Hardness of the Undirected Edge-Disjoint Paths Problem

Matthew Andrews
Bell Labs, Murray Hill, NJ
andrews@research.bell-labs.com

Lisa Zhang
Bell Labs, Murray Hill, NJ
ylz@research.bell-labs.com

Bell Laboratories
600-700 Mountain Avenue
Murray Hill, NJ 07974

February 28, 2005

Abstract

We show that there is no $\log^{\frac{1}{3}-\varepsilon} M$ approximation for the undirected Edge-Disjoint Paths problem unless $NP \subseteq ZPTIME(n^{\text{polylog}(n)})$, where M is the size of the graph and ε is any positive constant. This hardness result also applies to the undirected All-or-Nothing Multicommodity Flow problem and the undirected Node-Disjoint Paths problem.

1 Introduction

Consider an undirected graph G and a set $\{(s_i, t_i)\}$ of source-sink pairs. In the undirected Edge-Disjoint Paths problem (EDP) we wish to connect as many of these pairs as possible using edge-disjoint paths. EDP is generally regarded as one of the "classic" NP-hard problems. Past work on EDP and the more general Unsplittable Flow[1] (USF) problem includes [2, 3, 4, 6, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 21].

Suppose that G has N nodes and M edges. The best known approximation ratio for EDP is $O(\min(N^{2/3}, M^{1/2}))$ [11, 21, 17, 4, 23]. In [10], Guruswami et al. presented an almost matching lower bound for *directed* graphs. They showed that there is no $M^{\frac{1}{2}-\varepsilon}$ approximation algorithm for any $\varepsilon > 0$ unless $P = NP$.

In this paper we show a hardness result for the *undirected* problem. In particular we show that there is no $\log^{\frac{1}{3}-\varepsilon} M$-approximation for EDP unless $NP \subseteq ZPTIME(n^{\text{polylog}(n)})$.[2] Our reduction is via a reduction from Maximum-Independent-Set (MIS) on bounded degree graphs [22] and is motivated by the result of Guruswami et al. [10] that *Bounded-Length* EDP is hard to approximate to within $M^{\frac{1}{2}-\varepsilon}$ for any $\varepsilon > 0$. This is a version of EDP in which all paths are restricted to be of length at most L, for some parameter L that is given as input. At a high level, our construction involves embedding multiple copies of the instance of [10] into a graph that "almost" has high girth.

[1] In the USF problem each source-sink pair has a demand d_i and a profit p_i and each edge has a capacity. If we route all the demand d_i on a single path then we gain the profit p_i. The aim is to maximize the profit gained while respecting the edge capacities. In the case that the demands, profits and capacities are all 1, USF reduces to EDP. Since in this paper we are concerned with hardness results, all our results will apply directly to USF as well as EDP.

[2] Recall that $ZPTIME(n^{\text{polylog}(n)})$ is the set of languages that have randomized algorithms that always give the correct answer and have expected running time $n^{\text{polylog}(n)}$.

Outline. We prove our hardness result via a reduction from Maximum-Independent-Set in bounded-degree graphs. In [22], Trevisan showed that for *fixed* Δ, there is no poly-nomial-time $\Delta/2^{O(\sqrt{\log \Delta})}$-approximation for MIS on graphs of bounded degree Δ unless $P = NP$. Unfortunately, we cannot apply Trevisan's result directly since we need the degree bound Δ to grow with the size of the graph. We therefore use the following theorem. The proof is almost identical to Trevisan's proof and so we defer it to the Appendix.

Theorem 1 *Let $f(\cdot)$ be a polylogarithmic function. For any constant $\alpha > 0$ there is a randomized $|\psi|^{\Theta(\log\log|\psi|)}$ time reduction from a 3CNF formula ψ to a graph $\mathcal{F}$ with $n = |\psi|^{\Theta(\log\log|\psi|)}$ nodes and maximum degree at most $\Delta = f(n)$ such that for two parameters Z_1 and $Z_2 \leq Z_1/\Delta^{1-\alpha}$:*

- *If the ψ is satisfiable then $\mathcal{F}$ has an independent set of size Z_1.*
- *If ψ is not satisfiable then with probability $1 - 1/poly(|\psi|)$ the maximum independent set in $\mathcal{F}$ has size at most Z_2.*

This result immediately implies that MIS in n-node graphs of bounded-degree $\Delta = f(n)$ is hard to approximate to within a factor of $\Delta^{1-\alpha}$ unless $NP \subseteq ZPTIME(n^{\Theta(\log\log n)})$.

In the remainder of the paper we show how this implies that EDP is hard to approximate in undirected graphs. In Sections 2.1 and 2.2 we show how to translate an instance $\mathcal{F}$ of MIS into a randomized instance $\mathcal{H}$ of EDP for which we are able to show our hardness result. This instance has two important features. First, for each demand there is a special short path that we call the *canonical path* for the demand. Second, with high probability $\mathcal{H}$ is almost a high-girth graph and so for most demands, any non-canonical path for the demand is much longer than the canonical path. Therefore, it is impossible to route many demands on non-canonical paths. The idea of high-girth graph resembles that in [1] in which polylogarithmic lower bounds are shown for the buy-at-bulk problem.

In Section 3 we present an algorithm that takes a solution to the EDP problem in $\mathcal{H}$ and maps it back into an independent set in $\mathcal{F}$. We then analyze how the size of the MIS solution relates to the size of the EDP solution. For this purpose, in Section 3.2 we count how many demands can be routed on edge-disjoint non-canonical paths. As mentioned above, this number is small. In Section 3.3 we count how many demands can be routed on edge-disjoint canonical paths. This number depends on the size of the maximum independent set in $\mathcal{F}$. We tie all the analysis together in Section 4. In Section 5 we show how our analysis can be extended to give hardness results for the undirected All-or-Nothing Multicommodity Flow problem and the undirected Node-Disjoint Paths problem.

