Approximability of Packing Disjoint Cycles

Zachary Friggstad[*]   Mohammad R. Salavatipour[†]

Department of Computing Science

University of Alberta

Edmonton, Alberta T6G 2E8, Canada

`zacharyf,mreza@cs.ualberta.ca`

Abstract

Given a graph G, the edge-disjoint cycle packing problem is to find the largest set of cycles of which no two share an edge. For undirected graphs, the best known approximation algorithm has ratio $O(\sqrt{\log n})$ where $n = |V(G)|$ and is due to Krivelevich *et al* [14, 15]. In fact, they proved the same upper bound for the integrality gap of this problem by presenting a simple greedy algorithm. Here we show that this is almost best possible. By modifying integrality gap and hardness results for the edge-disjoint paths problem given by Chuzhoy and Khanna [1, 9], we show that the undirected edge-disjoint cycle packing problem has an integrality gap of $\Omega(\frac{\sqrt{\log n}}{\log \log n})$ and furthermore it is quasi-NP-hard to approximate the edge-disjoint cycle problem within ratio of $O(\log^{\frac{1}{2}-\epsilon} n)$ for any constant $\epsilon > 0$. The same results hold for the problem of packing vertex-disjoint cycles.

1   Introduction

In the problem of *edge-disjoint cycle packing* (EDC) we are given a graph G and our goal is to find a largest set of edge-disjoint cycles. The vertex analog of the problem, *vertex-disjoint cycle packing* (VDC), is the problem of finding a largest set of vertex-disjoint cycles in the given graph. The EDC problem has been studied extensively in both directed and undirected settings (e.g. see Balister et al. [3], Caprara et al.[5], and Seymour [18]). A discussion on the applications of packing cycles to computational biology and reconstructing evolutionary trees can be found in [3].

Both EDC and VDC are known to be NP-hard even for undirected graphs and for very restricted cases of the problem (see e.g. [10]). This motivates the study of approximation algorithms for these problems. Caprara, Panconesi and Rizzi [5] showed that EDC is APX-hard even when restricted on planar graphs. They also presented a simple greedy algorithm with approximation ratio $O(\log n)$. Recently, Krivelevich et al. [14, 15] showed that a modification of the simple greedy algorithm of [5] with a more careful analysis yields an $O(\sqrt{\log n})$-approximation for EDC on undirected graphs. In fact, the algorithm obtains an integer solution that is within factor $O(\sqrt{\log n})$ of the optimal fractional solution. They showed examples for which the solution obtained by the greedy algorithm was within $\Omega(\sqrt{\log n})$ of the optimal solution but it falls short of proving any super-constant lower bound on the

---

integrality gap or approximability of the problem. They also presented an $O(\sqrt{n})$-approximation for EDC on directed graphs and an $O(\log n)$-approximation for undirected VDC. Subsequently, in [16, 15], an integrality gap of $\Omega(\frac{\log n}{\log \log n})$ for EDC on directed graphs was proved. This result was followed by a hardness of approximation. There was proved that unless $\text{NP} \subseteq \text{DTIME}(n^{\text{polylog}(n)})$, it is hard to approximate EDC on directed graphs within $O(\log^{1-\epsilon} n)$ for any $\epsilon > 0$. However, the best known lower bound on the approximability of EDC on undirected graphs remains APX-hardness and the best lower bound for integrality gap is $O(1)$. For EDC on planar graphs, Caprara, Panconesi, and Rizzi give a $2 + \epsilon$-approximation algorithm [4].

For the related problem of edge-disjoint paths (EDP), on directed graphs the best approximation algorithms have ratio $O(\min\{n^{\frac{2}{3}} \log^{\frac{1}{3}} n, \sqrt{m}\})$ [6, 13, 19] and it is known the problem is hard to approximate within $O(m^{\frac{1}{2}-\epsilon})$ for any $\epsilon > 0$ [11]. For undirected graphs, the best known approximation ratio for EDP is $O(\sqrt{n})$ [7] whereas the best known hardness result is only $\Omega(\log^{\frac{1}{2}-\epsilon} n)$ for any $\epsilon > 0$ [1, 9]. The latter result was built on the major advance on the lower bound of EDP (from APX-hardness to $\Omega(\log^{\frac{1}{3}-\epsilon} n)$) by Andrews and Zhang [2].

In this paper, we improve the lower bounds for both EDC and VDC. More specifically, we first present an integrality gap construction which shows that the integrality gap upper bound of [14, 15] is almost tight.

Theorem 1.1 *The EDC problem on undirected graphs has an integrality gap of $\Omega(\frac{\sqrt{\log n}}{\log \log n})$.*

Then we show almost the same bound for the hardness of approximation for these problems.

Theorem 1.2 *The EDC problem on undirected graphs is hard to approximate within $O(\log^{\frac{1}{2}-\epsilon} n)$ for any $\epsilon > 0$ unless $\text{NP} \subseteq \text{ZPTIME}(n^{\text{polylog}(n)})$.*

This shows that the simple greedy algorithm of [14, 15] with approximation ratio $O(\sqrt{\log n})$ is almost best possible for EDC. The reduction in the proof of Theorem 1.2 works, without modification, to prove the same hardness result for VDC. Our results are heavily motivated by the hardness of the edge-disjoint paths problem presented by Chuzhoy and Khanna in [9]. Nevertheless, they show a rather surprising approximability threshold for a very natural packing problem. In fact there are very few problems known to have a sub-logarithmic approximability threshold ([8, 12]). One other important message to be taken from our results is that, in order to improve the hardness of approximation for EDP (from $\Omega(\log^{\frac{1}{2}-\epsilon} n)$ to beyond $\Omega(\sqrt{\log n})$), there has to be substantially new ideas developed that exploit the differences between EDC and EDP problems; since such a hardness result should not be adaptable to work for EDC (because we already have an $O(\sqrt{\log n})$-approximation for EDC).

