

Approximation Algorithms and Hardness Results for Cycle Packing Problems

Michael Krivelevich

Tel Aviv University

and

Zeev Nutov

Open University of Israel

and

Mohammad R. Salavatipour

University of Alberta

and

Jacques Verstraete

University of Waterloo

and

Raphael Yuster

University of Haifa

A preliminary version of this paper appeared as two separate papers, titled: “*Approximation Algorithms for Cycle Packing Problems*” in Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms, Vancouver, Canada, 2005, and “*Disjoint Cycles: Integrality Gap, Hardness, and Approximation*” in Proceedings of 11th Conference on Integer Programming and Combinatorial Optimization, Berlin, Germany, 2005.

Authors’ addresses: Michael Krivelevich, Department of Mathematics, Tel Aviv University, Tel Aviv, Israel. E-mail: krivelev@post.tau.ac.il. Research supported in part by USA-Israel BSF Grant 2002-133, and by grant 64/01 from the Israel Science Foundation.

Zeev Nutov, Department of Computer Science, The Open University of Israel, Tel Aviv, Israel. E-mail: nutov@openu.ac.il.

Mohammad R. Salavatipour, Department of Computing Science, University of Alberta, Edmonton, AB T6G2E8, Canada. E-mail: mreza@cs.ualberta.ca. Supported by NSERC and a faculty startup grant from University of Alberta.

Jacques Verstraete, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ON N2L3G1, Canada. E-mail: jverstra@math.uwaterloo.ca. Supported by NSERC Discovery Grant 298149.

Raphael Yuster, Department of Mathematics, University of Haifa, Haifa 31905, Israel. E-mail: raphy@research.haifa.ac.il.

Permission to make digital/hard copy of all or part of this material without fee for personal or classroom use provided that the copies are not made or distributed for profit or commercial advantage, the ACM copyright/server notice, the title of the publication, and its date appear, and notice is given that copying is by permission of the ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or a fee.

© 20YY ACM 0000-0000/20YY/0000-0001 \$5.00

The *cycle packing number* $\nu_e(G)$ of a graph G is the maximum number of pairwise edge-disjoint cycles in G . Computing $\nu_e(G)$ is an NP-hard problem. We present approximation algorithms for computing $\nu_e(G)$ in both undirected and directed graphs. In the undirected case we analyze a variant of the modified greedy algorithm suggested by Caprara et al. [J. of Algorithms 48 (2003), 239–256] and show that it has approximation ratio $\Theta(\sqrt{\log n})$ where $n = |V(G)|$. This improves upon the previous $O(\log n)$ upper bound for the approximation ratio of this algorithm. In the directed case we present a \sqrt{n} -approximation algorithm. Finally we give an $O(n^{2/3})$ -approximation algorithm for the problem of finding a maximum number of edge-disjoint cycles that intersect a specified subset S of vertices. We also study generalizations of these problems. Our approximation ratios are the currently best known ones and, in addition, provide upper bounds on the *integrality gap* of standard LP-relaxations of these problems. In addition, we give lower bounds for the integrality gap and approximability of $\nu_e(G)$ in directed graphs. Specifically, we prove a lower bound of $\Omega(\frac{\log n}{\log \log n})$ for the integrality gap of edge-disjoint cycle packing. We also show that it is quasi-NP-hard to approximate $\nu_e(G)$ within a factor of $O(\log^{1-\epsilon} n)$ for any constant $\epsilon > 0$. This improves upon the previously known APX-hardness result for this problem.

