, γ)-expander graph G and a set of pairs of vertices $\{(a_i, b_i) \mid i = 1, ..., \kappa\}$ satisfying the premises of Theorem 1. The output is a set of κ edge-disjoint paths, $P_1, \ldots, P_\kappa$ such that P_i connects a_i to b_i.

3.1 Phase 0.

We start by partitioning G into ten edge-disjoint graphs $G_i = (V, E_i)$, for $1 \leq i \leq 10$. We use the algorithm SPLIT of Theorem 2. We take $\epsilon = \alpha$ in the theorem and assume that $\beta \gg \alpha$. Thus each G_i satisfies

$$\Phi_i = \Phi(G_i) \geq \beta_0 r, \tag{4}$$

$$\beta_0 r \leq \delta(G_i) \leq \Delta(G_i) < r, \tag{5}$$

where

$$\beta_0 = \frac{\beta}{10} - 4\alpha > 3\alpha > 0. \tag{6}$$

3.2 Phase 1.

Choose $\tilde{A}, \tilde{B} \subseteq V$ uniformly and randomly *without* replacement. 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 $\tilde{b}_i$.

We connect A to $\tilde{A}$ 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 $|\{i : a_i = v\}|$ and each member of $\tilde{A}$ becomes a sink of capacity 1.

Then we find a flow from A to $\tilde{A}$ 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 $z \in \tilde{A}$, then we let $\tilde{a}_i = z$.

We carry out a similar construction involving B and $\tilde{B}$ in G_2.

Thus Phase 1 finds edge-disjoint paths $W_i^{(1)}$ from a_i to $\tilde{a}_i$ and $W_i^{(4)}$ from $\tilde{b}_i$ to b_i, $1 \leq i \leq \kappa$, 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 $\tilde{b}_i$, $1 \leq i \leq \kappa$. We deal with this in Phase 2(a).

3.3 Phase 2.

3.3.1 Algorithm GenPaths.

We (try to) construct edge-disjoint paths connecting $\tilde{a}_i, x_i, \tilde{b}_i$ for $1 \leq i \leq \kappa$. For $1 \leq i \leq \kappa$ we try to connect $\tilde{a}_i$ to x_i in graph Γ_3 by a shortest path $W_i^{(2)}$. Here $\Gamma_j = (V_j, F_j), j = 3, 4, 5$ denotes G_j after the deletion of some vertices and edges. We construct these paths in the order $\tilde{a}_1, \tilde{a}_2, \ldots, \tilde{a}_\kappa$. The edges of each such path are deleted before the next path is constructed. This keeps the paths edge-disjoint. This constitutes Phase 2(b).

In Phase 2(c) we use the same ideas and Γ_4 to join $\tilde{b}_1, \tilde{b}_2, \ldots, \tilde{b}_\kappa$ to $x_1, x_2, \ldots, x_\kappa$ by a shortest path $W_i^{(3)}$.

It is important for the above analysis to ensure that the construction of any shortest path takes place on a graph $\Gamma = (K, F)$ which is an expander. We can ensure this by keeping the degrees of vertices in the Γ_j close to their degree in G_j. This may involve deleting some vertices after a walk. We use the routine REMOVE to do this.

If the proposed start vertex v of a path on Γ does not lie in K then we try to *connect it back* to K by a path in Γ_5. The terminal endpoint of this walk is denoted by v'. 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 catenation of walks $W_i^{(t)}$, $t = 1, \ldots, 4$. These walks may each include a short walk W_{CB} at the beginning provided by CONNECTBACK.