2 Construction

2.1 Construction of basic instance

Consider an n-node graph $\mathcal{F}$ of degree Δ. We convert it into a basic instance $\mathcal{G}$ of EDP as follows. We do not define $\mathcal{G}$ using the convention of specifying its node set and edge set. Instead, we specify the set of paths that $\mathcal{G}$ supports.

For each edge ij in $\mathcal{F}$ we create a path segment S_{ij} in $\mathcal{G}$, which consists of $c = \text{polylog}(n)$ edges $e_{ij,k}$, $0 \leq k < c$. (The exact value of c will be given later.) Two adjacent edges $e_{ij,k}$ and $e_{ij,k+1}$ in the segment are connected by two disjoint paths; one via an *auxiliary node* for i and the other via an auxiliary node for j. (See Figure 2, Left.) Once we have created these segments in $\mathcal{G}$ we create a *canonical* path P_i in $\mathcal{G}$ for every node i in $\mathcal{F}$. The path P_i strings together segments $S_{ij_0}, S_{ij_1}, \ldots$ in

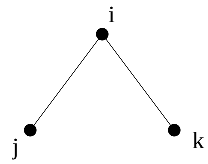

Figure 1: Graph $\mathcal{F}$, $\Delta = 2$.

an arbitrary order, where $j_0, j_1, \ldots$ are the (at most Δ) neighbors of i in $\mathcal{F}$. Within each segment, P_i follows the auxiliary nodes for i. We add a path of length 2 to connect the end of segment S_{ij_k} and the beginning of segment $S_{ij_{k+1}}$. We also attach a source node s_i to the beginning of the first segment and attach a destination node t_i to the end of the last segment. Our edge-disjoint paths problem has terminal pairs (s_i, t_i) for all i. (See Figure 2, Right.) The canonical paths have the following two key properties:

- **G-1:** Path P_i has length at most $\ell = 3c\Delta$.
- **G-2:** Path P_i is disjoint from path P_j if and only if i is not a neighbor of j in $\mathcal{F}$.

2.2 Construction of expanded instance

We now create an expanded instance $\mathcal{H}$ for which we can show that EDP is hard to approximate. The graph $\mathcal{H}$ is created randomly and is simple to describe. For each node v in $\mathcal{G}$ we create $X = O(n^{\mathrm{polylog}(n)})$ nodes v_x, $0 \leq x < X$. If v and v' are non-adjacent then there are no edges between v_x and $v'_{x'}$ for any x, x'. If v and v' are adjacent then we place a *random matching* between the set of nodes v_x and the set of nodes $v'_{x'}$. Therefore, there are X edges in $\mathcal{H}$ that *correspond to* every edge in $\mathcal{G}$. The first important property of this construction is:

- **H-1:** For any path $u, v, w, \ldots$ in $\mathcal{G}$, there are X paths of the form $u_x, v_{x'}, w_{x''}, \ldots$ in $\mathcal{H}$ that are both node disjoint and edge disjoint.

The above property means that there are X paths corresponding to the canonical path P_i. We use $P_{i,x}$ to denote the canonical path corresponding to P_i that starts at node $s_{i,x}$. If $t_{i,x'}$ is the endpoint of this canonical path then we let $(s_{i,x}, t_{i,x'})$ be a terminal pair in our new EDP instance.

We now consider a canonical path $P_{i,x}$ corresponding to node i in $\mathcal{F}$ and a canonical path $P_{j,y}$ corresponding to a neighbor j of i in $\mathcal{F}$. We say that path $P_{i,x}$ *meets* path $P_{j,y}$ at level k if and only if they both pass through the same edge in $\mathcal{H}$ that corresponds to edge $e_{ij,k}$ in $\mathcal{G}$. We have two important properties with respect to two neighboring nodes i and j in $\mathcal{F}$.

- **H-2:** For each path $P_{i,x}$ there exists exactly one y such that path $P_{i,x}$ meets $P_{j,y}$ at level k, i.e. the relationship "meets at level k" induces a matching between the X canonical paths for node i and the X canonical paths for node j.
- **H-3:** The event that two paths meet at level k is independent of the event that they meet at level $k' \neq k$. Hence the matchings induced between paths at level k are independent of the matchings induced between paths at level $k' \neq k$.

Lemma 2 *The size of the EDP instance on $\mathcal{H}$ is quasipolynomial, i.e. $O(n^{\mathrm{polylog}(n)})$. In particular the number of edges M is at most $Xn\ell$ and the number of terminal pairs is equal to nX.*

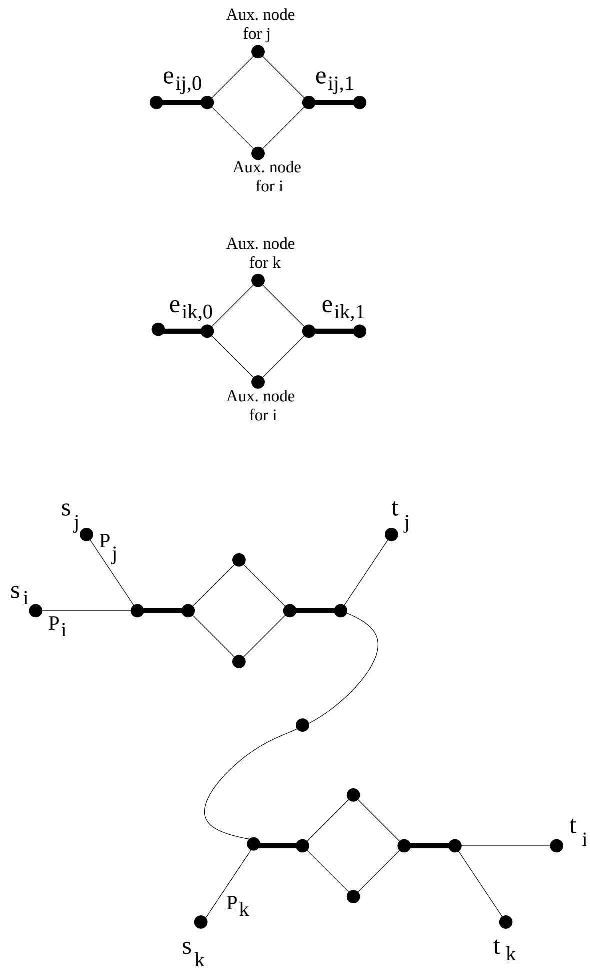

Figure 2: Graph $\mathcal{G}$ constructed from $\mathcal{F}$ in Figure 1, $c = 2$. (Top) Two segments S_{ij} and S_{ik}. (Bottom) Creation of canonical paths from the segments.

