# Disjoint Cycles: Integrality Gap, Hardness, and Approximation

Mohammad R. Salavatipour[1] * and Jacques Verstraete[2]**

[1] Department of Computing Science,
University of Alberta,
Edmonton, AB, Canada T6G 2E8
mreza@cs.ualberta.ca
[2] Department of Combinatorics and Optimization,
University of Waterloo,
Waterloo, ON, Canada N2L 3G1
jverstra@math.uwaterloo.ca

**Abstract.** In the edge-disjoint cycle packing problem we are given a graph $G$ and we have to find a largest set of edge-disjoint cycles in $G$. The problem of packing vertex-disjoint cycles in $G$ is defined similarly. The best approximation algorithms for edge-disjoint cycle packing are due to Krivelevich et al. [16], where they give an $O(\sqrt{\log n})$-approximation for undirected graphs and an $O(\sqrt{n})$-approximation for directed graphs. They also conjecture that the problem in directed case has an integrality gap of $\Omega(\sqrt{n})$. No non-trivial lower bound is known for the integrality gap of this problem. Here we show that both problems of packing edge-disjoint and packing vertex-disjoint cycles in a directed graph have an integrality gap of $\Omega(\frac{\log n}{\log \log n})$. This is the first super constant lower bound for the integrality gap of these problems. We also prove that both problems are quasi-NP-hard to approximate within a factor of $\Omega(\log^{1-\epsilon} n)$, for any $\epsilon > 0$. For the problem of packing vertex-disjoint cycles, we give the first approximation algorithms with ratios $O(\log n)$ (for undirected graphs) and $O(\sqrt{n})$ (for directed graphs). Our algorithms work for the more general case where we have a capacity $c_v$ on every vertex $v$ and we are seeking a largest set $\mathcal{C}$ of cycles such that at most $c_v$ cycles of $\mathcal{C}$ contain $v$.

## 1 Introduction

We study approximation algorithms, lower and upper bounds for the integrality gaps, and hardness results for the problems of packing disjoint cycles in a graph (directed or undirected). In the problem of packing edge-disjoint cycles (EDC ), we are given a graph $G$ (which can be directed or undirected) and we have to find a largest set of edge-disjoint cycles in $G$. The problem of packing vertex-disjoint

* Supported by a faculty startup grant, Department of Computing Science, University of Alberta.
** Supported by NSERC

cycles (VDC ) is defined similarly: find the maximum number of vertex-disjoint cycles in $G$. The EDC problem has been studied extensively both in undirected and directed graphs (see [16, 4, 6, 21]). Both EDC and VDC are fundamental problems in graph theory with applications in several areas (see the discussion in [4] for an application in computational biology and reconstruction of evolutionary trees). Let's denote by $\nu_e(G)$ and $\nu_v(G)$ the sizes of largest collections of edge-disjoint and vertex-disjoint cycles in $G$, respectively. It is well known that computing $\nu_e(G)$ and $\nu_v(G)$ are NP-hard even for undirected graphs. This motivates the study of approximation algorithms for these problems. An algorithm is called an $\alpha$-approximation for a maximization problem if the solution returned by the algorithm is at least a factor $1/\alpha$ of the optimal solution, and $\alpha$ is called the approximation (or performance) ratio of the algorithm. A natural generalization of EDC is the problem of $S$-cycle packing (denote by $s$-EDC ). In this problem, along with $G$ we are given a subset $S$ of vertices of $G$ and the goal is to find maximum number of edge-disjoint $S$-cycles in $G$, i.e. cycles each of which contains a vertex of $S$. The analogous problem of $s$-VDC is defined as finding maximum number of cycles in $G$ each of which contains a vertex of $S$ and are disjoint on the vertices in $V - S$.

**Known results:** Carpara et al. [6] showed that a simple greedy algorithm yields an $O(\log n)$-approximation for computing $\nu_e(G)$ and that the problem is APX-hard even for planar graphs (i.e. for an absolute constant $\epsilon_0 > 0$, no $(1+\epsilon_0)$-approximation exists unless P=NP). Very recently, Krivelevich et al. [16] showed that the greedy algorithm of [6] actually yields an $O(\sqrt{\log n})$-approximation for EDC and gave examples to show that this is tight. In fact they proved an upper bound of $O(\sqrt{\log n})$ for the integrality gap of EDC . For directed graphs, they gave an $O(\sqrt{n})$-approximation for EDC and an $O(n^{\frac{2}{3}})$-approximation for $s$-EDC . Their algorithm for directed EDC shows that the integrality gap of EDC in directed graphs is at most $O(\sqrt{n})$. No non-trivial lower bounds for the integrality gap of EDC in directed setting is known. The authors in [16] conjectured that there is a lower bound of $\Omega(\sqrt{n})$ for this integrality gap.

**Related results:** The dual problems of packing cycles (known as feedback sets problems) are also very well studied problems in both directed and undirected settings. The dual problem of VDC , known as Feedback Vertex Set (FVS), is the problem of finding minimum number of vertices in a graph whose removal makes the graph acyclic. This problem and its generalization (in which every vertex has a weight and we seek to minimize the total weight of selected vertices) has 2-approximation algorithms in undirected graphs (see [3, 5, 10]). The problem of finding minimum number of edges in a graph that meet every cycle (FES) is trivial for undirected graphs (complement of a spanning tree). For directed graphs, there is an easy reduction from FES to FVS. Seymour [21] showed that, if the optimal fractional FVS in a directed graph $G$ has value $\varphi^*$ then the optimal integral FVS in $G$ has value at most $O(\varphi^* \log \varphi^* \log\log \varphi^*)$. This yields an $O(\log \varphi^* \log\log \varphi^*)$-approximation algorithm for FVS in directed graphs [13]. Alon and Seymour (see [21]) showed that the integrality gap of FVS is $\Omega(\log \varphi^*)$.

As far as we know, the only hardness result for the problems of packing disjoint cycles is the the APX-hardness result proved in [6] for undirected EDC , and no better hardness or even a super constant lower bound for the integrality gap of EDC in directed graphs is known. As mentioned earlier, Krivelevich et al. [16] have conjectured that this gap is $\Omega(\sqrt{n})$. This conjecture seems conceivable, given the fact that the well known similar problem of edge-disjoint paths (EDP) has such an integrality gap (even in undirected case). Here we take the first step toward tightening this gap. Our main result is that EDC in directed setting has an integrality gap of $\Omega(\frac{\log n}{\log\log n})$. More importantly, we prove that it is quasi-NP-hard to approximate $\nu_e(G)$ within a factor of $\Omega(\log^{1-\epsilon} n)$ for any $\epsilon > 0$. Under stronger complexity assumptions we can prove a slightly better hardness of $\Omega(\frac{\log n}{\log\log n})$. As we will see, there are quite easy reductions between EDC and VDC in directed graphs. Therefore, our hardness result for EDC carries over to VDC , i.e. VDC in directed graphs is hard to approximate within a factor of $\Omega(\log^{1-\epsilon} n)$.