Algorithm GENPATHS
begin
  $\Gamma_i \leftarrow G_i$, $i = 3, 4, 5$.
  for $i = 1$ **to** κ **do**
  Execute REMOVE(Γ_3)
  Execute CONNECTBACK($V_3, \tilde{a}_i, \tilde{a}_i', i, W_{CB}$)
  if $i \notin L$ **then**
    Execute CONNECTBACK($V_3, x_i, x_i', i, W_{CB}$)
    if $i \notin L$ **then**
      Construct a shortest path $W_i^{(2*)}$ from $\tilde{a}_i'$ to x_i' in Γ_3.
      $W_i^{(2)} \leftarrow (W_{CB}, W_i^{(2*)}), \Gamma_3 \leftarrow \Gamma_3 \setminus E(W_i^{(2*)})$
    fi
  od
  for $i = 1$ **to** κ **do**
  Execute REMOVE(Γ_4)
  Execute CONNECTBACK($V_4, \tilde{b}_i, \tilde{b}_i', i, W_{CB}$)
  if $i \notin L$ **then**
    Execute CONNECTBACK($V_4, x_i, x_i'', i, W_{CB}$)
    if $i \notin L$ **then**
      Construct a shortest path $W_i^{(3*)}$ from $\tilde{b}_i'$ to x_i'' in Γ_4.
      $W_i^{(3)} \leftarrow (W_{CB}, W_i^{(3*)}), \Gamma_4 \leftarrow \Gamma_4 \setminus E(W_i^{(4*)})$
    fi
  od
end GENPATHS

3.3.2 Subroutine REMOVE

The purpose of REMOVE is to delete vertices which might prevent a graph (or graphs) from being an expander. In GENPATHS we apply REMOVE to Γ_3, or Γ_4. In CONNECTBACK we apply REMOVE to Γ_5.

In Step 4 we remove the set of vertices R_0 which have so far lost more than $\beta_0 r/4$ edges through the deletion of shortest paths. We then iteratively (Steps 5–12) remove vertices which have at least $\beta_0 r/4$ neighbours among previously removed vertices. We therefore see that for $t = 3, 4$

$$v \in V_t \text{ implies } d_{\Gamma_t}(v) \geq d_{G_t}(v) - \beta_0 r/2 \geq \beta_0 r/2. \tag{7}$$

1. **Algorithm** REMOVE(Γ_t)
2. **begin**
3. $\quad R_0 = \{v \in V_t : \ d_{\Gamma_t}(v) < d_{G_t}(v) - \beta_0 r/4\}$.
4. $\quad \ell \leftarrow 0$.
5. $\quad$ **begin**
6. $\quad\quad \bar{R}_\ell \leftarrow V_t \setminus R_\ell$.
7. $\quad\quad d \leftarrow \max_v\{d_{\Gamma_t}(v, R_\ell) : \ v \in \bar{R}_\ell\}$.
8. $\quad\quad$ **if** $d \leq \beta_0 r/4$ **terminate** REMOVE, otherwise
9. $\quad\quad R_\ell \leftarrow R_\ell \cup \{w\}$; $V_t \rightarrow V_t \setminus \{w\}$ where $w \in \bar{R}_\ell$ is such that $d_{\Gamma_t}(w, R_\ell) = d$.
10. $\quad\quad \ell \leftarrow \ell + 1$
11. $\quad\quad$ **goto** 6.
12. $\quad$ **end**
13. **end** REMOVE

We can see from (7) that throughout the algorithm

$$\Phi_{\Gamma_t} \geq \Phi_t - \beta_0 r/2 \geq \beta_0 r/2 \qquad \text{for } t = 3, 4. \tag{8}$$

Indeed, (7) implies that for $S \subseteq V_t$ we have

$$\text{out}_{\Gamma_t}(S) \geq \text{out}_{G_t}(S) - \beta_0 r|S|/2 \geq (\Phi_t - \beta_0 r/2)|S|.$$