3 From EDP to MIS

Suppose that the solution to our EDP instance $\mathcal{H}$ has size Y. We propose the following algorithm which we call EDP2MIS to construct an independent set S for $\mathcal{F}$. We partition the Y edge disjoint paths into two pieces, the set of terminals that are routed on canonical paths and the set of terminals that are routed on non-canonical paths. For parameter A defined in (5), if node i in $\mathcal{F}$ corresponds to more than A canonical paths chosen by the EDP solution instance, then we include i in S. We show later that the set S produced by EDP2MIS is indeed an independent set with high probability.

EDP2MIS allows us to relate the size of the MIS solution to the size of the EDP solution. Since for every node that is included in S at most X canonical paths can be included in the EDP solution, and for every node not included in S at most A canonical paths can be included in the solution, we have

$$Y \leq X|S| + (n - |S|)A + N_{nc}, \tag{1}$$

where N_{nc} is the number of demands routed along non-canonical paths. We dedicate the rest of this section to proving the following.

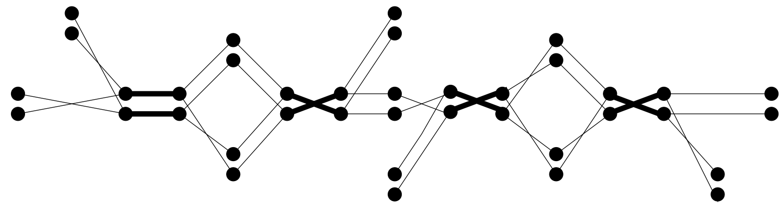

Figure 3: Graph $\mathcal{H}$ constructed from $\mathcal{G}$ in Figure 2, $X = 2$.

Theorem 3

1. *With probability* $2/3$*, in any solution to our EDP instance the number of demands that are routed along non-canonical paths is at most* $X + Xn\ell/(g - \ell)$.

2. *With a high probability,* EDP2MIS *finds an independent set* S.

3.1 Overview and Parameter Definitions

Due to the randomness in the construction of $\mathcal{H}$, we first show in Section 3.2 that not many terminal pairs can be routed along non-canonical paths. Recall that the *girth* of a graph is the length of its smallest cycle. In a random graph such as $\mathcal{H}$, not many cycles have small (i.e. polylogarithmic) size. We introduce a girth-related parameter g, and define X to be exponential in g. (This in turn implies that the size of $\mathcal{H}$ is exponential in in g.) We then show that the number of demands that are routed on short non-canonical paths is small since any short non-canonical path will form a short cycle with the corresponding canonical path. In addition, the number of demands that are routed on long non-canonical paths is small since any long non-canonical path will make use of many edges in $\mathcal{H}$. We therefore show that N_{nc} is small.

We then focus on the demands that are routed on canonical paths in Section 3.3. In the extreme case of $A = X$, if we include i and j in the set S then i and j cannot be neighbors in $\mathcal{F}$. This is because, as stated in property H-2, every canonical path of i meets with some canonical path of j at every level of $\mathcal{H}$, and therefore any EDP solution cannot include all X canonical paths of i and X canonical paths of j. Under our choice of A which is a logarithmic fraction of X, we show that for a *fixed* set of A canonical paths of i and a *fixed* set of A canonical paths of j, the probability p that none of these paths meet is small. Recall that in our construction we created c levels in the graphs. This means that if two neighboring i and j are included in S, their canonical paths must not meet at each of the c levels. By the independence property H-3 this happens with probability p^c. We then apply a union bound to show that for *any* set of A canonical paths of i and *any* set of A canonical paths of j, the probability p that none of these $2A$ paths meet is small.

We now discuss our parameter choices. The relationship among the parameters is fairly intricate. However, we attempt to give a high level idea of our choices here. We need X to be quasipolynomial in n in order to keep the number of demands that can be routed on short non-canonical paths small. For a given X, when A is close to X, the probability p as discussed in the previous section is desirably small. However, the bound (1) on the EDP solution Y also gets loose. As we explain at the end of Section 4, in order to get a logarithmic hardness of EDP, A needs to be a logarithmic fraction of X or smaller.

Given A, we use the c repetitions in the graph construction to make sure the probability p^c is sufficiently small in the canonical path analysis. Meanwhile, although a larger c favors the probabilistic analysis, c also lower bounds the girth parameter g we need Theorem 3, item 1 to hold and so $g > \ell = 3\Delta c$. Since X is exponential in g and the size of the EDP instance is polynomial in X, g and therefore c are polylog(n) in order to get a quasipolynomial reduction.

The value of Δ is constrained in a number of places. We need Δ to be large since the hardness of the MIS instance depends on Δ. However, as mentioned above $g > 3\Delta c$ and g is polylog(n). Hence Δ cannot be larger that polylog(n). We expand on this discussion in Section 4.

We now define the parameters precisely.

$$g = \log^{\phi} n \qquad \text{for some constant } \phi > 0 \tag{2}$$

$$X = g(4ng)^{g+1} \tag{3}$$

$$\gamma = \frac{\phi}{3(\phi+1)} \tag{4}$$

$$A = \frac{X}{\log^{\gamma} X} \tag{5}$$

$$c = 3\gamma(\log\log X)\log^{\gamma} X \tag{6}$$

$$\Delta = \min\{\log^{\gamma} X - 1, \sqrt{g/(9c)}\} = \Theta\left(\frac{\log^{\frac{\phi}{3}} n}{\sqrt{\log\log n}}\right) \tag{7}$$

3.2 Bounding the number of non-canonical paths

We now show that not many terminals can be routed on a non-canonical path. The key insight is that by our random construction, $\mathcal{H}$ is almost a high-girth graph. In particular, we use the following lemma whose proof is extremely similar to that of the Erdös-Sachs theorem [7] (which states that high-girth graphs exist).