We also consider the capcitated version of cycle packing problems. In the capacitated version of VDC , we are given a graph $G(V,E)$ with a positive capacity $c_v$ on every vertex $v \in V$ and the goal is to find a largest collection $\mathcal{C}$ of cycles in $G$ such that each vertex $v$ belongs to at most $c_v$ cycles of $\mathcal{C}$. We abuse the notation slightly by calling this capacitated version $\mathrm{VDC}_c$ , although the cycles are not required to be vertex-disjoint anymore. For undirected $\mathrm{VDC}_c$ , we give an $O(\log n)$-approximation. This also shows an upper bound of $O(\log n)$ on the integrality gap of $\mathrm{VDC}_c$ in undirected graphs. We are not aware of any earlier approximation algorithm for this problem. For directed $\mathrm{VDC}_c$ , we show that a simple randomized rounding algorithm (similar to the ones we presented in [9]) yields an $O(\sqrt{n})$-approximation. The same algorithm works for the even more general case of capacitated (directed) $s$-VDC , where every vertex $v \in V - S$ has a capacity $c_v$ and we have to find a largest collection $\mathcal{C}$ of $S$-cycles (i.e. cycles that intersect $S$) such that each vertex $v \in V - S$ belongs to at most $c_v$ cycles. We denote this problem by $s$-$\mathrm{VDC}_c$ . Note that for the $s$-EDC [16] gave an $O(n^{\frac{2}{3}})$ greedy approximation algorithm. It can be shown (as we will see) that this upper bound is tight for their greedy algorithm.

**Remark:** For the capacited version of EDC (denoted by $\mathrm{EDC}_c$ ), as noted by the authors in [16], their results for EDC can be easily extended to an $O(\sqrt{\log n})$-approximation algorithm for undirected $\mathrm{EDC}_c$ and an $O(\sqrt{n})$-approximation for directed $\mathrm{EDC}_c$ .

Throughout the paper, we use $n$ to denote the number of vertices of the input graph.

## 2 Approximation Algorithms

Let $G(V,E)$ be the given graph with a capacity $c_v$ for every vertex $v \in V$, and let $\mathcal{C}(G)$ denote the set of all cycles of $G$. The following is the standard integer program (IP) formulation of $\mathrm{VDC}_c$ :

$$\begin{array}{ll} \text{maximize} & \sum_{C\in\mathcal{C}} x_C \\ \text{subject to} & \forall v \in V : \sum_{C: v\in C} x_C \le c_v \\ & \forall C \in \mathcal{C} : \quad x_C \in \{0,1\} \end{array} \tag{1}$$

Trivially, for VDC all $c_v$ values are equal to 1. The fractional $\text{VDC}_c$ problem is the linear program relaxation of this IP. The integer program formulation for $\text{EDC}_c$ is defined similarly. Here, we will have a constraint of the form $\sum_{C:e\in C} x_C \le c_e$, for every edge $e \in E$. For the EDC problem, all the capacities are 1. Clearly, the solutions to the fractional problems (LP's) are upper bounds to the solutions for the corresponding integral problems (IP's). Let $\nu_v^*(G)$ and $\nu_e^*(G)$ denote the values of optimal fractional solutions for VDC and EDC , respectively.

**Theorem 1.** *There is an $O(\log n)$-approximation for (undirected) $VDC_c$ .*

*Proof.* We are going to incorporate a technique from [18] with the simple greedy algorithm, similar to the one for the EDC in [6, 16]. The idea of this greedy algorithm was implicit in [11].

Consider the problem of fractional $\text{VDC}_c$ , i.e. the LP corresponding to the IP given in (1). Note that the separation oracle for the dual of this LP is the problem of finding a shortest cycle in a weighted graph. Since this problem can be solved in polynomial time, using the same method as in Theorem 4.1 of [15] (or [7]) we can solve the primal LP in polynomial time. Let $\varphi(G)$ and $\varphi^*(G)$ denote the values of optimal solution to the IP and the corresponding LP for $\text{VDC}_c$ . Let $Y = \{x_1, \ldots, x_p\}$ be the set of primal variables that have value $> 0$ in the optimal fractional solution. One of the features of the algorithm of [15] is that $p$ (the number of fractional cycles) is polynomial in $n$, even-though the LP has an exponential number of variables. If $\sum_{i=1}^p \lfloor x_i \rfloor \ge \frac{1}{\log n} \sum_{i=1}^p x_i$ then $Y' = \{\lfloor x_1 \rfloor, \ldots, \lfloor x_p \rfloor\}$ is an integral solution with value at least $\frac{\varphi^*(G)}{\log n}$, which is at least $\frac{\varphi(G)}{\log n}$. In this case the algorithm returns the cycles corresponding to the variables in $Y'$ and stops. Otherwise, if $\sum_{i=1}^p \lfloor x_i \rfloor < \frac{1}{\log n} \sum_{i=1}^p x_i$ then

$$\varphi^*(G) = \sum_{i=1}^p x_i = \sum_{i=1}^p \lfloor x_i \rfloor + \sum_{i=1}^p (x_i - \lfloor x_i \rfloor) < \frac{\varphi^*(G)}{\log n} + p.$$

Therefore, with $Q = p \cdot (\frac{\log n}{\log n - 1})$: $\varphi^*(G) < Q$. This implies that for every vertex $v \in G$, at most a value of $\min\{c_v, Q\}$ of capacity of $v$ is used in any optimal (fractional or integral) solution. So we can decrease the capacity $c_v$ of every vertex $v$ to $\min\{c_v, Q\}$.

We are also going to assign capacities to the edges of $G$. Initially, every edge has infinite capacity. Throughout the algorithm, we will replace a vertex $v$ of degree 2 with neighbors $u$ and $w$ with an edge between $u$ and $w$ with capacity $c_v$. We perform the following algorithm on $G$ as long as $G$ has a cycle. The algorithm is based on the greedy algorithm proposed in [6] for the EDC . Initially $\mathcal{C} = \emptyset$.

1. While $G$ contains a vertex $v$ of degree $\leq 1$ or with capacity 0 delete $v$ (and all the edges incident with it).
2. While $G$ contains a vertex $v$ of degree 2 with neighbors $u$ and $w$ delete $v$ and add an edge $uw$ with capacity $c_v$ to $G$.
3. Find a shortest cycle $C$ in $G$ and add it to $\mathcal{C}$, decrease the capacity of every vertex in $C$ and every edge with finite capacity in $C$ by 1. Go to step 1.