3.3.3 Subroutine ConnectBack.

The purpose of ConnectBack is to connect a vertex x by a random walk to a set $K = V_3$ or V_4 of vertices of large degree in a particular subgraph. (If x already has large degree then ConnectBack does nothing except to relabel x as x'). All walks are done on vertices V_5 and in Step 3 we check that the start point x lies in V_5. If not, we put i into L, where $x = \tilde{a}_i, \tilde{b}_i, x_i$ or x''_i. Edge disjoint paths for the pairs $(a_i, b_i), i \in L$ are found in Phase 3. Let

$$\omega = \lceil \log\log n \rceil^2.$$

We do a random walk W_{CB} from x until we reach K or make ω steps. In the latter case we add the corresponding i to L.

subroutine CONNECTBACK(K, x, x', i, W_{CB})
begin
  if $x \in K$ **then** $x' \leftarrow x$ **exit fi else**
  Execute REMOVE(Γ_5)
  if $x \notin V_5$ **then** $L \leftarrow L \cup \{i\}$ **exit fi else**
  Do a random walk W_{CB} starting at x in Γ_5, until K is reached or ω steps have been taken.
  In the latter case $L \leftarrow L \cup \{i\}$ and we **exit else**
  $\Gamma_6 \leftarrow \Gamma_5 \setminus W_{CB}$
end CONNECTBACK

3.4 Phase 3.

There is still the set L of pairs $(\tilde{a}_i, \tilde{b}_i)$ which have not been connected by paths. We will show later that with probability at least $1 - o(1)$, $|L|$ is at most $n/(\log n)^4$. As such, these pairs can be dealt with by the algorithm of [5], using graphs G_6 – G_{10}.

4 Analysis of Phase 1

In this section we show that if (4) holds and

$$\beta_0 r \geq 1 \text{ and } \epsilon_2 \leq \beta_0 \tag{9}$$

then after we run SPLIT, we can find edge-disjoint paths from a_i to $\tilde{a}_i$ in G_1 and edge disjoint paths from b_i to $\tilde{b}_i$ in G_2, for $1 \leq i \leq \kappa$, for *any* choice of $a_1, \ldots, b_\kappa$ consistent with the premises of Theorem 1, and every choice for $\tilde{a}_1, \ldots, \tilde{a}_\kappa, \tilde{b}_1, \ldots, \tilde{b}_\kappa$.

Let A and $\tilde{A}$ be as defined in Section 3.2. For $S \subseteq V$, let

$$\alpha(S) = |\{i : \ a_i \in S\}| \text{ and } \xi(S) = |S \cap \tilde{A}|.$$

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

$$e_{G_1}(S, \bar{S}) \geq \xi(\bar{S}) - \alpha(\bar{S}), \qquad \forall S \subseteq V. \tag{10}$$

Given (10), the existence of the required flow in G_1 is a special case of a theorem of Gale [8] (see Bondy and Murty [1] Theorem 11.8). In which case we see that (10) implies a successful run of Phase 2.

Now

$$\alpha(\bar{S}) = \kappa - \alpha(S) \geq \kappa - \epsilon_2 r|S|$$

and so

$$\xi(\bar{S}) - \alpha(\bar{S}) \leq |\tilde{A} \cap \bar{S}| - \kappa + \epsilon_2 r|S| \leq \epsilon_2 r|S|.$$

Thus (4) verifies (10) for $|S| \leq n/2$ provided we have $\epsilon_2 \leq \beta_0$. For $|S| > n/2$ we have $\Phi_1 \geq 1$ and then

$$e_{G_1}(S,\bar{S}) = e_{G_1}(\bar{S},S) \geq \Phi_1|\bar{S}| \geq |\tilde{A} \cap \bar{S}| \geq |\tilde{A} \cap \bar{S}| - \kappa + \alpha(S) = \xi(\bar{S}) - \alpha(\bar{S})$$

and so Phase 1 succeeds with respect to $A, \tilde{A}$. The same argument applies to $B, \tilde{B}$. To ensure these paths are of length $O(\log n)$ we can solve a minimum cost maximum flow problem as indicated in Kleinberg and Rubinfeld [11].