Lemma 4 *Consider a random graph with ν nodes and let $\{e_0, e_1, \ldots, e_{\kappa-1}\}$ represent a set of $\kappa < g$ potential edges. If,*

$$Pr[e_0 \ \textit{exists}|e_1, \ldots, e_{\kappa-1} \ \textit{exist}] \leq \rho,$$

for all such sets of edges then the expected number of cycles of fixed length $g' \leq g$ is at most $(\nu\rho)^{g'}$. This implies that with probability $\frac{2}{3}$, the number of cycles of length less than or equal to g is at most $3(\nu\rho)^{g+1}$.

Proof: The total number of potential cycles of length g' is at most $\frac{1}{2g'}\frac{\nu!}{(\nu-g')!}$. Each such cycle occurs with probability at most $\rho^{g'}$. Therefore, the expected number of cycles of length g' is at most,

$$\frac{\nu!\rho^{g'}}{2g'(\nu-g')!} \leq (\nu\rho)^{g'}.$$

This implies that the expected number of cycles of length less than or equal to g is at most,

$$\sum_{g'=1}^{g} (\nu\rho)^{g'} \leq (\nu\rho)^{g+1}.$$

By Markov's inequality, with probability $\frac{2}{3}$ the number of cycles of length at most g is at most $3(\nu\rho)^{g+1}$. □

Proof of Theorem 3, item 1: The number of nodes in the graph $\mathcal{H}$ is at most $2Xn\ell$. The probability that a potential edge in $\mathcal{H}$ exists equals $1/X$ since we construct $\mathcal{H}$ using matchings of size X. Even if g edges are fixed in $\mathcal{H}$, the probability that some other edge exists is at most $1/(X-g)$. Hence we can take $\nu = 2Xn\ell$ and $\rho = 1/(X-g)$. Since X is exponentially larger than g, we can bound $\nu\rho$ by $4n\ell$. Therefore by Lemma 4, with probability 2/3, the number of cycles of length less than or equal to g is at most $3(4n\ell)^{g+1}$.

It is easy to verify that the graph $\mathcal{H}$ has maximum degree 3. Therefore, each node is within distance g of at most 3^g other nodes. This implies that with probability 2/3 the number of nodes that are within distance g of a cycle of length less than or equal to g is at most $g \cdot 3^g \cdot 3(4n\ell)^{g+1}$, which equals

$$g(12n\ell)^{g+1}.$$

We are now ready to count the demands that are routed on non-canonical paths. Consider a demand with source node $s_{i,x}$ that has canonical path $P_{i,x}$ but which is routed on a path $Q \neq P_{i,x}$. Since $P_{i,x}$ and Q are two nonidentical paths between the same pair of nodes, it is clear that the union of $P_{i,x}$ and Q must contain a cycle. There are two cases to consider.

- **Case 1.** Node $s_{i,x}$ is within distance g of a cycle of length less than or equal to g.
- **Case 2.** Node $s_{i,x}$ is *not* within distance g of a cycle of length less than or equal to g. Note that the union of $P_{i,x}$ and Q must contain more than g edges; otherwise $s_{i,x}$ would be within distance g from a cycle of length less than or equal to g.

By our previous analysis, with probability 2/3, Case 1 can be true for at most $g(12n\ell)^{g+1}$ source nodes. Since $\ell = 3c\Delta$ by definition and $9c\Delta \leq g$ due to (7), we have $g(12n\ell)^{g+1} \leq g(4ng)^{g+1}$ which equals X by definition (3). If Case 2 holds then since $P_{i,x}$ has length at most ℓ, path Q must have length at least $g-\ell$. Since the graph $\mathcal{H}$ has at most $Xn\ell$ edges, the number of edge-disjoint paths of length at least $g-\ell$ is at most $Xn\ell/(g-\ell)$. Theorem 3, item 1 follows. □

3.3 Proving that S is an independent set with high probability

Let C_i be the set of canonical paths corresponding to node i in $\mathcal{F}$. For two nodes i and j that are neighbors in $\mathcal{F}$ let I be a subset of C_i of size A and let J be a subset of C_j of size A. Recall that the relationship "meets at level k" induces a matching between C_i and C_j. If A is large enough we would expect some path in I would meet a path in J at level k. More formally, we say that a *bad event* $B(i,j,I,J,k)$ occurs if and only if there do *not* exist paths in I and J that meet at level k. We now analyze the probability of event $B(i,j,I,J,k)$ (where the probability is with respect to the random construction of the graph $\mathcal{H}$).

Lemma 5

$$Pr[B(i,j,I,J,k) \text{ occurs}] \leq e^{-\frac{A^2}{X}}.$$

Proof: The number of matchings between C_i and C_j at level k is $X!$. The number of matchings for which no path in I meets a path in J is equal to $(X-A)(X-A-1)\ldots(X-2A+1)(X-A)!$. Hence,

$$\begin{aligned} & Pr[B(i,j,I,J,k) \text{ occurs}] \\ = \; & \frac{(X-A)(X-A-1)\ldots(X-2A+1)}{X(X-1)\ldots(X-A+1)} \\ \leq \; & \left(\frac{X-A}{X}\right)^A \end{aligned}$$

$$
\begin{aligned}
&= \left(1 - \frac{A}{X}\right)^{\frac{X}{A}\frac{A^2}{X}} \\
&\leq e^{-\frac{A^2}{X}}.
\end{aligned}
$$