Categories and Subject Descriptors: F.2.2 [Nonnumerical Algorithms and Problems]: Computations on discrete structures; G.2.2 [Discrete Mathematics]: Graph Algorithms

General Terms: Approximation Algorithms, Hardness of Approximation

Additional Key Words and Phrases: cycle packing, edge-disjoint, integrality gap

1. INTRODUCTION

We study approximation algorithms, lower and upper bounds for the integrality gaps, and hardness of approximation 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 EDC problem has been studied extensively both in undirected and directed graphs (see, e.g., [Balister 2003; Caprara et al. 2003; Seymour 1995]). Packing disjoint cycles is a fundamental problem in graph theory with applications in several areas (see the discussion in [Balister 2003] for an application in computational biology and reconstruction of evolutionary trees). We denote by $\nu_e(G)$ the size of a largest collection of edge-disjoint cycles in G . It is well known that computing $\nu_e(G)$ is NP-hard even for undirected graphs. Already the very special case of deciding whether a graph (digraph) has a triangle decomposition is known to be NP-Complete (see, e.g. [Dor and Tarsi 1992] for a more general theorem on the NP-Completeness of such decomposition problems). This motivates the study of approximation algorithms for this problem. An algorithm is called an α -approximation for a maximization problem if the solution returned by the algorithm is at least a factor $1/\alpha$ of the optimal solution, and α is called the approximation (or performance) ratio of the algorithm.

A natural generalization of EDC that we consider is the problem of S -cycle packing (denote by S -EDC). In S -EDC we are given a directed graph G and a subset S of its vertices and the goal is to find among the cycles that intersect S (henceforth, S -cycles) a maximum number $\nu_e(G, S)$ of edge-disjoint ones. We note that on directed simple graphs, S -EDC is a special case of the extensively studied *edge-disjoint paths* problem. See [Chekuri and Khanna 2003] for an $O(n^{4/5})$ -approximation algorithm

and [Varadarajan and Venkataraman 2004] for an $O(n^{2/3} \log^{2/3} n)$ -approximation algorithm for the edge-disjoint paths problem in directed graphs.

The approximation algorithms we present also provide upper bounds for the integrality gap of the corresponding problem. Given a graph $G = (V, E)$, a *fractional edge-disjoint cycle packing* in G is a function ψ from the subset \mathcal{C} of all cycles in G to $[0, 1]$ satisfying $\sum_{C: e \in C} \psi(C) \leq 1$ for each $e \in E$. Letting $|\psi| = \sum_{C \in \mathcal{C}} \psi(C)$, the *fractional edge-disjoint cycle packing number* $\nu_e^*(G)$ of G is defined to be the maximum of $|\psi|$ taken over all fractional cycle packings ψ in G . The *edge cycle cover number* $\tau_e(G)$ of G is the minimum number of edges whose deletion makes G acyclic. Clearly, $\nu_e(G) \leq \nu_e^*(G) \leq \tau_e(G)$ for any graph/digraph G .

Previously known results: A recent result of Caprara, Panconesi, and Rizzi [Caprara et al. 2003] shows that by slightly modifying the greedy algorithm one obtains an $O(\log n)$ -approximation algorithm for undirected EDC 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 $P=NP$). Although the EDC and S -EDC problems are closely related to the well-known problem of finding the maximum number of edge-disjoint paths in a graph, none of the results for the edge-disjoint paths problem implies any of the results presented in this paper. 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 packing *vertex-disjoint cycles* (denoted by VDC), is known as Feedback Vertex Set (FVS) problem. This is the problem of finding the 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 [Bafna et al. 1995; Becker and Geiger 1994; Chudak et al. 1998]). The dual of EDC is the problem of finding the minimum number of edges in a graph that meet every cycle (FES). This problem is trivial for undirected graphs (complement of a spanning tree in each component). For directed graphs, there is an easy reduction from FES to FVS. Seymour [Seymour 1995] showed that, if the optimal fractional FVS in a directed graph G has value φ^* 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 [Even et al. 1998]. Alon and Seymour (see [Seymour 1995]) showed that the integrality gap of FVS is $\Omega(\log \varphi^*)$.