5 Analysis of Phase 2

Lemma 1 *Throughout the algorithm*

$$|V_j| \geq (1-\gamma_0)n, \qquad j = 3,4,$$

where

$$\gamma_0 = \frac{\beta_0\gamma}{10.}$$

Proof: First consider V_3. We know from (8) that Γ_3 is a $(\beta_0 r/2)$-expander throughout the execution of Phase 2. We can use the strong edge-expansion of Γ_3 to prove some vertex-expansion and conclude the diameter of Γ_3 is at most $\tau_1 = \lceil 2\log_{1+\beta_0/2} n\rceil + 1$. Indeed, in a θr-expander, every set S, $|S| \leq n/2$, has at least $\theta|S|$ neighbours. Thus the total number of edges in the paths that are removed from G_3 is $\leq \kappa\tau_1$. Hence the vertices B_3 of G_3 which are incident with $\beta_0 r/4$ edges of these paths satisfy

$$|B_3| \leq \frac{4\kappa\tau_1}{\beta_0 r} \leq \frac{\gamma_0 n}{3}$$

provided

$$\epsilon_1 \leq \frac{\beta_0\gamma_0}{25}\log\left(1+\frac{\beta_0}{2}\right). \tag{11}$$

Let $X = \{x_1, x_2, \ldots,\}$ be the remaining vertices removed by REMOVE. We claim that if $|B_3| \leq \frac{\gamma_0}{3}n$ then $|X| \leq 2|B_3| \leq \frac{2\gamma_0}{3}n$ implying that $|V_3| \geq (1-\gamma_0)n$.

Indeed, if $X_i = \{x_1, x_2, \ldots, x_i\}$ then $X_i \cup B_3$ has $i + |B_3|$ vertices and contains at least $i\beta_0 r/4$ edges. The existence of x_i, $i = 2|B_3|$ contradicts (1) with $S = X_i \cup B_3$. So,

$$\mathrm{in}(S) \geq |B_3|\beta_0 r/2 \geq |S|\beta_0 r/6 > |S|\alpha r/2$$

using (6). This proves the lemma for V_3 and the argument for V_4 is identical. □

Our next task is to bound the size of the set L of pairs of vertices which are left to Phase 3. For this we need to establish some facts about random walks on graphs.

5.1 Random Walks

A *random walk* on an undirected graph $G = (V, F)$ 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 $1/(2d)_v$ if $\{v, w\} \in E$ and 0 otherwise. The particle stays at v with probability 1/2. This removes the possibility of periodicity and allows us to use the conductance bound of Jerrum and Sinclair. Its stationary distribution, denoted by π, is given by $\pi(v) = \frac{d_v}{2|E|}$ for $v \in V$.

Let P be the transition matrix of the associated Markov chain. Let λ be the second largest eigenvalue of P. According to Jerrum and Sinclair [18]

$$\lambda \leq 1 - \frac{\Psi^2}{2} \tag{12}$$

where Ψ denotes the *conductance* of a random walk on G.

Here,

$$\begin{aligned}
\Psi &= \min_{\pi(S)\leq 1/2} \frac{1}{\pi(S)} \sum_{\substack{v\in S \\ w\notin S}} \pi(v)P(v,w) \\
&\geq \min_{\pi(S)\leq 1/2} \frac{1}{\pi(S)} \sum_{\substack{v\in S \\ w\notin S}} \frac{d_v}{\Delta|V|} \cdot \frac{1}{2d_v} \\
&= \min_{\pi(S)\leq 1/2} \frac{\text{out}(S)}{2\Delta|V|\pi(S)} \\
&\geq \frac{\Phi\delta}{2\Delta^2}.
\end{aligned} \tag{13}$$

Another fact we will need is

$$|P^t(v,w) - \pi(w)| \leq \sqrt{\frac{\Delta}{\delta}}\lambda^t. \tag{14}$$

A proof of this can be found for example in [18].