□

Let $B(i,j,I,J)$ be the bad event that $B(i,j,I,J,k)$ occurs at *all* levels k. By property H-3, the event that two paths meet at level k is independent of the event that they meet at level $k' \neq k$. We immediately have,

$$Pr[B(i,j,I,J) \text{ occurs}] \leq e^{-\frac{cA^2}{X}}.$$

Let $B(i,j)$ be the bad event that *there exist* sets I and J of size A such that $B(i,j,I,J)$ occurs. There are $\binom{X}{A}^2$ choices for the sets I and J. Therefore,

$$
\begin{aligned}
Pr[B(i,j) \text{ occurs}] &\leq \binom{X}{A}^2 e^{-\frac{cA^2}{X}} \\
&\leq \left(\frac{eX}{A}\right)^{2A} e^{-\frac{cA^2}{X}} \\
&= e^{2A(\log X + 1 - \log A) - \frac{cA^2}{X}}.
\end{aligned}
$$

Finally, let B be the bad event that $B(i,j)$ happens for *some* pair of neighboring nodes (i,j). By a union bound,

$$Pr[B \text{ occurs}] \leq n^2 e^{2A(\log X + 1 - \log A) - \frac{cA^2}{X}}.$$

By the definitions of A in (5) and c in (6) we can rewrite the above probability bound by

$$Pr[B \text{ occurs}] \leq n^2 \cdot e^{-\frac{\gamma X \log\log X - 2X}{\log^\gamma X}}.$$

By the definition of X in (3), $\frac{\gamma X \log\log X - 2X}{\log^\gamma X}$ is $\omega(\log n)$. Hence,

Lemma 6 *For the parameters chosen in (2)-(7), the bad event B does not happen with probability $1 - 1/poly(n)$.*

Recall our algorithm, EDP2MIS described in Section 3, for finding an independent set for $\mathcal{F}$ from a set of edge-disjoint paths in $\mathcal{H}$. We first verify that the set S defined by EDP2MIS is indeed an independent set of $\mathcal{F}$ with high probability. If nodes i and j are neighbors in $\mathcal{F}$, the probabilistic analysis in the previous section states with high probability, any set of A paths in C_i and any set of A paths in C_j will intersect at some level k in the graph $\mathcal{H}$. Therefore, it is unlikely that both i and j have at least A paths in a solution to the EDP instance on $\mathcal{H}$. As a result, it is unlikely that both i and j belong to the set S. Therefore, Lemma 6 is a restatement of Theorem 3, item 2.

4 From EDP to 3SAT

Finally, we define an algorithm for satisfiability, which we call EDP2SAT, as follows. Given a 3CNF formula ψ, we first use the reduction from Theorem 1 to create an instance of MIS, namely

$\mathcal{F}$, with polylogarithmic node degree Δ, as defined in (7). We then use the construction described in Sections 2.1 and 2.2 to create an instance of EDP, namely $\mathcal{H}$. Suppose that the optimal solution to the EDP instance is Y_{opt} and we have a β-approximation algorithm that produces a solution of size Y. From Y, the algorithm EDP2MIS defines a set S.

- We declare ψ unsatisfiable if S is an independent set and $Y < XZ_1/\beta$, where Z_1 was one of the parameters defined in Theorem 1 and X is defined in (3).
- We declare ψ satisfiable otherwise, i.e. if S is an independent set and $Y \geq XZ_1/\beta$, or if S is not an independent set.

Theorem 7 *For any constant $\alpha > 0$, there is no $\Delta^{1-\alpha}/3$ approximation for EDP unless $NP \subseteq ZPTIME(n^{\text{polylog}(n)})$.*

Proof: Suppose there is a β approximation for EDP where $\beta < \Delta^{1-\alpha}/3$, and this approximation algorithm finds Y edge-disjoint paths in $\mathcal{H}$. If the 3CNF formula ψ is satisfiable, then $\mathcal{F}$ has an independent set of size at least Z_1 by Theorem 1. Since the canonical paths corresponding to this independent set are edge disjoint by property H-1, we have $Y_{opt} \geq Z_1 X$. A β-approximation algorithm for EDP guarantees $Y \geq Y_{opt}/\beta \geq Z_1 X/\beta$. Hence we always declare ψ to be satisfiable.

If ψ is unsatisfiable, then by Theorem 1 the graph $\mathcal{F}$ has an independent set of size more than Z_2 with probability at most $1/poly(|\psi|)$. Therefore by Theorem 3, with probability more than 1/2 the following two events both occur:

- S is an independent set with size at most Z_2.
- At most $X + Xn\ell/(g-\ell)$ demands are routed along non-canonical paths in the EDP solution.

Now suppose ψ is unsatisfiable and the above two events both occur. We now show that we always declare ψ to be unsatisfiable. From (1) we have,

$$\begin{aligned} Y &\leq X|S| + (n-|S|)A + X + Xn\ell/(g-\ell) \\ &\leq (X-A)Z_2 + nA + X(1+n\ell/(g-\ell)) \\ &= XZ_2(1-\log^{-\gamma} X) + X\log^{-\gamma} Xn \\ &\qquad + X(1+n\ell/(g-\ell)). \end{aligned} \tag{8}$$

The second inequality follows from the fact that $|S| \leq Z_2$ and the equality follows from the definition of A in (5). We proceed to show that every term in the above is at most XZ_2. One simple algorithm for finding an independent set for a graph with degree Δ is to iteratively choose any remaining node and eliminate all its neighbors. This algorithm guarantees an independent set of size at least $n/(\Delta+1)$. Hence, $Z_2 \geq n/(\Delta+1)$. In addition, $\Delta \leq \log^{\gamma} X$-1 due to (7). We therefore have $Z_2 \geq n\log^{-\gamma} X$. Since $\Delta \leq \sqrt{g/(9c)}$ due to (7) and $\ell = 3\Delta c$, we also have $Z_2 \geq \frac{n}{\Delta+1} \geq 1 + \frac{n}{3\Delta-1} = 1 + \frac{3n\Delta c}{9\Delta^2 c - 3\Delta c} \geq 1 + \frac{n\ell}{g-\ell}$. Therefore,

$$Y < 3XZ_2 \leq \frac{3XZ_1}{\Delta^{1-\alpha}} < \frac{XZ_1}{\beta}.$$

Hence we declare ψ to be unsatisfiable.