It is easy to see that steps 1 and 2 don't change the value of an optimal solution. Also, since the capacity of every vertex is polynomial in $n$, the size of the graph is always a polynomial factor of the initial size of $G$. Let $S_i$ be the $i$'th iteration in which we perform step 3 and let $G_i$ be the graph at the beginning of iteration $S_i$, and $n_i = |G_i|$. It is well known that every graph with minimum degree at least 3 on $n$ vertices has girth (size of the shortest cycle) at most $O(\log n)$ [11]. Since graph $G_i$ has minimum degree 3, the girth of $G_i$ is at most $O(\log n_i)$. Therefore, the cycle found in step $S_i$ intersects at most $O(\log n_i)$ (which is $O(\log n)$) cycles of the optimal fractional solution. This is true for every step $S_i$. Thus, $\varphi^*(G) \leq O(\log n)|\mathcal{C}|$, i.e. the algorithm is an $O(\log n)$ approximation. □

Our next theorem gives an $O(\sqrt{n})$-approximation algorithm for directed $s$-VDC$_c$ . First we show that the greedy algorithm of [16] for $s$-EDC (and its adapted version for $s$-VDC ) will have a ratio of at least $\Omega(n^{\frac{2}{3}})$. We give the construction of a graph $G$ which is a simple modification of a construction given by Chekuri and Khanna [8] for the problem of edge-disjoint paths (EDP) in directed graphs. $G$ consists of two layered graphs $G_1$ and $G_2$, where $G_1$ contains layers $X_1, \ldots, X_q$ and $G_2$ contains layers $Y_1, \ldots, Y_q$ of vertices, with $q = n^{\frac{2}{3}}$. Each $X_i$ and $Y_i$ has $n^{\frac{1}{3}}$ vertices and every vertex in $X_i$ (in $Y_i$), $1 \leq i < q$, is connected by an edge to every vertex in $X_{i+1}$ (in $Y_{i+1}$). For every odd value of $i \leq q-2$ pick a representative vertex $x_i^r$ from $X_i$ and one $y_{i+1}^r$ from $Y_{i+1}$. Connect $x_i^r$ to $y_{i+1}^r$ and connect $y_{i+1}^r$ to $x_{i+2}^r$. Finlay put an edge between every vertex in $Y_q$ to every vertex in $X_1$. Let $S = X_1$. It is easy to verify that $G$ has $\Omega(n^{\frac{2}{3}})$ edge-disjoint $S$-cycles (and they are in fact vertex disjoint on $V - S$). If the greedy algorithm picks in its first $S$-cycle all the edges between $G_1$ and $G_2$ then it finds only one cycle (since the edges from $G_1$ to $G_2$ form a cut of size $O(q)$ from $G_1$ to $G_2$). Therefore, the ratio of greedy algorithm proposed in [16] for $s$-EDC (and its analogous for $s$-VDC ) is at least $\Omega(n^{\frac{2}{3}})$.

**Theorem 2.** *There is an $O(\sqrt{n})$-approximation algorithm for directed $s$-VDC$_c$ .*

*Proof.* The proof of this theorem follows the same steps as the approximation algorithms of [9] for disjoint Steiner trees in directed graphs. Let $G(V, E)$, set $S \subseteq V$, and capacity $c_v$ for every vertex $v \in V - S$ be the given instance of $s$-VDC$_c$ . Consider the IP/LP formulation of the $s$-VDC$_c$ problem, which is the same as (1) given for VDC$_c$ except that $\mathcal{C}(G)$ will be the set of all $S$-cycles of $G$. Consider the optimal fractional solution which is again computed using the technique of [15] or [7]. We can compute this fractional solution in polynomial time since the separation oracle for the dual LP is the problem of finding a shortest $S$-cycle in a weighted graph, and this can be computed in polynomial time. Consider a solution to the LP of $s$-VDC$_c$ and let $X^* = \{x_1^*, x_2^*, \ldots x_{p(n)}^*\}$ be the set of fractional cycles in this optimal fractional solution, where $p(n)$ is some polynomial in $n$. For every $S$-cycle $C$ with fractional value $x_C^* \geq 1$ we "take out" $\lfloor x_C^* \rfloor$ copies of that cycle from the graph and put them in the final integral solution that we are computing. This way we will find a set $\mathcal{C}_1$ of size at least $\varphi_1^* = \sum_{i=1}^{p(n)} \lfloor x_i^* \rfloor$ integral $S$-cycles. We also decrease the capacity of every vertex in cycle $C$ by $\lfloor x_C^* \rfloor$, and replace $x_C^*$ with $x_C^* - \lfloor x_C^* \rfloor$ in the fractional solution. Now let $\varphi_2^* = \sum_{i=1}^{p(n)} x_i^*$. Note that the value of the solution to the LP of $s$-VDC$_c$ is $\varphi_1^* + \varphi_2^*$. We show how to find a set $\mathcal{C}_2$ of size at least $O(\frac{\varphi_2^*}{\sqrt{n}})$ integral cycles. The final solution will be $\mathcal{C}_1 \cup \mathcal{C}_2$ which clearly has size at least $O(\varphi_1^* + \frac{\varphi_2^*}{\sqrt{n}})$, and since the value of the solution to the original LP is $\varphi_1^* + \varphi_2^*$, we get an $O(\sqrt{n})$-approximation.

If $\varphi_2^* \leq 30\sqrt{n}$ then it is enough to find just one $S$-cycle in $G$ and place it in $\mathcal{C}_2$. So let's assume that $\varphi_2^* > 30\sqrt{n}$. For every cycle $C \in \mathcal{C}(G)$, pick that cycle with probability $x_c^*/\lambda$ for a $\lambda > 0$ to be defined soon. Define $Y_C$ to be random variable that is 1 if and only if cycle $C$ is selected. So for $Y = \sum_{C \in \mathcal{C}(G)} Y_C$ (i.e. total number of cycles placed in $\mathcal{C}_2$), we have:

$$E[Y] = \sum_{C \in \mathcal{C}(G)} Pr[Y_C = 1] = \sum_{C \in \mathcal{C}(G)} \frac{x_C^*}{\lambda} = \frac{\varphi_2^*}{\lambda}.$$

Define the bad event $A_v$ to be the event that more than $c_v$ cycles containing vertex $v \in V - S$ are selected. We can show that with positive probability none of these events happens (so no vertex capacity is violated) *and* that the number of cycles selected is at least $O(\varphi_2^*/\lambda)$. We borrow the following lemma from [9]:

**Lemma 1.** *[9] Assume that $A = \{a_1, \ldots, a_n\}$ is a set of $n$ non-negative reals and let $\mathcal{A}_k$ be the set of all subsets of size $k$ of $A$. If $\sum_{i=1}^n a_i \leq Q$ , then $\sum_{\{a_{i_1}, \ldots, a_{i_k}\} \in A_k} a_{i_1} a_{i_2} \ldots a_{i_k} \leq \binom{n}{k} (Q/n)^k$.*