Now consider our random walks. Arguing as in Lemma 1 we first note that since $\kappa\omega = o(n)$ we will have $|V_5| = n - o(n)$ throughout the algorithm.

The minimum and maximum degrees of Γ_5 will satisfy

$$\beta_0 r/2 \leq \delta \leq \Delta \leq r.$$

Thus Γ_5 has at least $(1 - o(1))\beta_0 rn/4$ edges and then for sufficiently large n, the steady state for a random walk on Γ_5 will always satisfy

$$\frac{\beta_0}{2n} \leq \pi(v) \leq \frac{2}{\beta_0(1 - o(1))n} \qquad \text{for all } v \in V_j,$$

where π denotes the steady state distribution of a random walk on Γ_5.

From (8) and (13) we see that the conductance Ψ satisfies

$$\Psi \geq \frac{\beta_0^2}{8}. \tag{15}$$

Applying (12) we see that the second eigenvalue λ of a random walk on Γ_5 always satisfies

$$\lambda_5 \leq 1 - \frac{\beta_0^4}{64}.$$

Using this in (14) we obtain that

$$|P_{\Gamma_j)}^{(t)}(u,v) - \pi_j(v)| \leq e^{-t\beta_0^4/64}. \tag{16}$$

So we see that if

$$\tau_0 = 256\beta_0^{-4} \ln n$$

then

$$|P_{\Gamma(j)}^{(\tau_0)}(u,v) - \pi_j(v)| = O(n^{-4}). \tag{17}$$

We also need a large deviation result. This can be taken from the works of Dinwoodie [4], Gillman [9] and Kahale [10]. We quote the consequences of Theorem 2.1 of [9]: Let q be the distribution of the start vertex of a random walk on a graph G. Let S be a fixed set of vertices of G. Let Y denote the number of visits to S in the first t steps.

$$\mathbf{Pr}(Y - t\pi(S) \leq -u) \leq \left(1 + \frac{(1-\lambda)u}{10t}\right) N_q e^{-(1-\lambda)u^2/(20t)}, \tag{18}$$

where

$$N_q = \left(\sum_{v \in V} \frac{q(v)^2}{\pi(v)}\right)^{1/2}.$$

5.2 Analysis of CONNECTBACK

Fix $j = 3$ or 4. Consider all calls to connect back a vertex to V_j. Let $L = L_5 \cup L_7$ where L_θ consists of the indices added to L in Step θ of CONNECTBACK.

To probabilistically bound $|L_5|$ we first bound the expected value of the number $M_j, j = 3,4$ of vertices which are incident with 5 or more walks in executions of Step 6 of CONNECTBACK which connecting back to V_j. Fix a $j = 3$ or 4 and enumerate these walks as $W_1, W_2, \ldots, W_m$, $m \leq 2\kappa$. Here walk W_i can have one or zero vertices if the proposed start vertex z satisfies $z \in V_j$ or $z \notin V_5$. Then for $c = 9/\beta_0^2$,

$$\begin{aligned}\mathbf{E}(M_j) &\leq \sum_{i_1,\ldots,i_5,v} \mathbf{Pr}(W_{i_t},\, t = 1,\ldots,5 \text{ go through } v) \\ &\leq \binom{2\kappa}{5} n \left(\frac{c\,\omega}{n - o(n)}\right)^5 = O\left(\frac{\omega^5 n}{(\log n)^5}\right). \end{aligned} \tag{19}$$

Explanation of (19) We first show that $c\omega/(n-o(n))$ bounds the probability that walk $W_{i_t}=(w_1,w_2,\ldots,w_p)$ passes through v, given $W_{i_s}, 1\le s<t$ pass through v.

Suppose first that $j=3$. Then, given $X=\{\tilde{a}_1,\tilde{a}_2,\ldots,\tilde{a}_{j_1-1}\}$, (or $X=\{x_1,x_2,\ldots,x_{j_1-1}\}$) $w_1=\tilde{a}_{j_1}$ is chosen randomly from $V_3\setminus X$ (or $w_1=x_{j_1}$ is chosen randomly from V).

$$\mathbf{Pr}(x_1=v)\le\frac{1}{n-o(n)}\le\frac{3}{\beta_0}\pi(v). \tag{20}$$

If $j=4$ the argument is identical.