Note that the size of $\mathcal{H}$ is $n^{\text{polylog}(n)} = |\psi|^{\text{polylog}(|\psi|)}$. If $\beta < \Delta^{1-\alpha}/3$, we have described a $coRTIME(n^{\text{polylog}(n)})$ procedure[3] that solves 3SAT. By a standard result, if $NP \subseteq coRTIME(n^{\text{polylog}(n)})$

[3]When we write complexity classes such as $coRTIME(n^{\text{polylog}(n)})$, n simply denotes a parameter. It is not meant to refer to the number of nodes in $\mathcal{F}$.

then $NP \subseteq ZPTIME(n^{\text{polylog}(n)})$. Therefore, unless $NP \subseteq ZPTIME(n^{\text{polylog}(n)})$ there is no $\Delta^{1-\alpha}/3$-approximation for EDP. □

Finally, let us express $\Delta^{1-\alpha}/3$ in terms of M, where $M \leq Xn\ell$ is the number of edges in $\mathcal{H}$. Since $g = \log^{\phi} n$, we have $\log X = \Theta(\log^{\phi+1} n)$ and $\log M = O(\log^{\phi+1} n)$. Since $\gamma = \frac{\phi}{3(\phi+1)}$, the definition of Δ implies $\Delta = \min\{\log^{\gamma} X - 1, \sqrt{g/(9c)}\} = \Omega(\log^{\frac{\phi}{3}} n/\sqrt{\log\log n})$. Hence, $\Delta^{1-\alpha} = \Omega\left((\log n)^{\frac{\phi}{3} - \frac{2\phi\alpha}{3}}\right)$, which is $\Omega\left((\log M)^{\frac{\phi}{3(\phi+1)} - \frac{2\phi\alpha}{3(\phi+1)}}\right)$. By choosing ϕ large and α small we have,

Theorem 8 *For any constant $\varepsilon > 0$, there is no $O(\log^{\frac{1}{3}-\varepsilon} M)$ approximation algorithm for the undirected Edge-Disjoint Paths problem unless $NP \subseteq ZPTIME(n^{\text{polylog}(n)})$.*

Now that we have completed our proof let us look at the parameter choices again, in particular how Δ and A are chosen. We require $\Delta \leq \frac{X}{A} - 1$ when upper bounding the second term of (8). We also require $\Delta \leq \sqrt{g/(9c)}$ when upper bounding the third term of (8). Since g is polylog(X), Δ can be at most polylog(X). Since the inapproximability of EDP is $\Delta^{\Theta(1)}$, a larger Δ is better. Therefore we choose A and Δ so that both $\frac{X}{A}$ and Δ are polylog(X).

5 Related problems

In this section we discuss the extent to which our result applies to other routing problems.

5.1 All-or-Nothing Multicommodity Flow

All-or-nothing multicommodity flow is a relaxation of EDP in which each demand is allowed to be routed on fractional paths subject to the constraint that the total demand routed through an edge is at most 1. The objective is to maximize the number of demands for which the entire demand is routed. Our hardness result is easy to adapt to this relaxed problem. Recall that for the algorithm EDP2MIS we partition the routed demands according to whether or not they were routed on canonical paths. For the All-or-Nothing problem we partition the demands according to whether or not strictly more than half of the demand is routed along the canonical path. Clearly, if two demands have strictly more than half of their demand routed along their canonical paths then these paths are edge-disjoint. Therefore, if ψ is not satisfiable our analysis in Section 3.3 immediately implies that with high probability there are at most $XZ_2 + (n - Z_2)A$ demands of this type. Moreover, the analysis of Section 3.2 implies that with high probability, the total amount of demand that can be routed along non-canonical paths is at most $X + Xn\ell/(g - \ell)$, even if this demand is routed fractionally. Therefore, the total number of demands for which at least half the demand is routed along non-canonical paths is at most $2(X + Xn\ell/(g - \ell))$. It is now easy to adapt the arguments of Section 4 to obtain,

Theorem 9 *For any constant $\varepsilon > 0$, there is no $O(\log^{\frac{1}{3}-\varepsilon} M)$ approximation algorithm for All-or-Nothing Multicommodity Flow unless $NP \subseteq ZPTIME(n^{\text{polylog}(n)})$.*

We remark that Chekuri et al. presented a polylogarithmic approximation for this problem in [5].

5.2 Node-Disjoint Paths

In the undirected Node-Disjoint Paths (NDP) problem we wish to route as many demands as possible on node-disjoint paths. Our reduction for EDP applies directly to NDP. Note that any set of node-disjoint paths is also edge-disjoint. Therefore, for any node-disjoint solution in $\mathcal{H}$, we can apply the algorithm EDP2SAT to determine if the 3CNF formula ψ is satisfiable. If ψ is satisfiable then there are $Z_1 X$ demands that can be routed on canonical paths. It is easy to see that in our construction these canonical paths are node-disjoint as well as being edge-disjoint. If ψ is not satisfiable then the maximum number of demands that can be routed using node-disjoint paths is no bigger than the maximum number of demands that can be routed using edge-disjoint paths. Therefore, our analysis shows,

Theorem 10 *For any constant $\varepsilon > 0$, there is no $O(\log^{\frac{1}{3}-\varepsilon} M)$ approximation algorithm for the undirected Node-Disjoint Paths problem unless $NP \subseteq ZPTIME(n^{\mathrm{polylog}(n)})$.*

Acknowledgements

The authors wish to thank Chandra Chekuri and Bruce Shepherd for many helpful comments and for suggesting that we look at the problems discussed in Section 5. The authors are also grateful to Luca Trevisan and Salil Vadhan for helping us to better understand the analysis in [22]. Lastly, the authors are grateful to an anonymous STOC referee for their insightful report.

References

[1] M. Andrews. Hardness of buy-at-bulk network design. In *Proceedings of the 45th Annual Symposium on Foundations of Computer Science*, pages 115 – 124, Rome, Italy, October 2004.

[2] Y. Aumann and Y. Rabani. Improved bounds for all optical routing. In *Proceedings of the 6th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 567 – 576, 1995.