The rest of the paper is separated into three more sections. In Section 2 we describe the construction of a graph with large integrality gap for EDC. The ideas from this section motivate the proof of hardness of approximation result in the subsequent section. Section 3 starts by recalling a PCP characterization of NP presented in [9] based on the results of Samorodnitsky and Trevisan [17]. Following this, the construction of the EDC instance is presented as a reduction from 3SAT using this PCP result. Section 4 analyzes this construction to prove the hardness result for EDC.

2 Integrality Gap

The construction of an instance of the EDC problem with a large integrality gap is similar to the construction used by Chuzhoy and Khanna in [9]. We begin by generating a random graph G and use this graph to generate another

graph H. With sufficiently large probability, the resulting graph H has a super-constant integrality gap. We use a model of random graph different from [9] which enables us to handle some special cases that are overlooked in the analysis of [9].

2.1 Constructing the Gap

Given a sufficiently large integer n, define $\beta_1 = \frac{\sqrt{\log n}}{8 \log \log n}$ and $\beta_2 = 5\beta_1 \ln \beta_1$. The ultimate goal is to construct a graph with $O(n^2)$ nodes and integrality gap $\Omega(\beta_1)$. Start by building a random Hamiltonian cycle F on n vertices. Then we add a random graph $G_1 = G_{n,p}$ to F with $p = \frac{2\beta_2}{n-1}$, i.e. for each pair of nodes, if there is no edge between them already (due to F), we add it randomly (and independently) with probability $p = \frac{2\beta_2}{n-1}$. This is our graph G.

Now, from graph G we will create another graph H as follows. For each edge $e_i \in G$, add vertices ℓ_i and r_i to H and connect them with an edge $\ell_i r_i$ which will be called a *special* edge. Finally, for each vertex v of G, let $e_{v_1}, e_{v_2}, \ldots, e_{v_k}$ be the edges incident with v in some arbitrary order. Add edges $r_{v_i} \ell_{v_{i+1}}$ to H for all $1 \le i \le k$ where $v_{k+1} = v_1$. Call the sequence of vertices $\ell_{v_1}, r_{v_1}, \ell_{v_2}, r_{v_2}, \ldots, \ell_{v_k}, r_{v_k}$ the *canonical cycle* of v denoted by C_v. Notice that each special edge $\ell_i r_i$ in H (corresponding to edge $e_i \in G$) appears in exactly two canonical cycles C_u and C_v where u and v are the endpoints of e_i in the original graph G. Every other edge appears in exactly one canonical cycle in H. Note that since the minimum degree of G is 2, every vertex in G has a corresponding canonical cycle in H. So we have n canonical cycles in H.
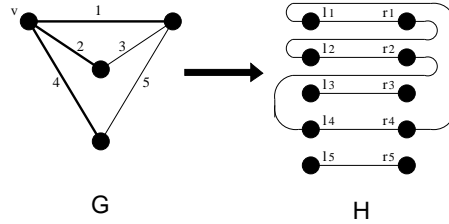

Figure 1: Constructing the canonical cycle for v.

2.2 Analysis

If we assign a fractional value of $\frac{1}{2}$ to each canonical cycle in H, each special edge has total fractional value 1 and all the other edges have fractional value $\frac{1}{2}$. Thus, no edge constraint is violated and we have a fractional packing of cycles with total value $n/2$ in H (as there are n canonical cycles in H).

We now bound the number of cycles in any integral packing $\mathcal{C}$ in H. First observe that the expected degree of each node v in G_1, $\mathrm{E}[d(v)]$, is $2\beta_2$. Using Chernoff bound, $\Pr[d(v) < \beta_2] \le \Pr[d(v) < \frac{1}{2}\mathrm{E}[d(v)]] \le e^{-\beta_2^2/4}$. So the expected number of nodes with degree smaller than β_2 in G_1 is at most $n \cdot e^{-\beta_2^2/4} \le \frac{n}{8\beta_1}$. Thus using Markov's inequality, with probability at least $\frac{7}{8}$ the number of nodes with degree smaller than β_2 in G_1, and therefore in G, is at most $\frac{n}{\beta_1}$.

Second, let $M_1 = |E(G_1)|$. Since $\mathrm{E}[M_1] = p \cdot \binom{n}{2} = \beta_2 n$, again using Chernoff bound, the probability of $|M_1 - \beta_2 n| > \beta_2 n/4$ is exponentially small. So we can assume that with probability at least $\frac{7}{8}$:

$$\frac{3}{4}\beta_2 n < M_1 < \frac{5}{4}\beta_2 n. \tag{1}$$

3

If we define $M = |E(G)|$, since $M_1 \leq M \leq M_1 + n$, using (1), the probability of event $|M - \beta_2 n| > \beta_2 n/2$ is at most $\frac{7}{8}$. Let the bad event $\mathcal{E}_0$ be the event that either $|M - \beta_2 n| > \beta_2 n/2$ or there are more than $\frac{n}{\beta_1}$ nodes with degree smaller than β_2. From above it follows with probability at least $\frac{3}{4}$ event $\mathcal{E}_0$ does not happen.