By induction on t we get

$$\mathbf{Pr}(x_t=v)=\sum_{w\in V_6}\mathbf{Pr}(x_{t-1}=w)P(w,v)\le\sum_{w\in V_5}\frac{3}{\beta_0}\pi(w)P(w,v)=\frac{3}{\beta_0}\pi(v)\le\frac{9}{\beta_0^2 n}$$

and we have the claimed bound of $\frac{9\omega}{\beta_0^2 n}$ for the (conditional) probability that W_{i_s} goes through v. There are at most $\binom{2\kappa}{t}$ choices for $W_{i_s}, 1\le s<t$ and n choices for v and (19) follows.

It follows from (19) and the Markov inequality that

$$\mathbf{Pr}\left(M_j\ge\frac{n}{8(\log n)^4}\right)=o(1).$$

In addition to these $M=M_3+M_4$ vertices we consider those vertices which are removed by REMOVE(Γ_6). Arguing as in Lemma 1 we see that if $M=o(n)$ then $|L_5|\le 3M$. We deduce that

$$\mathbf{Pr}\left(|L_5|\ge\frac{n}{2(\log n)^4}\right)=o(1). \tag{21}$$

We now estimate the probability that $i\in L_7$. We apply (18) with $G=\Gamma_5$, $S=V_j$, $j=3$ or 4, and $q(v)=\frac{1+o(1)}{|V_5|}$ for $v\in V_5$. Then we have

$$1-\lambda\ge\frac{\beta_0^4}{64},\ N_q\le\left(\frac{3}{\beta_0}\right)^{1/2}\ \text{and}\ \pi(S)\ge 1-\frac{3\gamma_0}{\beta_0}\ge\frac{1}{2}.$$

Putting $t=\omega$ and $u=t\pi(S)$ we get

$$\mathbf{Pr}(W\cap V_j=\emptyset)=O(e^{-\beta_0^4\omega/5120})$$

and so

$$\mathbf{E}(|L_7|)=O(e^{-\beta_0^4\omega/5120}\kappa)$$

and

$$\mathbf{Pr}\left(|L_7|\ge\frac{n}{2(\log n)^4}\right)=o(1).$$

Combining this with (21) we see that $\mathbf{Pr}(|L|\ge n/(\log n)^4)=o(1)$.

6 Analysis of Phase 4

We join the pairs in L using the algorithm of [5]. The algorithm is capable of joining $\Omega(n/(\log n)^{2+o(1)})$ distinct pairs, provided the graph has sufficient edge-expansion. Notice that $\tilde{a}_i, \tilde{b}_i$ are chosen as distinct vertices. We briefly describe how we can make this algorithm route $m \leq \frac{n}{(\log n)^4}$ pairs using the graphs G_6 – G_{10}, assuming only that $\Phi_6, \ldots, \Phi_{10} \geq 1$. Let $\lambda = \lceil \log n \rceil$.

(a) The aim here is to choose w_j, W_j, $1 \leq j \leq 2m$ such that (i) $w_j \in W_j$, (ii) $|W_j| = \lambda+1$, (iii) the sets $W_j, 1 \leq j \leq 2m$ are pairwise disjoint and (iv) W_j induces a connected subgraph of G_8.