For every vertex $v \in V - S$, denote the number of fractional cycles $C$ with $x_C^* > 0$ that contain $v$ by $\psi_v$. By this definition:

$$\Pr[A_v] \leq \sum \prod_{i=1}^{c_v+1} x_{C_{a_i}}^* / \lambda,$$

where the summation is over all subsets $\{C_{a_1}, \ldots, C_{a_{c_v+1}}\}$ of size $c_v + 1$ of cycles with $x_{C_{a_i}}^* > 0$ that contain vertex $v$. Therefore, using Lemma 1:

$$\Pr[A_v] \leq \binom{\psi_v}{c_v + 1} \left(\frac{c_v}{\lambda \psi_v}\right)^{c_v+1} \leq \left(\frac{v\psi_v}{c_v + 1}\right)^{c_v+1} \left(\frac{c_v}{\lambda \psi_v}\right)^{c_v+1} \leq \frac{e^2}{\lambda^2},$$

where we have used the fact $\binom{n}{k} \leq (\frac{en}{k})^k$ for the second inequality. It is intuitively clear that if $\overline{A_v}$ holds then it does not increase the probability of any other $A_{v'}$. In other words, events $\overline{A_v}$ are "positively correlated". Therefore: $\Pr[\bigwedge_{v\in V-S}\overline{A_v}] \geq \prod_{v\in V-S}\Pr[\overline{A_v}] \geq (1-\frac{e^2}{\lambda^2})^n$. Also, by Chernoff bound, for $0 \leq \delta < 1$: $\Pr[Y < (1-\delta)\mathrm{E}[Y]] \leq e^{-\delta^2\varphi_2^*/2\lambda}$. Thus:

$$\Pr[(Y < (1-\delta)\mathrm{E}[Y]) \vee (\exists v \in V-S : A_v)] \leq e^{-\delta^2\varphi_2^*/2\lambda} + 1 - (1-e^2/\lambda^2)^n.$$

If we show that for suitable $\delta$ and $\lambda$: $(1-e^2/\lambda^2)^n > e^{-\delta^2\varphi_2^*/2\lambda}$ then using the method of conditional probability, we can efficiently find a selection $\mathcal{C}_2$ of $S$-cycles such that $|\mathcal{C}_2| \geq (1-\delta)\varphi_2^*/\lambda$ and that no vertex capacity constraint is violated. If $\varphi_2^* \leq n$ then with $\delta = \frac{1}{2}$ and $\lambda = e\sqrt{n}$ we find a collection $\mathcal{C}_2$ of $S$-cycles that obey the capacity constraints of vertices with $|\mathcal{C}_2| \geq \varphi_2^*/2e\sqrt{n}$. If $\varphi_2^* > n$ then there is a constant $\sigma > 0$ such that with $\delta = \frac{1}{2}$ and $\lambda = \sigma$: $(1-e^2/\lambda^2)^n > e^{-\delta^2\varphi_2^*/2\lambda}$. Again, we can find a collection $\mathcal{C}_2$ of $S$-cycles that satisfy the capacity constraints of vertices and $|\mathcal{C}_2| \geq \frac{\varphi_2^*}{2\sigma}$. In any case, we find a set $\mathcal{C}_2$ of size at least $\Omega(\frac{\varphi_2^*}{\sqrt{n}})$. Therefore, the algorithm is an $O(\sqrt{n})$-approximation for $s$-VDC$_c$ . □

## 3 Integrality Gap and Hardness of Directed EDC and VDC

In this section, we prove that each of EDC and VDC has an integrality gap of $\Omega(\frac{\log n}{\log\log n})$. Furthermore, each of $\nu_v(G)$ and $\nu_e(G)$ is quasi-NP-hard to approximate within a factor of $O(\log^{1-\epsilon} n)$, for any $\epsilon > 0$. First, we provide approximate preserving reductions between EDC and VDC .

**Theorem 3.** *Given a directed graph $G(V,E)$ as an instance of VDC (of EDC ) there is an instance $G'(V',E')$ of EDC (of VDC ) with $|G'| = poly(|G|)$, such that $G$ has $k$ vertex-disjoint cycles (edge-disjoint cycles) if and only if $G'$ has $k$ edge-disjoint cycles (vertex-disjoint cycles).*

*Proof.* 1st direction: For each node $v \in V$, $G'$ contains two nodes $v_1, v_2$. We add $v_1v_2$ to $E'$. Furthermore, for every edge $uv \in E$ we create an edge $u_2v_1$ in $E'$ and for every edge $vw \in E$ we create an edge $v_2w_1$ in $E'$. It is easy to see that if $\mathcal{C}$ is a collection of integral (or fractional) vertex-disjoint cycles in $G$ with size $k$ then there is a collection $\mathcal{C}'$ of $k$ integral (or fractional) edge-disjoint cycles in $G'$. Conversely, suppose that $\mathcal{C}'$ is a collection of $k$ integral (or fractional) edge-disjoint cycles in $G'$. Then for every edge $v_1v_2$ (corresponding to a vertex $v \in V(G)$) there is at most a total of one integral (or fractional) cycle(s) containing that edge. Therefore, by contracting the edges of the form $v_1v_2$ on each cycle of $\mathcal{C}'$ we obtain a collection of $k$ integral (or fractional) vertex-disjoint cycles in $G$.

2nd direction: Suppose $G$ is an instance of EDC . For every edge $xy$ in $G$ create a vertex $v_{xy}$ in $G'$. For every vertex $x \in G$ with ingoing edges $y_1x, y_2x, \ldots, y_px$ and outgoing edges $xz_1, xz_2, \ldots, xz_q$ add the following edges to $G'$: $v_{y_ix}v_{xz_j}$ for every $1 \leq i \leq p$ and $1 \leq j \leq q$. It can be seen that $G$ has $k$ edge-disjoint integral (or fractional) cycles if and only if if $G'$ has $k$ vertex-disjoint integral (or fractional) cycles. □

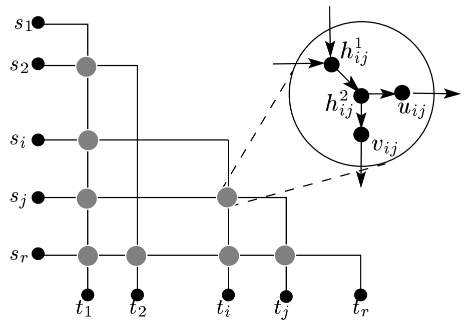


**Fig. 1.** Construction of $D_r$: each gray circle corresponds to an intersection module

So it is enough to prove our lower bound for integrality gap and the hardness result for EDC . Then we use Theorem 3 to deduce a similar result for VDC .

**Theorem 4.** *The directed EDC problem has an integrality gap of $\Omega(\frac{\log n}{\log\log n})$.*

We give the construction of a graph $G$ on $n$ vertices, such that $\frac{\nu_e^*(G)}{\nu_e(G)} \in \Omega(\frac{\log n}{\log\log n})$. Our starting point is a grid-like graph which gives the $\Omega(\sqrt{n})$ integrality gap for the well-known problems of disjoint paths. An instance of the edge-disjoint paths (EDP) problem consists of a (directed) graph $G$ with pairs of vertices $s_i, t_i$, for $1 \leq i \leq k$, and the goal is to connect maximum number of pairs $s_i, t_i$ using edge-disjoint paths. The vertex-disjoint paths (VDP) problem is defined similarly.