Note that for every pair of nodes uv, for the probability of having an edge $e = uv$ in G we have:

$$
\begin{aligned}
\Pr[e \notin G] &= \Pr[e \notin F] \cdot \Pr[e \notin G_1 | e \notin F] \\
&= \left(1 - \frac{2(n-2)!}{(n-1)!}\right) \cdot \left(1 - \frac{2\beta_2}{n-1}\right) \\
&= \left(1 - \frac{2}{n-1}\right) \cdot \left(1 - \frac{2\beta_2}{n-1}\right) \\
&= 1 + \frac{4\beta_2}{(n-1)^2} - \frac{2\beta_2 + 2}{n-1}
\end{aligned}
$$

Thus $\Pr[e \in G] = \frac{2\beta_2+2}{n-1} - \frac{4\beta_2}{(n-1)^2} \leq \frac{3\beta_2}{n-1}$. Defining $p' = \frac{3\beta_2}{n-1}$ we can assume each edge exists in G with probability at most p'. For $g = 6\beta_1\beta_2$, we say that a cycle is short if it is of length less than g and long otherwise. Let $\mathcal{C}_1, \mathcal{C}_2$, and $\mathcal{C}_3$ be the set of canonical cycles, long cycles, and short cycles of $\mathcal{C}$, respectively. So $\mathcal{C} = \mathcal{C}_1 \cup \mathcal{C}_2 \cup \mathcal{C}_3$. We will bound the size of each $\mathcal{C}_i$ by $O(n/\beta_1)$ which implies $|\mathcal{C}| \in O(n/\beta_1)$. Let the bad event $\mathcal{E}_1$ be the event that there are more than n/β_1 edge-disjoint canonical cycles in $\mathcal{C}$, i.e. $|\mathcal{C}_1| > n/\beta_1$.

Lemma 2.1 *The probability of bad event $\mathcal{E}_1$ happening is at most $\frac{1}{4}$.*

Proof. First we obtain a bound on the probability that some fixed subset $S \subseteq V(G)$ of size n/β_1 doesn't contain an edge. Since each edge exists with probability p', the probability that a fixed set S of size n/β_1 is empty is at most:

$$
\begin{aligned}
\left(1 - \frac{3\beta_2}{n-1}\right)^{\binom{n/\beta_1}{2}} &\leq \left(1 - \frac{3\beta_2}{n}\right)^{n^2/(4\beta_1^2)} \\
&\leq e^{-\frac{\beta_2 n}{2\beta_1^2}}.
\end{aligned}
$$

The number of such sets S is $\binom{n}{n/\beta_1} \leq (e\beta_1)^{n/\beta_1} \leq \beta_1^{2n/\beta_1}$; so by union bound the probability of having any set S of size n/β_1 that does not contain an edge is at most:

$$
\beta_1^{\frac{2n}{\beta_1}} \cdot e^{\frac{-\beta_2 n}{2\beta_1^2}} \leq e^{\frac{n}{\beta_1}\left(2\ln\beta_1 - \frac{\beta_2}{2\beta_1}\right)} \leq e^{-\frac{n \ln \beta_1}{2\beta_1}} \leq \frac{1}{4}.
$$

$\square$

To bound $|\mathcal{C}_2|$, first observe that graph H has $3M$ edges. Since all cycles in $\mathcal{C}_2$ are of length at least g then $|\mathcal{C}_2|$ is easily bound by $\frac{3M}{g} \leq \frac{n}{\beta_1}$, assuming that $\mathcal{E}_0$ does not happen.

Now we bound the size of $\mathcal{C}_3$. First, obtain the multi-graph H' from H by contracting all special edges $\ell_i r_i$ to a single vertex u_{e_i}. So each such vertex u_{e_i} now corresponds to an edge e_i in G. If there are two edges between two nodes of H' then the only explanation can be that the edges come from the same canonical cycle corresponding to a degree 2 vertex of G. If we assume bad event $\mathcal{E}_0$ does not happen, there are at most $\frac{n}{\beta_1}$ cycles of length 2 in H'. Now we bound the number of cycles of length $3 \leq k < g$ in H'. It is easy to see that a bound on the number of

4

cycles of length less than g in H' is an upper bound on the number of cycles of length less than g in G. Denote by $\mathcal{E}_2$ the event that there are more than $\frac{n}{\beta_1}$ simple cycles in H'.

Lemma 2.2 *The probability of bad event $\mathcal{E}_2$ occurring is at most $\frac{1}{4}$.*

Proof. We begin by bounding the expected number of cycles of some fixed length $3 \leq k < g$ in H'. Let $C = e_{i_1}, e_{i_2}, \ldots, e_{i_k}$ be an ordered sequence of edges forming a cycle in H' where $e_{i_j} = u_{i_j} u_{i_{j+1}}$ and all u_{i_j}'s are distinct for $1 \leq j \leq k$ where $i_{k+1} = i_1$. Denote the edge in G corresponding to u_{i_j} by h_{i_j} for all $1 \leq j \leq k$. By the construction of H', for each two consecutive nodes $u_{i_j}, u_{i_{j+1}}$ in C $(1 \leq j \leq k)$, the corresponding edges h_{i_j} and $h_{i_{j+1}}$ in G must be incident with a vertex i.e. have a common end-point (note that $h_{i_1}, \ldots, h_{i_k}$ do not necessarily form a cycle in G since, for example, $h_{i_j}, h_{i_{j+1}}$, and $h_{i_{j+2}}$ can all be incident to the same vertex). Given a sequence of pairs of nodes in G like $h_{i_1}, \ldots, h_{i_k}$, whose corresponding nodes in H' form a simple cycle like C, the probability that all pairs $h_{i_1}, \ldots, h_{i_k}$ are actually edges in G is $\left(\frac{3\beta_2}{n-1}\right)^k$. A loose upper bound on the number of such sequences of pairs of nodes in G (that correspond to a simple cycle in H') is $(2n)^k$ since once we select a pair of nodes, there are at most $2n - 4$ other pairs of nodes, each of which has an end-point in common with the previous one. Also, every sequence of k edges in G, like $h_{i_1}, \ldots, h_{i_k}$ corresponds to a sequence of k nodes in H', say $u_{i_1}, \ldots, u_{i_k}$, and if this sequence forms a (simple) cycle then it forms at most 2^k cycles in H' because between every pair of nodes in H' there are at most two edges. Thus, the expected number of cycles of length k in H' is at most $2^k \cdot (2n)^k \cdot \left(\frac{3\beta_2}{n-1}\right)^k \leq (16\beta_2)^k$. Summing over all cycle lengths $3 \leq k < g$ gives an upper bound of $(16\beta_2)^g$ on the expected number of short (simple) cycles. By Markov's inequality, the probability that there are more than $4(16\beta_2)^g$ cycles of length $3 \leq k < g$ in H' is at most $\frac{1}{4}$. We show that this quantity is bound by n/β_1.