As in [12] we can partition an arbitrary spanning tree T of G_8. Since T has maximum degree at most r we can find $2m$ vertex disjoint subtrees T_j, $1 \leq j \leq 2m$ of T, each containing between $\lambda + 1$ and $(r-1)\lambda + 2$ vertices. We can find T_1 as follows: choose an arbitrary root ρ and let $Q_1, Q_2, \ldots, Q_\sigma$ be the subtrees of ρ. If there exists l such that Q_l has between $\lambda+1$ and $(r-1)\lambda+2$ vertices then we take $T_1 = Q_l$. Otherwise we can search for T_1 in any Q_ℓ with more than $(r-1)\lambda + 2$ vertices. Since $T \setminus T_1$ is connected, we can choose all of the T_j's in this way. Finally, W_j is the vertex set of an arbitrary $\lambda + 1$ vertex subtree of T_j and w_j is an arbitrary member of W_j for $j = 1, 2, \ldots, 2m$.

(b) Let S_A, S_B denote the set of sources and sinks that need to be joined. Using a network flow algorithm in G_6 connect in an arbitrary manner the vertices of $S_A \cup S_B$ to $W = \{w_1, \ldots, w_{2m}\}$ by $2m$ edge disjoint paths. The expansion properties of G_6 ensure that such paths always exist.

Let $\tilde{a}_k$ (resp. $\tilde{b}_k$) denote the vertex in W_i that was connected to the original end-point a_k (resp. b_k). Our problem is now to find edge disjoint paths joining $\tilde{a}_k$ to $\tilde{b}_k$ for $1 \leq k \leq m$.

(c) If w_t has been renamed as $\tilde{a}_k$ (resp. $\tilde{b}_k$) then rename the elements of W_t as $\tilde{a}_{k,\ell}$, (resp. $\tilde{b}_{k,\ell}$,) $1 \leq \ell \leq \lambda$. Choose $\xi_j, 1 \leq j \leq \lambda m$ and $\eta_j, 1 \leq j \leq \lambda m$ independently at random from the steady state distribution π of a random walk on G_{10}. Using a network flow algorithm as in (b), connect $\{\tilde{a}_{k,\ell} : 1 \leq k \leq m, 1 \leq \ell \leq \lambda\}$ to $\{\xi_j : 1 \leq j \leq \lambda m\}$ by edge disjoint paths in G_8. Similarly, connect $\{\tilde{b}_{k,\ell} : 1 \leq k \leq m, 1 \leq \ell \leq \lambda\}$ to $\{\eta_j : 1 \leq j \leq \lambda m\}$ by edge disjoint paths in G_9. Rename the other endpoint of the path starting at $\tilde{a}_{k,\ell}$ (resp. $\tilde{b}_{k,\ell}$) as $\hat{a}_{k,\ell}$ (resp. $\hat{b}_{k,\ell}$). Once again the expansion properties of G_8, G_9 ensure that flows exist.

(d) Choose $\hat{x}_{k,\ell}, 1 \leq k \leq m, 1 \leq \ell \leq \lambda$ independently at random from the steady state distribution π of a random walk on G_{10}. Let $W'_{k,\ell}$ (resp. $W''_{k,\ell}$) be a random walk of length $\theta \log n$ from $\hat{a}_{k,\ell}$ (resp. $\hat{b}_{k,\ell}$) to $\hat{x}_{k,\ell}$. Here θ is sufficiently large that a random walk of this length on G_{10} is "well mixed". The use of this intermediate vertex $\hat{x}_{k,\ell}$ helps to break some conditioning caused by the pairing up of the flow algorithm.

Let B'_k (resp. B''_k) denote the *bundle* of walks $W'_{k,\ell}, 1 \leq \ell \leq \lambda$ (resp. $W''_{k,\ell}, 1 \leq \ell \leq \lambda$). Following [14] we say that $W'_{k,\ell}$ is *bad* if there exists $k' \neq k$ such that $W'_{k,\ell}$ shares an edge with a walk in a bundle $B'_{k'}$ or $B''_{k'}$. Each walk starts at an independently chosen vertex and moves to an independently chosen destination. The steady state of a random walk is uniform on edges and so at each stage of a walk, each edge is equally likely to be crossed.