Let $r$ be a positive integer and define a directed graph which consists of vertices $s_i, t_i$ $(1 \leq i \leq r)$ together with vertices $h_{ij}, u_{ij}, v_{ij}$, $1 \leq j \leq i \leq r$. There is an edge from $h_{ij}$ to $u_{ij}$ and an edge from $h_{ij}$ to $v_{ij}$ $(1 \leq j \leq i \leq r)$. There are also edges $u_{ij}h_{i(j+1)}$ and $v_{ij}h_{(i+1)j}$ for $1 \leq j < i < r$. Furthermore, for every $1 \leq i < r$ it has edges $u_{ii}h_{(i+1)(i+1)}$, $s_ih_{i1}$, and $v_{ri}t_i$. Finally $u_{r(r-1)}$ is connected to $t_r$. Since this graph has a drawing on the plane, there cannot be two *vertex-disjoint* paths $P_i$ and $P_j$ $(1 \leq i \neq j \leq r)$ where $P_i$ starts from $s_i$ and ends in $t_i$ and $P_j$ starts from $s_j$ and ends in $t_j$. Because we want to have edge-disjoint property, we "split" every vertex $h_{ij}$ into two copies $h^1_{ij}$ and $h^2_{ij}$, where the ingoing edges of $h_{ij}$ are now going into $h^1_{ij}$ and the outgoing edges of $h_{ij}$ are going out of $h^2_{ij}$ and put the edge $h^1_{ij}h^2_{ij}$ in (see Figure 1). Let's call this graph $D_r$ and the subgraph induced by four vertices $h^1_{ij}, h^2_{ij}, u_{ij}, v_{ij}$ an *intersection module* of $D_r$. Again, it is easy to see that there cannot be two *edge-disjoint* paths from $s_i$'s to $t_i$'s (because we can route at most one path through every intersection module). Note that:

**Fact 1:** The half-integral fractional solution for EDP in $D_r$ has value $\geq \frac{r}{2}$.

This creates a gap of $\Omega(r)$, which is $\Omega(\sqrt{n})$, with $n$ being the number of vertices in the graph. We will use this fact again, later on. A natural attempt to extend this result to the cycle packing problem would be to add directed edges $t_is_i$, for $1 \leq i \leq r$. Unfortunately, this new graph will have an integral solution of

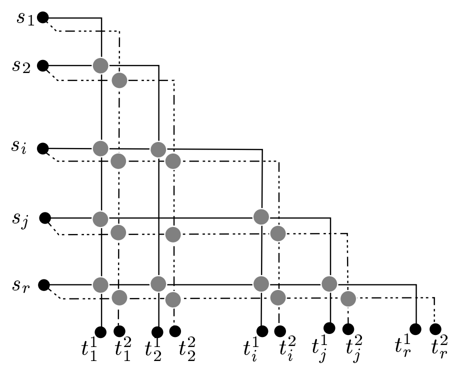


**Fig. 2.** Construction of $H_r$ from two copies of $D_r$

value $\Omega(r)$ (for e.g. consider the directed cycle that goes from $s_1$ to $t_r$ along the diagonal path, then to $s_r$ and to $t_1$ and back to $s_1$. We can pick $\frac{r}{4}$ such cycles). So this doesn't create the desired gap. The problem here is caused because the cycles are not bond to follow a path directly from $s_i$ to $t_i$ (they may go through other $s_j$'s and $t_j$'s before reaching $t_i$). Our idea to resolve this problem is to make it "too costly" for the cycles to do so. In other words, we are going to combine many copies of $D_r$ in a special manner so that if a cycle consist of a "non-trivial" path from $s_i$ to $t_i$ then it has a very long length; so long that we cannot have many of them. This will create the desired gap.

Using two copies of graph $D_r$ we construct another graph $H_r$ in the following way. Consider graph $D_r$ with input vertices $s_1, \ldots, s_r$ and output vertices $t_1, \ldots, t_r$. Take two copies of this graph, $D_r^1$ and $D_r^2$, and identify (only) the input vertices of them. Let $s_1, \ldots, s_r$ be the new set of (unified) input vertices and $t_1^1, \ldots, t_r^1$ and $t_1^2, \ldots, t_r^2$ be the set of output vertices. Let's call this graph $H_r$ (see Figure 2). An important observation to make here is that $H_r$ is acyclic. This is crucial to our main construction. We call the triple $s_i, t_i^1, t_i^2$ "block" $i$ with *start point* $s_i$ and *end points* $t_i^1, t_i^2$, $1 \le i \le k$. Consider $H_r$ and the $2r$ pairs $s_i, t_i^1$ and $s_i, t_i^2$ (two pairs for each block) as an instance of the EDP problem. We say *block $i$ is fully routed* in a solution to this instance if there are edge-disjoint paths connecting both pairs $s_i, t_i^1$ and $s_i, t_i^2$ in the solution. If only one of these paths exists in the solution then we say *block $i$ is partially routed*. It is easy to see:

**Fact 2:** Any optimal (integral) solution for EDP on $H_k$ with $2k$ pairs, either contains only one routed block or two partially routed blocks. Furthermore, there is a half-integral solution in which every block is fully routed (with value $\frac{1}{2}$ on each path).

We will use the following technical lemma in our construction.

**Lemma 2.** *For given positive integers $r$, $k$, and $g$ with $r < k$, there is an explicit construction for a $k$-uniform $r$-regular hypergraph of girth at least $g$ where the size of the construction (number of vertices) is $O(\frac{k^{2g}}{r})$.*

*Proof.* We start with an explicit $k$-regular graph $G(V,E)$ of size at most $O(k^{2g-1})$ and girth at least $2g$. These graphs exist (see, for instance, [17] and the references there). Construct a bipartite graph $G'(A \cup B, E')$ from $G$ where $A$ and $B$ are copies of $V$ with $a_i$ and $b_i$ being the vertices of $A$ and $B$ (respectively) corresponding to vertex $v_i \in V$, and $a_i b_j \in E'$ if and only if $v_i v_j \in E$. It is easy to see that $G'$ is $k$-regular with girth at least $2g$ and has size $O(k^{2g-1})$. To simplify our calculations, let's assume that $k$ is a power of 2 (this only affects the implicit multiplicative constant in the lemma).

Construct a new bipartite graph $G''(A' \cup B, E'')$ from $G'$ in this way: for each vertex $a_i \in A$ create two vertices $a_i^1$ and $a_i^2$ in $A'$. Then join $a_i^1$ to half of the neighbors of $a_i$ and join $a_i^2$ to the other half. Repeating this procedure $\log(k/r)$ times, we get the bipartite graph $G''$ in which every vertex in $A'$ has degree $\frac{k}{2^{\log(k/r)}} = r$ and every vertex in $B$ has degree $k$, $|A'| = \frac{k^{2g}}{r}$, $|B| = k^{2g-1}$, and the girth is $2g$. Now we define hypergraph $\mathcal{H}$ with vertex set $A'$ whose edge-set is the set of neighborhoods of vertices of $B$. This hypergraph is $k$-uniform, $r$-regular, with girth $g$ and has size $O(\frac{k^{2g}}{r})$. $\square$