[3] A. Baveja and A. Srinivasan. Approximation algorithms for disjoint paths and related routing and packing problems. *Mathematics of Operations Research*, 25:255–280, 2000.

[4] C. Chekuri and S. Khanna. Edge-disjoint paths revisited. In *Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2003.

[5] C. Chekuri, S. Khanna, and F.B. Shepherd. The all-or-nothing multi-commodity flow problem. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, 2004.

[6] Y. Dinitz, N. Garg, and M. Goemans. On the single source unsplittable flow problem. In *Proceedings of the 39th Annual IEEE Symposium on Foundations of Computer Science*, pages 290–299, Palo Alto, CA, November 1998.

[7] P. Erdös and H. Sachs. Reguläre graphen gegebener Taillenweite mit minimaler Knotenzahl. *Wiss. Z. Uni. Halle-Wittenburg (Math. Nat.)*, 12:251–257, 1963.

[8] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Interactive proofs and the hardness of approximating cliques. *Journal of the ACM*, 43(2):268–292, 1996.

[9] N. Garg, V. Vazirani, and M. Yannakakis. Primal-dual approximation algorithms for integral flow and multicut in trees. *Algorithmica*, 18:3 – 20, 1997.

[10] V. Guruswami, S. Khanna, R. Rajaraman, B. Shepherd, and M. Yannakakis. Near-optimal hardness results and approximation algorithms for edge-disjoint paths and related problems. In *Proceedings of the 31st Annual ACM Symposium of Theory of Computing*, pages 19–28, Atlanta, GA, May 1999.

[11] J. Kleinberg. *Approximation algorithms for disjoint paths problems.* PhD thesis, MIT, Cambridge, MA, May 1996.

[12] J. Kleinberg. Single-source unsplittable flow. In *Proceedings of the 37th Annual Symposium on Foundations of Computer Science*, pages 68 – 77, Burlington, VT, October 1996.

[13] J. Kleinberg. Decision algorithms for unsplittable flow and the half-disjoint paths problem. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 530 – 539, Dallas, TX, May 1998.

[14] 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.

[15] J. Kleinberg and E. Tardos. Disjoint paths in densely embedded graphs. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, pages 52 – 61, 1995.

[16] S. Kolliopoulos and C. Stein. Improved approximation algorithms for unsplittable flow problems. In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science*, pages 426 – 435, Miami Beach, FL, October 1997.

[17] S. Kolliopoulos and C. Stein. Approximating disjoint-path problems using greedy algorithms and packet integer programs. In *Integer Programming and Combinatorial Optimization*, 1998.

[18] T. Leighton and S. Rao. An approximate max-flow min-cut theorem for uniform multicommodity flow problems with applications to approximation algorithms. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*, pages 422 – 431, 1988.

[19] B. Ma and L. Wang. On the inapproximability of disjoint paths and minimum steiner forest with bandwidth constraints. *Journal of Computer and System Sciences*, 60:1–12, 2000.

[20] A. Samorodnitsky and L. Trevisan. A PCP characterization of NP with optimal amortized query complexity. In *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, 2000.

[21] A. Srinivasan. Improved approximations for edge-disjoint paths, unsplittable flow, and related routing problems. In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science*, pages 416 – 425, Miami Beach, FL, October 1997.

[22] L. Trevisan. Non-approximability results for optimization problems on bounded degree instances. In *Proceedings of the 33rd Annual ACM Symposium of Theory of Computing*, 2001.

[23] K. Varadarajan and G. Venkataraman. Graph decomposition and a greedy algorithm for edge-disjoint paths. In *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2004.

A Hardness of Maximum Independent Set on Bounded-Degree Instances

In this section we prove Theorem 1 which states that the Maximum Independent Set problem is hard to approximate to within $\Delta^{1-\alpha}$ in graphs of degree $\Delta = f(n)$ where $f(\cdot)$ is a polylogarithmic function and α is an arbitrarily small constant. Our proof is a minor extension of the result in [22] which shows a similar result for graphs of fixed degree Δ.

The reduction uses a PCP characterization of NP. We consider a verifier that is given a 3CNF instance ψ and oracle access to an alleged proof P that ψ is satisfiable. After examining ψ the verifier tosses $O(\log|\psi|)$ random bits and makes a series of queries into the oracle proof. On receiving the answers to these queries the verifier decides whether or not to accept. The completeness of the PCP is the minimum probability that the verifier accepts when ψ is satisfiable. The soundness of the PCP is the maximum probability that the verifier accepts when ψ is not satisfiable. A configuration of the PCP is the specification of the random string together with the values of the bits of the proof that are read by the verifier when using that random string. We make use of the following result from [20].

Lemma 11 *For every $\mu > 0$ and every $k > 0$ there is a PCP characterization of NP with the following properties:*

- *The verifier has completeness $1 - \mu$, soundness $2^{-k^2} + \mu$ and queries $2k + k^2$ bits.*
- *For each random string there are 2^{2k} satisfying configurations.*
- *For each random string and for each bit that is read, the number of accepting configurations where the bit is zero equals the number of accepting configurations where the bit is one.*

In the following we let $r = O(\log|\psi|)$ be the number of random bits used in the PCP and we set $\mu = 2^{-k^2}$. We use $\frac{1}{2}$ as a lower bound on the completeness.

We now consider λ independent repetitions of the above PCP. In this case the number of random bits is λr, the number of bits queried is $\lambda(2k + k^2)$, the completeness is reduced to $\frac{1}{2^\lambda}$ and the soundness is reduced to $2^\lambda \cdot 2^{-\lambda k^2}$.

We now apply the well-known FGLSS [8] reduction to this PCP. This reduction creates a graph with one node for each accepting configuration. Two vertices are connected by an edge if and only if the configurations are inconsistent (i.e. they assign different values to the same bit in the proof). In our case, when we apply the FGLSS reduction we obtain a graph $\mathcal{F}$ with $n = 2^{\lambda(r+2k)}$ nodes such that if ψ is satisfiable then $\mathcal{F}$ has an independent set of size $\frac{1}{2^\lambda} \cdot 2^{\lambda r}$ whereas if ψ is not satisfiable then $\mathcal{F}$ has no independent set of size greater than $2^\lambda \cdot 2^{-\lambda k^2} \cdot 2^{\lambda r}$.