Thus

$$\mathbf{Pr}(W'_{k,\ell} \text{ is bad}) \leq \frac{2\lambda m\theta^2(\log n)^2}{\beta_0 rn} = O\left(\frac{1}{\log n}\right).$$

We say that index k is bad if either B'_k or B''_k contain more than $\lambda/3$ bad walks. If index k is not bad then we can find a walk from $\hat{a}_{k,\ell}$ to $\hat{b}_{k,\ell}$ through $\hat{x}_{k,\ell}$ for some ℓ which is edge disjoint from all other walks. This gives a walk
$a_k - \tilde{a}_k - \tilde{a}_{k,\ell} - \hat{a}_{k,\ell} - \hat{x}_{k,\ell} - \hat{b}_{k,\ell} - \tilde{b}_{k,\ell} - \tilde{b}_k - b_k$,
which is edge-disjoint from all other such walks.

The probability that index k is bad is at most

$$2\,\mathbf{Pr}(B(\lambda, O(1/(\log n)) \geq \lambda/3) = O(n^{-2}).$$

So with probability 1-o(1) there are no bad indices. □

Acknowledgement: We thank Aravind Srinivasan for a careful reading of the paper.

References

[1] J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*, North-Holland 1976.

[2] A. Z. Broder, A. M. Frieze, and E. Upfal, *Existence and construction of edge disjoint paths on expander graphs*, SIAM Journal on Computing 23 (1994) 976-989.

[3] A. Z. Broder, A. M. Frieze, and E. Upfal, *Existence and construction of edge low congestion paths on expander graphs*, Random Structures and Algorithms 14 (1999) 87-109.

[4] I.H.Dinwoodie, *A probability inequality for the occupation probability of a reversible Markov chain*, The Annals of Applied Probability 5 (1995) 37-43.

[5] A.M.Frieze, Disjoint Paths in Expander Graphs via Random Walks: a Short Survey, Proceedings of Random '98, Lecture Notes in Computer Science 1518 (1998) Springer, 1-14.

[6] A.M.Frieze and M.Molloy, *Splitting an expander graph*, Journal of Algorithms 33 (1999) 166-172.

[7] A.M.Frieze and L.Zhao, *Edge disjoint paths in random regular graphs*, Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms (1999) 291-299.

[8] D. Gale, *A theorem on flows in networks*, Pacific Journal of Mathematics 7 (1957) 1073-1082.

[9] D.Gillman, *A Chernoff bound for random walks on expander graphs*, SIAM Journal on Computing 27 (1998) 1203-1220.

[10] N.Kahale, *Large deviation bounds for Markov chains*, DIMACS Technical Report, DIMACS, Rutgers University, New Brunswick NJ, 1994.

[11] J.Kleinberg and R.Rubinfeld, Short paths in expander graphs, *Proceedings of the 37th Annual IEEE Symposium on Foundations of Computer Science*, (1996) 86-95.

[12] T.Leighton and S.Rao, Circuit switching: a multicommodity flow based approach, Proceedings of a Workshop on Randomized Parallel Computing 1996.

[13] T.Leighton, S.Rao and A.Srinivasan, Multi-commodity flow and circuit switching, *Proceedings of the Hawaii International Conference on System Sciences*, 1998.

[14] T.Leighton, S.Rao and A.Srinivasan, New algorithmic aspects of the local lemma with applications to partitioning and routing.

[15] A. Lubotsky, R. Phillips, and P. Sarnak, *Ramanujan graphs*, Combinatorica 8 (1988) 261-277.

[16] D. Peleg and E. Upfal, *Constructing disjoint paths on expander graphs*, Combinatorica 9, (1989) 289-313.

[17] N. Robertson and P. D. Seymour, *Graph minors-XIII: The disjoint paths problem*, to appear.

[18] A. Sinclair and M. Jerrum, *Approximate counting, uniform generation, and rapidly mixing Markov chains*, Information and Computation 82 (1989) 93-133.