**Proof of Theorem 4:** Let $r$, $k$, and $g$ be some positive integers to be specified later and let $r' = \binom{r}{2}$. Consider a $k$-uniform $r'$-regular girth $g$ hypergraph $\mathcal{H}$. Such graphs exist by Lemma 2. The underlying structure of the main graph for the integrality gap is hypergraph $\mathcal{H}$. Let $p$ and $q$ be the number of vertices and hyperedges of $\mathcal{H}$, respectively. Take a set $P_r = \{D_r^1, \ldots, D_r^p\}$ containing $p$ copies of graph $D_r$ (constructed earlier), one corresponding to each vertex in hypergraph $\mathcal{H}$. Also take a set $Q_k = \{R_k^1, \ldots, R_k^q\}$ containing $q$ copies of $H_k$, one corresponding to each hyperedge of $\mathcal{H}$. For every graph in $P_r$ we fix an arbitrary ordering of its intersection modules (note that the number of intersection modules of $D_r$ is $\binom{r}{2} = r'$; the same as degree of a vertex in $\mathcal{H}$). Similarly, for each graph in $Q_k$ we fix an arbitrary ordering of its blocks (note that the number of blocks of each graph in $Q_k$ is $k$; the same as the size of a hyperedge in $\mathcal{H}$). Initially, we assign a green flag to every intersection module of every graph in $P_r$ and to every block of every graph in $Q_k$. Soon we will start modifying the blocks and modules and change their flags to "red". For each pair $s_i^j, t_i^j$ in each copy $D_r^j \in P_r$ add the directed edge $t_i^j s_i^j$ to $D_r^j$. We call these edges *feedback* edges.

Consider an arbitrary hyperedge $e_i \in \mathcal{H}$ and let $R_k^i \in Q_k$ be the copy of $H_k$ in $Q_k$ which corresponds to hyperedge $e_i$. Note that $R_k^i$ has $k$ blocks; let's denote these blocks by $b_1^i, \ldots, b_k^i$, where $b_\lambda^i$ consist of triple $s_\lambda^i, t_\lambda^{1,i}, t_\lambda^{2,i}$. Furthermore, look at the vertices of hyperedge $e_i$ and find the corresponding copies of $D_r$ in $P_r$. More precisely, let

$$S_i = \{D_r^{a_j} \in P_r |\text{ the vertex of } \mathcal{H} \text{ corresponding to } D_r^{a_j} \text{ belongs to } e_i,\ 1 \le j \le k\}.$$

Pick the first green block of $R_k^i$, say $s_\lambda^i, t_\lambda^{i,1}, t_\lambda^{i,2}$ (for some $1 \le \lambda \le k$) according to the fixed ordering of the blocks of $R_k^i$ and change its flag to red. Also, pick the first green intersection module of $D_r^{a_\lambda}$ (from its fixed ordering), say $h_{ab}^1, h_{ab}^2, u_{ab}, v_{ab}$ (for some $1 \le a, b \le r$) and change its flag to red. Remove vertex $h_{ab}^2$ and its incident edges (i.e. edges $h_{ab}^1 h_{ab}^2$, $h_{ab}^2 u_{ab}$, and $h_{ab}^2 v_{ab}$) from $D_r^{a_\lambda}$ and add the following edges: $h^1_{ab}s^i_\lambda$, $t^{1,i}_\lambda u_{ab}$, and $t^{2,i}_\lambda v_{ab}$. We will consider these new three edges (instead of the three edges that were removed from $D^{a_\lambda}_r$) as part of $D^{a_\lambda}_r$. Do this for all the blocks of $R^i_k$. This process is going to modify (and change the flag from green to red for) one intersection module from each graph in $S^i$ (i.e. $D^{a_1}_r, \ldots, D^{a_K}_r$); one for every block of $R^i_k$. Repeat the same procedure for all the hyperedges of $\mathcal{H}$ (i.e. for all graphs in $Q_k$). We obtain a huge directed graph $G_{r,k,g}$, which has constant degree and $O(r^2p + k^2q)$ vertices. Note that, since each graph $R^i_k \in Q_k$ is acyclic, every cycle in $G_{r,k,g}$ must contain one of the feedback edges.

The basic idea behind the construction is that intersection modules in copies of $D_r$ (graphs in $P_r$) are now replaced with "blocks" of copies of $H_k$ (graphs in $Q_k$) and in order to go from $h^1_{ab}$ to $u_{ab}$ in the intersection module of $D^{a_\lambda}_r$, we have to go from $s^i_\lambda$ to $t^{1,i}_\lambda$ in a "block" of $R^i_k$. For the moment, assume that:

(i) all the blocks where completely independent of each other i.e. they were not part of the same graph and therefore there was no way to start at the start point of a block $b_i$ (of a copy of $H_k$) and end at an end point of another block $b_j$.
(ii) among all the blocks, only an $\epsilon$-fraction could be (partially or fully) routed, and for the other $(1-\epsilon)$-fraction no routing existed at all.

This would imply that $G_{r,k,g}$ has no more than $\epsilon rp$ cycles. The reason is that each cycle must contain a feedback edge, and so goes from some $s_\alpha$ in a copy of $D_r$ in $P_r$ to $t_\alpha$ in the same copy. Therefore, it goes through at least $r$ blocks (previously intersection modules), since each cycle in a copy of $D_r$ uses at least $r$ intersection modules. This would give us the required gap. Fortunately, the assumption (i) above is easy to prove (by Fact 2), i.e. a large fraction of all of the blocks of graphs in $Q_k$ do not have any routing (neither partial nor full). But the trouble is that the second assumption is not correct. That is, the blocks are not completely independent as we assumed, and they appear in groups of size $k$ in one graph (a copy of $H_k$ in $Q_k$). For this reason, cycles in $G_{r,k,g}$ may have complicated structures and go through several copies $D^i_r$'s in $P_r$. For instance, a cycle $C$ may start (as a path) at some vertex $s_\alpha$ in a copy $D^i_r$ in $P_r$ ($D^i_r$ corresponds to vertex $v_i \in \mathcal{H}$) and then at some vertex $h^1_{ab} \in D^i_r$ the path enters the start point of a block $b_x$ in a graph $R^j_k$ (which is a copy of $H_k$ in $Q_k$). But instead of going out from an end point of the same block $b_x$ (of $R^j_k$) it goes (within $R^j_k$) to an end point of another block $b_y$ of $R^j_k$. We call this situation a *jump between blocks* of $R^j_k$. This way, the path may end-up in another copy $D^{i'}_r$ (before going back to $s_\alpha$). Looking from a higher level at the underlying hypergraph structure (which has a structure like $\mathcal{H}$), we can think of this path as going from vertex $v_i$ (graph $D^i_r \in P_r$) to $v_{i'}$ (graph $D^{i'}_r \in P_r$) in $\mathcal{H}$ through hyperedge $e_j$ (through graph $R^j_k \in Q_k$ by starting at the start point of one block and going down to an end point of another block of $R^j_k$). But if this happens, since the start point ($s_\alpha$) is in $D^i_r$, this path must eventually come back to $t_\alpha$ in $D^i_r$ (because $t_\alpha \in D^i_r$ is the only vertex that has an edge to $s_\alpha$). However, because $\mathcal{H}$ has girth $g$, the path has to go through at least $g$ other graphs in $P_r$

before getting back into $D_r^i$. Therefore, the cycle contains at least $\Omega(g)$ edges from the graphs in $P_r$. We call these cycles (that go through several graphs in $P_r$) *long* cycles (because $g$ is going to be large) and those that are within one graph of $P_r$ (and so do not jump between blocks of graphs in $Q_k$), *short* cycles. This implies that the total number of long cycles can be at most a fraction $\frac{1}{g}$ of the total number of edges in the graphs in $P_r$. If $g$ is large and $r$ is small the total number of short and long cycles will be small.