For each index i in the proof let O_i be the set of vertices in $\mathcal{F}$ where index i is queried and the answer is 1 and let Z_i be the set of vertices in $\mathcal{F}$ where index i is queried and the answer is 0. Let $n_i = |O_i|$. By the third property of Lemma 11 we also have $n_i = |Z_i|$. The graph $\mathcal{F}$ contains a complete bipartite graph between the nodes in O_i and the nodes in Z_i. It is easy to see that $\mathcal{F}$ consists of precisely the union of these complete bipartite graphs.

We now replace each of these bipartite graphs with a low-degree expander. In particular, by a simple probabilistic argument that is similar to the analysis in Section 3.3 we have,

Lemma 12 *Consider a random δ-regular bipartite graph with n_i nodes in each partition. (Such a graph can be constructed by taking δ random bipartite matchings). We say that this graph is bad* *if there exist sets A, B such $|A| \geq cn_i$, $|B| \geq cn_i$ and A and B are not connected by an edge. If $\delta = \frac{3}{c}\log\frac{1}{c}$ then the probability that the graph is bad is at most,*

$$e^{-n_i c(\log\frac{1}{c}-2)}.$$

We replace every bipartite graph between sets O_i and Z_i with a random δ-regular graph. Recall that the number of bits that are queried for each random string is $\lambda(2k+k^2)$. Therefore the total number of bits that could be queried is at most $2^{\lambda r}\lambda(2k+k^2)$. We can also assume without loss of generality that $n_i \geq 2^{\lambda r/2}$ since if not the verifier could always toss additional random bits and then not use them in its computation.

Lemma 13 *The probability that one of the bipartite graphs is bad is at most,*

$$2^{\lambda r}\lambda(2k+k^2)e^{-2^{\lambda r/2}c(\log\frac{1}{c}-2)}.$$

Suppose now that for all i the bipartite graph between O_i and Z_i is good. Note that the maximum independent set in the new graph can only be bigger. Moreover, for each i an independent set either contains at most cn_i nodes from O_i or at most cn_i nodes from Z_i. Therefore we can remove at most $\sum_i cn_i$ nodes from the independent set in the new graph to get an independent set in the original graph which used complete bipartite graphs. Since each accepting configuration belongs to at most $\lambda(2k+k^2)$ sets of the form O_i or Z_i, we have $\sum_i cn_i \leq c\lambda(2k+k^2)n$.

We set $c = 2^{\lambda(1-k^2+r)}/\lambda(2k+k^2)n$. The above argument shows that it is hard to decide whether the new graph has an independent set of size $\frac{1}{2^\lambda}2^{\lambda r}$ or whether all independent sets have size less than $2^\lambda \cdot 2^{-k^2\lambda}\cdot 2^{\lambda r} + c\lambda(2k+k^2)n$. The latter expression is at most $2^{\lambda(2-k^2+r)}$. The "gap" between these two quantities is given by,

$$H = 2^{\lambda(k^2-3)}.$$

Our other important parameters are:

$$\begin{aligned} c &= 2^{\lambda(1-k^2-2k)}/\lambda(2k+k^2) \\ \delta &= 3\lambda(2k+k^2)2^{\lambda(k^2+2k-1)}\cdot(\lambda(k^2+2k-1)+\log(\lambda(2k+k^2))) \\ n &= 2^{\lambda(r+2k)}. \end{aligned}$$

Since each accepting configuration involves $\lambda(2k+k^2)$ queries the maximum degree in the graph G is at most $\Delta := \lambda(2k+k^2)\delta$. For sufficiently large k and for any λ we have,

$$\begin{aligned} \Delta &\leq 3\lambda^3(2k+k^2)^2 2^{\lambda(k^2+2k-1)}\cdot(k^2+2k-1+\log(\lambda(2k+k^2))) \\ &\leq 4\cdot 2^\lambda\cdot(3(2k+k^2)^2)2^{\lambda(k^2+2k-1)}\cdot(\lambda+k^2+2k-1+\log(2k+k^2)) \\ &\leq 2^\lambda 2^k 2^{\lambda(k^2+2k-1)}2^\lambda 2^k \\ &\leq 2^{\lambda(k^2+6k-1)}. \end{aligned}$$

For any fixed $\alpha > 0$ we choose k such that $k^2-3 \geq (1-\alpha)(k^2+6k-1)$. Now consider an increasing function $f(\cdot)$. If we can choose λ such that,

$$\lambda = \frac{1}{k^2+6k-1}\log(f(2^{\lambda(r+2k)})), \tag{9}$$

then,

$$\begin{aligned} \Delta &\leq f(n) \\ H &\geq \Delta^{1-\alpha}. \end{aligned}$$

If $f(\cdot)$ is a polylogarithmic function then Equation 9 is satisfied for $\lambda = \Theta(\log\log|\psi|)$. For this choice of λ the error probability from Lemma 13 is at most $1/poly(|\psi|)$ and the number of nodes in the graph G is $n = 2^{\lambda(r+2k)} = |\psi|^{\Theta(\log\log|\psi|)}$.

Theorem 1 *Let $f(\cdot)$ be a polylogarithmic function. For any constant $\alpha > 0$ there is a randomized $|\psi|^{\Theta(\log\log|\psi|)}$ time reduction from a 3CNF formula ψ to a graph $\mathcal{F}$ with $n = |\psi|^{\Theta(\log\log|\psi|)}$ nodes and maximum degree at most $\Delta = f(n)$ such that for two parameters Z_1 and $Z_2 \leq Z_1/\Delta^{1-\alpha}$:*

- *If the ψ is satisfiable then $\mathcal{F}$ has an independent set of size Z_1.*
- *If ψ is not satisfiable then with probability $1 - 1/poly(|\psi|)$ the maximum independent set in $\mathcal{F}$ has size at most Z_2.*