$$
\begin{aligned}
4(16\beta_2)^g &\leq \beta_2^{2g} = e^{2g \ln \beta_2} = e^{12\beta_1 \beta_2 \ln \beta_2} \\
&\leq e^{60\beta_1^2 \ln \beta_1 \ln \beta_2} \\
&\leq e^{120\beta_1^2 \ln^2 \beta_1} \\
&\leq e^{\frac{120 \ln n}{64(\ln \ln n)^2} \cdot \frac{1}{(2 \ln \ln n)^2}} \\
&\leq e^{\frac{\ln n}{2}} \\
&\leq \frac{n}{\beta_1}.
\end{aligned}
$$

Therefore, the probability that there are more than n/β_1 short (simple) cycles in $\mathcal{C}_3$ is at most $\frac{1}{4}$. $\square$

Therefore, assuming that $\mathcal{E}_0$ and $\mathcal{E}_2$ do not happen $|\mathcal{C}_3| \leq 2n/\beta_1$. By union bound, the probability of $\mathcal{E}_0 \cup \mathcal{E}_1 \cup \mathcal{E}_2$ is at most $\frac{3}{4}$. So with probability at least $\frac{1}{4}$ graph G (and accordingly graph H) with the above properties exist and so any collection of disjoint cycles of H is of size at most $|\mathcal{C}| \leq |\mathcal{C}_1| + |\mathcal{C}_2| + |\mathcal{C}_3| \leq \frac{4n}{\beta_1}$. Since we can pack $n/2$ cycles fractionally in H, then the integrality gap is $\Omega(\beta_1) = \Omega\left(\frac{\sqrt{\log n}}{\log \log n}\right)$. The number of vertices of graph H is $N = 2M \in O(n^2)$. Therefore, the integrality gap in H is $\Omega\left(\frac{\sqrt{\log N}}{\log \log N}\right)$.

3 The Hardness Construction

In this section we prove Theorem 1.2. We show how a modification of the construction used to prove the hardness of approximating edge-disjoint paths by Chuzhoy and Khanna in [9] can be used to show the same hardness for EDC. Our starting point is a PCP characterization of NP introduced in [17].

3.1 A PCP Characterization of NP

To begin, we use the same PCP characterization of NP used in [9] which is a slight modification of the characterization obtained by Samorodnitsky and Trevisan in [17]. Let Φ be an instance of 3SAT with n variables. For any constant $k > 0$, consider a PCP verifier that uses $r = O(\log n)$ random bits and queries $q = k^2$ bits of a proof Π. Let R be a random string of length r and denote the indices of the bits of the proof that are read given the random string R as $b_1(R), \ldots, b_q(R)$. Define a *configuration* to be the tuple $(R, a_1, \ldots, a_q)$ where R is a random string of length r and $a_i = \Pi_{b_i(R)} \in \{0, 1\}$, for $1 \leq i \leq q$, are the values of the bits read in the proof. A configuration $(R, a_1, \ldots, a_q)$ is called *accepting* if the PCP verifier accepts upon using random string R and reading proof bits $a_1, \ldots, a_q$. It follows ([9]) from the construction of [17] that for every constant $k > 0$ and for sufficiently large constant $\beta >> k^2$ there exists a a PCP verifier for Φ with the following properties:

- $\lambda r = O(\log n \log \log n)$ random bits are used with $r = O(\log n)$ and $\lambda = \frac{2\beta \log \log n}{k^2}$.

- Exactly $q = \lambda k^2 = O(\log \log n)$ bits of the proof are queried for each random string.

- If Φ is satisfiable, then there exists a proof Π such that the acceptance probability of the PCP verifier upon reading Π is at least $2^{-\lambda}$.

- If Φ is not satisfiable, then the acceptance probability of the PCP verifier upon reading Π is at most $2^{-\lambda k^2}$ for all proofs Π.

- Every random string R participates in $2^{\lambda(2k-1)}$ accepting configurations.

- For every random string R and for every $j = 1 \ldots q$, the number of accepting configurations with $\Pi_{b_j(R)} = 0$ and the number of accepting configurations with $\Pi_{b_j(R)} = 1$ are equal.

- Let Z_j to be the set of all accepting configurations with $\Pi_j = 0$ and let O_j be the set of all accepting configurations with $\Pi_j = 1$. Let $n_j = |Z_j| = |O_j|$. Then $n_j \geq 2^{\lambda r/2}$.

- Let $\mathcal{A}$ be the set of all accepting configurations. Then $|\mathcal{A}| \leq 2^{\lambda r} \cdot 2^{2\lambda k}$.

For a given instance of Φ of 3SAT with n variables, we assume that V is a PCP verifier with aforementioned properties and we choose k to be a large enough constant.