**Lemma 3.** *$\nu_e^*(G_{r,k,g}) \in \Omega(rp)$, that is, $G_{r,k,g}$ has a fractional cycle packing solution of value $\Omega(rp)$.*

*Proof.* Recall that by Fact 1, there is a half-integral solution (for EDP problem) in any instance $D_r$, which contains one half-integral path for each pair $s_i, t_i$. If we add edges $t_i s_i$ (for $1 \le i \le r$) to $D_r$ then there are at least $r$ half-integral cycles in $D_r$. In this fractional solution, we route exactly two (half-integral) cycles through each intersection module.

We do have the feedback edges in $G_{r,k,g}$ (in every graph $D_r^j \in P_r$). Also, by Fact 2, for every graph $R_k^i \in Q_k$ there is a half-integral fractional solution in which all the blocks in $R_k^i$ are fully routed (with value $\frac{1}{2}$). Therefore, all the blocks in all graphs in $Q_k$ (which have replaced all the intersection modules in graphs in $P_r$) are fully routed (with value $\frac{1}{2}$). These two imply that each (modified) graph $D_r^j \in P_r$ has $r$ half-integral (short) cycles (where parts of the fractional cycles go through blocks of the graphs in $Q_k$). Since there are $p$ graphs in $P_r$ we get $\Omega(rp)$ half-integral cycles. □

**Lemma 4.** $\nu_e(G_{r,k,g}) \in O(\frac{q}{r} + \frac{r^2 p}{g})$.

**Proof sketch:** By Fact 2, for every graph $R_k^i \in Q_k$, there is at most two blocks that can be (partially or fully) routed. So over all graphs in $Q_k$, there are at most $2q$ blocks that can be (partially or fully) routed. Since blocks have replaced the intersection modules of the graphs in $P_r$ and every short cycle in a graph in $P_r$ goes through at least $r$ blocks, plus the fact that at most two cycles can go through any routed block, there can be at most $4q/r$ directed short cycles in the graphs of $P_r$ in $G_{r,k,g}$.

Now we upper bound the number of long cycles. Because every graph in $P_r$ has constant degree and $O(r^2)$ vertices, the total number of edges of $G_{r,k,g}$ that are parts of the graphs in $P_r$ is $O(r^2 p)$. Therefore, by the arguments before Lemma 3, there are at most $O(\frac{r^2 p}{g})$ long cycles in $G_{r,k,g}$. Thus the total number of short and long cycles is $O(\frac{q}{r} + \frac{r^2 p}{g})$. ■

Recall that for hypergraph $\mathcal{H}$, the number of vertices $p$ and hyperedges $q$ are in $O(\frac{k^{2g}}{r})$ and $O(k^{2g-1})$, respectively. Let $r$ be some (not too small) constant and $k = g$. This implies that $p \in O(k^{2k})$ and $q \in O(k^{2k-1})$. The total number of vertices $n$ in graph $G_{r,k,g}$ is $O(r^2 p + k^2 q)$ which is $O(k^{2k+1})$. By Lemmas 3 and 4, the integrality gap is at least $\Omega((rp)/(\frac{q}{r} + \frac{r^2 p}{g}))$ which is $\Omega(k)$. This is $\Omega(\frac{\log n}{\log \log n})$, which completes the proof of Theorem 4 □ ■

Combining Theorems 3 and 4 and noting that the constructions in Theorem 3 have size polynomial, we obtain:

**Corollary 1.** *Directed VDC has an integrality gap of* $\Omega(\frac{\log n}{\log\log n})$.

The construction for the hardness result has similar structure and uses the hardness of directed EDP by Ma and Wang [19] which is based on PCP theorem [1, 2] together with Raz [20] parallel repetition theorem.

**Theorem 5.** *[19] For any* $\epsilon > 0$, *directed EDP cannot be approximated within ratio* $2^{-\log^{1-\epsilon} n}$ *unless* $NP \subseteq DTIME(2^{polylog(n)})$.

A careful analysis of proof of Theorem 5 reveals that in fact their proof implies the following stronger version:

**Theorem 6.** *Given an instance* $I$ *of directed EDP, which consists of an acyclic digraph* $G$ *(on* $n$ *vertices) and* $k$ *source-sink pairs* $(s_1, t_1), \ldots, (s_k, t_k)$ *in* $G$, *where* $k \in \Omega(n^\delta)$ *for some absolute* $\delta > 0$, *it is quasi-NP-hard to decide between the following two cases:*

1. *(Yes-instance) All pairs* $(s_i, t_i)$ *can be routed by disjoint paths, or*
2. *(No-instance) At most a fraction* $2^{-\log^{1-\epsilon} n}$ *of the pairs can be routed.*

**Theorem 7.** *For any* $\epsilon > 0$, *there is no* $O(\log^{1-\epsilon} n)$*-approximation for EDC unless* $NP \subseteq DTIME(2^{polylog(n)})$.

*Proof.* Let $I_{EDP}$ be an instance of (directed) EDP as in Theorem 6 which consists of a directed acyclic graph $G$ and $k$ pairs $(s_i, t_i)$, $1 \le i \le k$. Take two copies of $I_{EDP}$, named $I_P^1$ and $I_P^2$ and identify the source $s_i^1 \in I_P^1$ with $s_i^2 \in I_P^2$ and call this new vertex $s_i$ $(1 \le i \le k)$. Denote this new graph by $H_k$, with $2k$ source-sink pairs $s_i, t_i^1$ and $s_i, t_i^2$, $1 \le i \le k$. As in the construction of the integrality gap, we name the triple $s_i, t_i^1, t_i^2$ *block* $i$ of $H_k$ with start point $s_i$ and end points $t_i^1$ and $t_i^2$. In a solution to EDP problem with instance $H_k$ and the $2k$ pairs $s_i, t_i^1$ and $s_i, t_i^2$ $(1 \le i \le k)$ we say block $i$ is fully routed if there are two paths, one from $s_i$ to $t_i^1$ and one from $s_i$ to $t_i^2$, in the solution. If only one of these paths exists then we say block $i$ is partially routed. If none of them exists block $i$ is not routed at all. Again, the fact that $H_k$ is acyclic will be crucial in the analysis of our construction. Let $r$ and $g$ be some positive integers (to be specified later) and take a $\binom{r}{2}$-regular $k$-uniform hypergraph $\mathcal{H}$ with girth $g$. As before, let $p$ and $q$ be the number of vertices and hyperedges of $\mathcal{H}$, respectively. We construct graph $G_{r,k,g}$ whose underlying structure is hypergraph $\mathcal{H}$ in the same manner we did in Theorem 4 except that now we use copies of $H_k$ (defined above) to place in $Q_k$. The rest of the construction remains the same. That is, we take $p$ copies of $D_r$ and put them in the set $P_r$ and then replace the intersection modules of them with blocks of copies of $H_k$ in $Q_k$ in the same manner. Let $P_r = \{D_r^1, \ldots, D_r^p\}$ and $Q_k = \{R_k^1, \ldots, R_k^q\}$. We define *short* and *long* cycles in $G_{r,k,g}$ in the same way as we did in Theorem 4.