3.2 The Bit Gadget

The basic construction here is identical to that of [9]. Let M and X be two parameters which will be specified later. We only note that X will be exponentially larger than M, i.e. $X >> 2^M$. For each proof bit Π_i, we construct a *bit gadget* $G(i)$ in the following manner. Recall that Z_i and O_i are the set of accepting configurations in which bit

Π_i is zero and one, respectively. For each accepting configuration $\alpha \in Z_i \cup O_i$ and for each $1 \le m \le M + 1$, we create X vertices $v_{x,m}(\alpha, i)$, for $1 \le x \le X$, called *level m vertices*. Let $Z_m(i) = \{v_{1,m}(\alpha, i), \ldots, v_{X,m}(\alpha, i)\}$ be the set of level m vertices when $\alpha \in Z_i$. Similarly define $O_m(i)$ to be the set of level m vertices when $\alpha \in O_i$. Between levels m and $m + 1$, for $1 \le m \le M$, create Xn_i vertices $L_m(i) = \{\ell_{1,m}(i), \ldots, \ell_{Xn_i,m}(i)\}$ as well as Xn_i vertices $R_m(i) = \{r_{1,m}(i), \ldots, r_{Xn_i,m}(i)\}$ where $n_i = |Z_i| = |O_i|$.

The edges in the bit-gadget are specified as follows. For each $1 \le m \le M$, create a random matching between the Xn_i level m vertices associated with some $\alpha \in Z_i$ and the vertices in $L_m(i)$. Similarly, create a random matching between the vertices in $R_m(i)$ and the Xn_i level $m + 1$ vertices associated with some $\alpha \in Z_i$. Repeat the same process between vertices associated with some $\alpha \in O_i$. Finally, for each $1 \le m \le M$ and for each $1 \le j \le Xn_i$, join $\ell_{m,j}(i)$ and $r_{m,j}(i)$ with an edge which we call a *special* edge. Figure 2 illustrates this construction.

For each configuration $\alpha \in Z_i \cup O_i$, we define a canonical path $P_x(\alpha, i)$ for $1 \le x \le X$, as being the path

$$(v_{x_1,1}(\alpha, i), \ell_{a_1,1}(i), r_{a_1,1}(i), v_{x_2,2}(\alpha, i), \ldots, \ell_{a_M,M}(i), r_{a_M,M}(i), v_{x_{M+1},M+1}(\alpha, i))$$

where the indices $x_1 = x$ and the remaining x_m, a_m indices are defined by the random matchings. Essentially, a canonical path corresponding to configuration α begins at one of the X vertices $v_{x,1}(\alpha, i)$ and follows the random matchings between levels while never visiting vertices in O_i if $\alpha \in Z_i$ or never visiting vertices in Z_i if $\alpha \in O_i$.
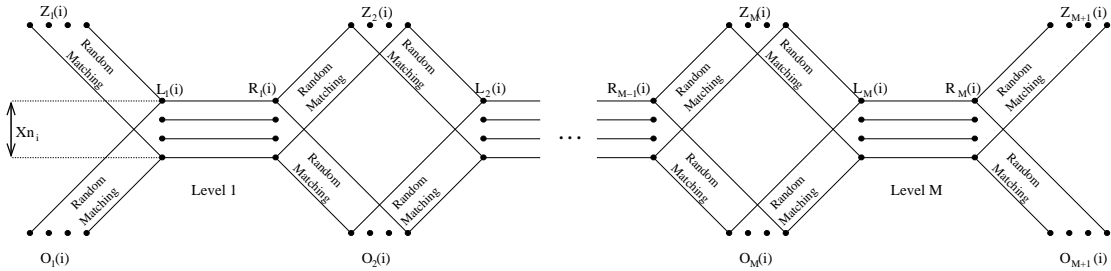

Figure 2: Bit gadget construction for proof Π_i

Note that the canonical paths corresponding to the Xn_i configurations in Z_i are all edge-disjoint. Similarly, the canonical paths to the configurations in O_i are edge-disjoint. Each special edge belongs to exactly two canonical paths (one corresponding to a configuration in Z_i and one in O_i) and every other edge belongs to exactly one canonical path. Consider the set of Xn_i special edges at level m ($1 \le m \le M$). Since each such special edge participates in exactly one canonical path representing a configuration in Z_i and one canonical path representing a configuration in O_i, the set of special edges in level m, defines a matching between canonical paths corresponding to configurations in Z_i and canonical paths corresponding to configuratinos in O_i. This matching is random (because of the random matchings placed before these special edges). So overall, the M levels of special edges define M random matchings between the canonical paths corresponding to configurations in Z_i and in O_i.

Let $\Delta = \frac{M}{8 \log M}$, noting that $M \ge 8\Delta \log \Delta$ holds. For each index i of proof Π, let $\mathcal{P}_0(i)$ be the set of canonical paths corresponding to a configuration in Z_i and $\mathcal{P}_1(i)$ be the set of canonical paths corresponding to a configuration in O_i. A bit gadget $G(i)$ is said to be *bad* if there is a pair of subsets $A \subseteq P_0(i)$, $B \subseteq P_1(i)$ with $|A| = |B| = \frac{Xn_i}{\Delta}$ such that all paths in $A \cup B$ are edge disjoint. Define bad event $\mathcal{B}_1$ to be the event that there is some bit gadget that is bad. The next lemma claims that with sufficiently high probability $\mathcal{B}_1$ does not happen. The proof is a simple first-moment analysis. The idea is that each path from A and each path from B can be matched by any of the M

7

random matchings defined by the special edges, in which case the two paths are not edge-disjoint. Since our bit gadget is identical to the bit gadget constructed in [9], the following result holds as well.

Lemma 3.1 *[9] The probability that bad event $\mathcal{B}_1$ happens is at most $\frac{1}{\text{poly}(n)}$.*

3.3 The Main Construction

In this subsection we show how to combine the bit gadgets into the final construction. This is essentially the same construction as in [9] with the modification that the corresponding source-sink pairs are connected by a new set of edges, called back-edges.