If $I_{EDP}$ is a *Yes* instance then all the $k$ blocks in $H_k$ can be fully routed. This means that every block of every $R_k^i \in Q_k$ can be fully routed. So for every graph $D_r^j \in P_r$ there are $r$ disjoint paths from $s_i$ to $t_i$, one for each $1 \leq i \leq r$, and because of the existence of feedback edges (connecting $t_i$ to $s_i$) we have $r$ edge-disjoint cycles in each $D_r^j \in P_r$. This gives a total of $\Omega(rp)$ edge-disjoint cycles.

If $I_{EDP}$ is a *No* instance then at most a fraction $2^{-\log^{1-\epsilon} n}$ of the $k$ pairs can be routed. Since $k \in \Omega(|I_{EDP}|^{\delta})$, this fraction is at most $2^{-\left(\frac{\log k}{\delta}\right)^{1-\epsilon}}$, which we name it $\alpha$. So, at most $2\alpha k$ blocks in each graph in $Q_k$, and therefore, at most $2\alpha kq$ blocks over all the graphs in $Q_k$ can be (fully or partially) routed. Because each short cycle in a graph $D_r^i \in P_r$ goes through $r$ blocks (previously intersection modules), the number of short cycles over all graphs in $P_r$ is at most $2\alpha kq/r$. The same argument we had in Theorem 4 for long cycles implies that the number of long cycles here is at most a $\frac{1}{g}$ fraction of the total number of edges in all graphs in $P_r$. This is at most $O(\frac{r^2 p}{g})$. All together, the number of short and long cycles is $O(\frac{2\alpha kq}{r} + \frac{r^2 p}{g})$, which is $O(rp(2\alpha + \frac{r}{g}))$, because $q = \frac{r^2 p}{k}$.

The above arguments, together with Theorem 6 imply that deciding between $\Omega(rp)$ cycles and $O(rp(2\alpha + \frac{r}{g}))$ cycles in $G_{r,k,g}$ is quasi-NP-hard. Equivalently, it is quasi-NP-hard to have an approximation algorithm with factor $\Omega(1/F(r,k,g))$ where $F(r,k,g) = 2\alpha + \frac{r}{g}$. Let $r$ be a (not too small) constant and $g = O(\log^c k)$ for an arbitrary large constant $c > 0$. This implies that the hardness factor (i.e. $1/F(r,k,g)$) is $\Omega(\log^c k)$. With this setting of parameters, hypergraph $\mathcal{H}$ has $p = O(k^{\log^c k})$ vertices and $q = \frac{rp}{k} = O(k^{\log^c k - 1})$ edges. So, if $N$ denotes the number of vertices of $G_{r,k,g}$ (i.e. the size of construction), then it will be at most $O(k^{\log^c k})$ (for the vertices in graphs in $P_r$) plus $O(k^{\frac{2}{\delta} + \log^c k})$ (for the vertices in graphs in $Q_k$). So overall, $N \in O(k^{\frac{2}{\delta} + \log^c k})$, which is quasi-polynomial in the size of input (instance $I_{EDP}$). Rewriting the hardness factor $\Omega(\log^c k)$ in terms of $N$ gives a hardness of $\Omega(\log^{\frac{c}{c+1}} N)$. □

Under a stronger complexity assumption that for some sufficiently small $\delta > 0$, $NP \not\subseteq DTIME(2^{n^{\delta}})$, we can improve the hardness result to $\Omega(\frac{\log n}{\log \log n})$.

## References


1. S. Arora, C. Lund, R. Motwani, M. Sudan, M. Szegedy, *Proof verification and hardness of approximation problems*, J. of the ACM, 45(3):501–555, 1998.
2. S. Arora and S. Safra, *Probabilistic checking of proofs: A new characterization of NP*, J. of the ACM, 45(1):70-122, 198.
3. V. Bafna, P. Berman, and T. Fujito, *Constant ratio approximation of the weighted feedback vertex set problem for undirected graphs*, In Proc. of ISAAC'95, Lecture Notes in Computer Science No. 1004, pages 142-151, Springer-Verlag, 1995.
4. P. Balister, *Packing digraphs with directed closed trials*, Combinatorics, Probability, and Computing, 12(1):1–15, 2003.
5. A. Becker and D. Geiger, *Approximation algorithms for the loop cutset problem*, In Proc. of 10th Conf. on Uncertainty in ARtificial Intelligence, pages 60-68, 1994.

6. A. Carpara, A. Panconesi, and R. Rizzi, *Packing cycles in undirected graphs*, J. Algorithms 48(1): 239–256, 2003.
7. B. Carr and S. Vempala, *Randomized meta-rounding*, In Proc. of STOC 2000.
8. C. Chekuri and S. Khanna, *Edge Disjoint Paths Revisited*, Proc. of SODA 2003.
9. J. Cheriyan and M.R. Salavatipour, *Hardness and Approximation Results for Packing Steiner Trees Problems*, In Proc. of ESA'04.
10. F. A. Chudak, M. X. Goemans, D. S Hochbaum, and D. P. Williamson, *A primal-dual interpretation of two 2-approximation algorithms for the feedback vertex set problem in undirected graphs*, Operations Research Letters, 22, 111–118, 1998.
11. P. Erdös and L. Pósa, *On the maximal number of disjoint circuits of a graph*, Publ. Math. Debrecen 9:3–12, 1962.
12. G. Even, J. Naor, B. Schieber, and L. Zosin, *Approximating minimum subset feedback sets in undirected graphs with applications*, SIAM J. Discrete Math. 13(2):255-267, 2000.
13. G. Even, J. Naor, B. Schieber, and M. Sudan, *Approximating minimum feedback sets and multicuts in directed graphs*, Algorithmica 20(2): 151–174, 1998.
14. 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.
15. K.Jain, M.Mahdian, M.R.Salavatipour, *Packing Steiner trees*, In Proc. SODA 2003.
16. M. Krivelevich, Z. Nutov, R. Yuster, *Approximation algorithms for cycle packing problems*, To appear in Proc. of SODA'05.
17. F. Lazebnik, V.A. Ustimenko and A.J. Woldar, *New upper bounds on the order of cages*, Electronic Journal of Combinatorics, Volume 14, R13: 1–11, 1997.
18. L.C.Lau, *An approximate max-Steiner-tree-packing min-Steiner-cut theorem*, In Proc. 45th IEEE FOCS 2004.
19. B. Ma and L. Wang, *On the inapproximability of disjoint paths and minimum steiner forest with bandwidth constraints*, J. of Computer and Systems Sciences, 60(1):1–12, 2000.
20. R. Raz, *A parallel repetition theorem*, SIAM J. of Computing, 27(3): 763–803, 1998.
21. P. D. Seymour, *Packing directed circuits fractionally*, Combinatorica 15(2):281–288, 1995.