Let $\alpha = (R, a_{i_1}, a_{i_2}, \ldots, a_{i_q})$ be an accepting configuration with $i_1, \ldots, i_q$ being the indices of the proof bits queried upon reading the random string R. For each $1 \leq j < q$, we connect bit gadget $G(i_j)$ to bit gadget $G(i_{j+1})$ by creating a random matching between the sets of vertices $\{v_{1,M+1}(\alpha, i_j) \ldots v_{X,M+1}(\alpha, i_j)\}$ and $\{v_{1,1}(\alpha, i_{j+1}), \ldots v_{X,1}(\alpha, i_{j+1})\}$. For each $1 \leq x \leq X$, we define canonical path $P_x(\alpha) = (P_{x_1}(\alpha, i_1), \ldots, P_{x_q}(\alpha, i_q))$ where the x_j's are recursively defined as follows: $x_1 = x$ and x_j corresponds to the canonical path in $G(i_j)$ whose start point is matched with the end-point of $P_{x_{j-1}}(\alpha, i_{j-1})$ in $G(i_{j-1})$ for each $2 \leq j \leq q$.

After performing the random matching, add an edge, called a *back* edge, for each canonical path $P_x(\alpha)$ between the start and end vertices in that path. From this, we define a *canonical cycle $C_x(\alpha)$* to be the cycle formed by the canonical path and the associated back edge. Denote the set of all canonical cycles by $\mathcal{C}$. A few important facts about this graph are noted. First, the length of each canonical cycle is $(3M+1)q \leq 4M\lambda k^2$. Second, for each accepting configuration α, there are X edge-disjoint canonical cycles associated with α. Finally, the degree of each vertex is at most 3. Figure 3 illustrates this final construction.

We set $X = 2^{2^{2\lambda(k^2+4k)}}$ and $M = 2^{\lambda(k^2+k)}$ in the final construction. Then $X = 2^{\text{polylog}(n)}$ and $M = \text{polylog}(n)$. Each vertex and edge participate in at least one canonical cycle and $|\mathcal{C}| \leq X \cdot 2^{\lambda r} \cdot 2^{\lambda(2k-1)}$ with the length of each cycle in $\mathcal{C}$ being bound by $4M\lambda k^2$. Denoting the number of vertices in the final construction by N we have $N \leq X \cdot 2^{\lambda r} \cdot M \cdot 2^{2\lambda k} \leq X \cdot 2^{O(\log n \log \log n)}$.
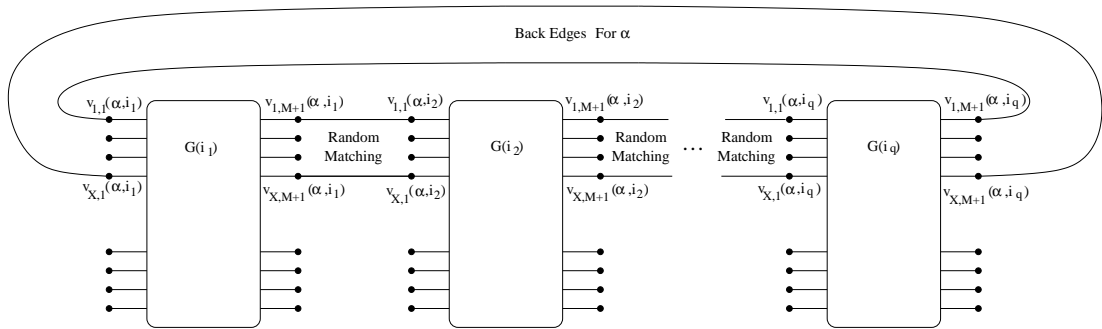

Figure 3: The final instance for configuration α.

3.4 Analysis

Here we show that if Φ is a satisfiable instance of 3SAT then there are many edge-disjoint cycles in the instance we built (those corresponding to the canonical cycles). On the other hand if Φ is a no-instance then the number of

edge-disjoint cycles is small. For this part we show that the number of canonical as well as non-canonical cycles is small.

3.4.1 Φ is Satisfiable

If Φ is satisfiable, then there exists a proof Π' for which the probability of acceptance of verifier V is at least $2^{-\lambda}$. For each of the at least $2^{\lambda r - \lambda}$ random strings R that result in verifier V accepting proof Π', choose all of the X canonical cycles corresponding to the configuration $(R, a_1, \ldots, a_q)$ where the a_j's, $1 \le j \le q$, are the values of the bits read in proof Π' when the random string is R. It is easy to see that the set of all these canonical cycles are edge-disjoint. Denoting the number of edge-disjoint cycles when Φ is satisfiable by C_{YI}, we have $C_{YI} \ge X \cdot 2^{\lambda r - \lambda} \ge \frac{|\mathcal{C}|}{2^{2\lambda k}}$.

3.4.2 Φ is not satisfiable

Suppose that Φ is not satisfiable and let $\mathcal{C}'$ be a collection of edge-disjoint cycles of the constructed graph G. Define $g = 2^{2\lambda(k^2+k)}$. We say a cycle is short if its length is less than g; otherwise the cycle is called long. Partition $\mathcal{C}'$ into sets $\mathcal{C}_1, \mathcal{C}_2$, and $\mathcal{C}_3$ where $\mathcal{C}_1$ is the set of all canonical cycles in $\mathcal{C}'$, $\mathcal{C}_2$ is the set of long non-canonical cycles, and $\mathcal{C}_3$ is the set of short non-canonical cycles. We bound the sizes of each of $\mathcal{C}_1, \mathcal{C}_2$, and $\mathcal{C}_3$. The proofs of the following two lemmas are essentially the same as the the corresponding arguments in [9]. We skip repeating them here.

Lemma 3.2 *If bad event $\mathcal{B}_1$ does not happen, then $|\mathcal{C}_1| \le \frac{2C_{YI}}{2^{\lambda k^2 - 2\lambda k - \lambda}}$.*

The number of long non-canonical cycles is at most $\frac{|E(G)|}{g} \le \frac{|\mathcal{C}| \cdot 4M\lambda k^2}{g}$. Since $g = 2^{2\lambda(k^2+k)}$ and $M = 2^{\lambda(k^2+k)}$ it follows that $g/(4M\lambda k^2) \ge 2^{\lambda k^2}$. Thus:

Lemma 3.3 $|\mathcal{C}_2| \le \frac{|\mathcal{C}|}{2^{\lambda k^2}} \le \frac{C_{YI}}{2^{\lambda k^2 - 2\lambda k}}$.

To bound the number of short non-canonical cycles we have to be more careful. For that we first define bad event $\mathcal{B}_2$ as the event $|\mathcal{C}_3| > \frac{C_{YI}}{2^{\lambda k^2}}$.

Lemma 3.4 *Event $\mathcal{B}_2$ happens with probability at most $\frac{1}{3}$.*

Proof. Let G' be the resultant graph when all of the special edges of G are contracted. An upper bound for the number of cycles of length less than g in G' is clearly an upper bound for the number of cycles of length less than g in G as well. Consider any length $g' < g$ and let us bound the number of non-canonical cycles of length g'. There are two types of edges in G': those that come from random matchings in G and those that are back-edges in G.

Claim 3.5 *The probability of each edge $e = uv$ appearing in the graph G' given the existence of $g' - 1$ other edges that do not form a canonical path from u to v, is at most $\frac{1}{X - g' + 1}$.*

This is easy to see for the case of a non-back-edge (*i.e.* random matching edge) as each matching edge exists with probability at most $\frac{1}{X - g' + 1}$ given the existence of $g' - 1$ other edges. The case of a potential back-edge is different as the back-edges are not completely random (each is created between the source and sink of a canonical path; but the path is created randomly). Consider a potential back edge $e = uv$ between a source node u and a sink node v (note that u and v are not necessarily the end points of a canonical path) and suppose we are given the

existence of up to $g' - 1$ other edges that do not form a canonical path from u to v. Moreover, consider the partial canonical paths from u and from v using the other at most $g' - 1$ other edges. Since there is currently no canonical path from u to v (otherwise we have a canonical cycle with e), then the probability that u and v are endpoints of the same canonical path is exactly the probability that they will be connected with a new random-matching edge. Thus, the probability that e exists is at most $\frac{1}{X - g' + 1}$. Using these arguments, for any potential non-canonical cycle C of length g' the probability that all edges of C exist is at most $(\frac{1}{X - g' + 1})^{g'} \leq (2/X)^{g'}$. A coarse upper bound on the number of potential cycles of length g' in G' is $N^{g'}$ which yields that the expected number of non-canonical cycles of length g' being no more than $\left(\frac{2N}{X}\right)^{g'}$. Summing over all $g' < g$, this yields an upper bound of $\left(\frac{2N}{X}\right)^{g}$ on the expected number of short non-canonical cycles.

Since $N \leq X \cdot 2^{\lambda r + 2\lambda k + \lambda(k^2 + k)}$, the expected number of cycles of length less than g is at most $2^{\lambda g(r + k^2 + 4k)} \leq 2^{3\lambda rg}$. By Markov's inequality, the probability that the number of cycles of length less than g is greater than $2^{4\lambda rg} \geq 3 \times 2^{3\lambda rg}$ is at most $\frac{1}{3}$. $\qquad \square$

Therefore, if event $\mathcal{B}_2$ does not happen, then:

$$|\mathcal{C}_3| \leq 2^{4\lambda rg} \leq 2^{2^{2\lambda(k^2 + 3k) + \log \log n}}$$

because $r = O(\log n)$. Also, since $\lambda = \beta \log \log n / k^2$ for $\beta >> k^2$, then $\lambda k \geq \log \log n$ resulting in

$$|\mathcal{C}_3| \leq 2^{2^{2\lambda(k^2 + 4k)}} \leq X \leq \frac{C_{YI}}{2^{\lambda(r-1)}} \leq \frac{C_{YI}}{2^{\lambda k^2}}.$$

3.4.3 Wrap up

If neither of bad events $\mathcal{B}_1$ nor $\mathcal{B}_2$ happens, then $|\mathcal{C}'| = |\mathcal{C}_1| + |\mathcal{C}_2| + |\mathcal{C}_3| \leq \frac{C_{YI}}{2^{\lambda(k^2 - 3k)}}$. So the gap between the size of the solution of G for the case that Φ is a yes-instance and for the case that Φ is a no-instance of 3SAT is $\Omega(2^{\lambda(k^2 - 3k)})$. Remembering that $N \leq X \cdot 2^{\lambda r} \cdot M \cdot 2^{2\lambda k}$, we have $\log N \leq 2^{2\lambda(k^2 + 4k)} + 3\lambda r$. By selecting β a large constant we have $\log N \leq 2^{2\lambda(k^2 + 5k)}$ which yields $\sqrt{\log N} \leq 2^{\lambda(k^2 - 3k)} \cdot 2^{8\lambda k} = \left(2^{\lambda(k^2 - 3k)}\right)^{1 + \frac{8}{k-3}}$. Therefore, $2^{\lambda(k^2 - 3k)} = \log^{\frac{1}{2} - (\frac{4}{k+5})}$ and so for any $\epsilon > 0$, we can choose $k = k(\epsilon) > 0$ such that the gap is at least $\log^{\frac{1}{2} - \epsilon} N$.

The probability of either of events $\mathcal{B}_1$ or $\mathcal{B}_2$ occurring is at most $1/(\text{poly}(n)) + 1/3 \leq 1/2$. So, if a $(\log^{\frac{1}{2} - \epsilon} n)$-approximation algorithm exists for the edge-disjoint cycles problem for any $\epsilon > 0$, then a co-RPTIME$(n^{\text{polylog}(n)})$ algorithm for 3SAT exists, which in turn implies the existence of a ZPTIME$(n^{\text{polylog}(n)})$ algorithm for 3SAT by a standard result. Thus, for any $\epsilon > 0$, it is hard to approximate the edge-disjoint cycle packing problem within a factor of $\Omega(\log^{\frac{1}{2} - \epsilon} n)$ unless NP $\subseteq$ ZPTIME$(n^{\text{polylog}(n)})$.

3.5 The Hardness of VDC

Since each vertex has degree at most 3 in the construction of G, an edge-disjoint cycle packing must also be a vertex-disjoint cycle packing. Also, a vertex-disjoint cycle packing must be an edge-disjoint cycle packing. Therefore, the preceding construction and analysis also shows that it is hard to approximate the vertex-disjoint cycle packing problem to within a factor of $\Omega(\log^{\frac{1}{2} - \epsilon} n)$ for any $\epsilon > 0$ unless NP $\subseteq$ ZPTIME$(n^{\text{polylog}(n)})$. The integrality gap for EDC holds for VDC by the same reasoning.

4 Concluding Remarks

Theorem 1.2 together with the results of [14, 15] yield an almost tight ratio for approximability of EDC in the undirected setting ($O(\sqrt{\log n})$ v.s. $\Omega(\log^{\frac{1}{2}-\epsilon} n)$ for any $\epsilon > 0$). However, the gap between the best approximation ratio and hardness lower bounds for undirected VDC as well as directed EDC (and VDC) are pretty wide; as said earlier, the best upper and lower bounds for undirected VDC are $O(\log n)$ and $\Omega(\log^{\frac{1}{2}-\epsilon} n)$, and for directed EDC are $O(\sqrt{n})$ and $\Omega(\log^{1-\epsilon} n)$ (for any $\epsilon > 0$), respectively. The bounds for directed EDC are in the same ballpark as the upper and lower bounds for approximability of the edge-disjoint paths problem in the *undirected* setting; this does not seem to be coincidental as both the approximation algorithms and the lower bound techniques used for these problems are similar in nature and we seem to need substantial new ideas to improve upon any of these. As we mentioned in the Introduction, what is interesting to note is that, given that our hardness result for EDC uses essentially the same construction for hardness result for EDP, any new improved hardness of approximation for EDP (beyond $O(\sqrt{\log n})$) needs substantially new ideas that exploit the differences between EDP and EDC problems since the best ratio for undirected EDC essentially meets the current hardness for undirected EDP.

References

[1] M. Andrews, J. Chuzhoy, S. Khanna and L, Zhang, *Hardness of the undirected edge-disjoint paths problem with congestion*, In Proc. of 46th IEEE FOCS, (2005), pp. 226–244.

[2] M. Andrews and L. Zhang, *Hardness of the undirected edge-disjoint paths problem*, In Proc. of 37th ACM STOC, ACM Press (2005), pp. 276–283.

[3] P. Balister, *Packing digraphs with directed closed trials*, Combin. Probab. Comput. 12 (2003), pp. 1–15.

[4] A. Caprara, A. Panconesi and R. Rizzi, *Packing cuts in undirected graphs*, J. Algorithms 44 (1999), pp. 1–11.

[5] A. Caprara, A. Panconesi and R. Rizzi, *Packing cycles in undirected graphs*, J. Algorithms 48 (2003), pp. 239–256.

[6] C. Chekuri and S. Khanna, *Edge disjoint paths revisited*, In Proc. of 14th ACM-SIAM SODA (2003), pp. 628–637.

[7] C. Chekuri, S. Khanna, and B. Shepherd, *An $O(\sqrt{n})$ Approximation and Integrality Gap for Disjoint Paths and UFP*, Theory of Computing, Vol 2 (2006), pp. 137–146.

[8] J. Chuzhoy, S. Guha, E. Halperin, S. Khanna, G. Kortsarz, R. Krauthgamer, and S. Naor. *Tight lower bounds for the asymmetric k-center problem*, Journal of ACM, 52(4):538-551, 2005

[9] J. Chuzhoy and S. Khanna, *New hardness results for undirected edge disjoint paths*, Manuscript, 2005.

[10] D. Dor and M. Tarsi, *Graph decomposition is NPC – A complete proof of Holyer's conjecture*, In Proc. of 20th ACM STOC, ACM Press (1992), pp. 252–263.

[11] 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*, J. of Computer and System Sciences 67(3):473–496, 2003. Earlier version in STOC'99.

[12] M. M. Halldorsson, G. Kortsarz, J. Radhakrishnan and S. Sivasubramanian, *Complete Partitions of Graphs*, Combinatorica, to appear.

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

[14] M. Krivelevich, Z. Nutov and R. Yuster, *Approximation algorithms for cycle packing problems*, In Proc. of 16th ACM-SIAM SODA (2005), pp. 556–561.

[15] M. Krivelevich, Z. Nutov, M. R. Salavatipour, J. Verstraete, and R. Yuster, *Approximation Algorithms and Hardness Results for Cycle Packing Problems*, To appear in ACM Transactions on Algorithms.

[16] M. R. Salavatipour, J. Verstraete, *Disjoint cycles: Integrality gap, hardness, and approximation*, In Proc. of IPCS 2005, LNCS volume 3509, pp. 51–65.

[17] A. Samorodnitsky and L. Trevisan, *A PCP characterization of NP with optimal amortized query complexity*, In Proc. of 32nd ACM STOC, ACM Press(2000), pp. 191–199.

[18] P. D. Seymour, *Packing directed circuits fractionally*, Combinatorica 15 (1995), pp. 281–288.

[19] K.R. Varadarajan and G. Venkataraman, *Graph decomposition and a greedy algorithm for edge-disjoint paths*, In Proc. of 15 ACM-SIAM SODA (2004), pp. 379–380.