Our results: We present approximation algorithms and hardness results for different versions of cycle packing problems. For undirected EDC we present an $O(\sqrt{\log n})$ -approximation algorithm by combining the modified greedy algorithm of [Caprara et al. 2003] with an ordinary greedy algorithm. In particular, we obtain the following result.

THEOREM 1.1. *There exists an $O(\sqrt{\log n})$ -approximation algorithm for the undirected maximum cycle packing problem.*

We also prove that the approximation guarantee of the algorithm is $\Omega(\sqrt{\log n})$. For directed EDC we prove the following result.

THEOREM 1.2. *There exists a \sqrt{n} -approximation algorithm for the problem of directed EDC.*

The algorithms in Theorems 1.1 and 1.2 are easily adjusted to the capacitated version of the problems as well, where every edge has a given capacity and the goal is to find the maximum number of cycles such that the number of cycles containing each edge is at most its capacity. For simplicity of exposition, we prove our results for the uncapacitated case, and then show how they extend to the capacitated case. For the S -EDC problem on directed graphs we prove:

THEOREM 1.3. *There exists an $O(n^{2/3})$ -approximation algorithm for the directed S -EDC on simple digraphs.*

The approximation ratios in Theorems 1.1, 1.2, and 1.3 provide bounds on the integrality gap of the standard LP-relaxations to the problems. Specifically, each of the algorithms computes a packing \mathcal{C} so that: $|\mathcal{C}|/\nu_e^*(G) = \Omega(1/\sqrt{\log n})$ in Theorem 1.1, $|\mathcal{C}|/\nu_e^*(G) \geq 1/\sqrt{n}$ in Theorem 1.2, and $|\mathcal{C}|/\tau_e(G, S) = \Omega(n^{-2/3})$ in Theorem 1.3, where $\tau_e(G, S)$ is the minimum number of edges needed to cover all the S -cycles in G .

We also study the integrality gap and hardness of approximation of these problems. Our main result is that EDC in directed graphs 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 $\epsilon > 0$. Note that the only known hardness result prior to this work was APX-hardness [Caprara et al. 2003] (for undirected EDC).

THEOREM 1.4. *For any $\epsilon > 0$, there is no $O(\log^{1-\epsilon} n)$ -approximation for directed EDC unless $\text{NP} \subseteq \text{DTIME}(2^{\text{poly}(\log n)})$.*

Under the stronger complexity assumption that $\text{NP} \not\subseteq \text{DTIME}(2^{n^\epsilon})$ for some $\epsilon > 0$, we can prove a slightly better hardness of $\Omega\left(\frac{\log n}{(\log \log n)^2}\right)$.

Very recently, [Friggstad and Salavatipour 2006] have proved that the undirected EDC is $\Omega(\log^{\frac{1}{2}-\epsilon})$ -hard to approximate, for any $\epsilon > 0$, unless $\text{NP} \subseteq \text{DTIME}(n^{\text{poly}(\log n)})$. This shows that the ratio of the algorithm in Theorem 1.1 is (almost) tight.

The organization of the paper is as follows. In the next section we present the upper bound results, namely we prove Theorems 1.1, 1.2, and 1.3. Section 3 contains the lower bound results. We prove lower bounds for the integrality gap of EDC and a variation of this problem. Finally, we will prove Theorem 1.4.

2. APPROXIMATION ALGORITHMS

2.1 Cycle Packing in Undirected Graphs

Perhaps the most natural greedy algorithm that finds a large number of disjoint cycles in a given graph G is to find the smallest cycle at each iteration and delete it from the graph. As shown in [Caprara et al. 2003], the approximation ratio of this algorithm is $\Theta(\sqrt{n})$. They suggested the following Modified Greedy algorithm for finding edge-disjoint cycles in G . The algorithm starts with $\mathcal{C} = \emptyset$ and performs the following steps iteratively, until there are no edges left in G :

- (1) While G contains a vertex v of degree ≤ 1 , delete v (and the edge incident to v , if exists).

- (2) While G contains a vertex v of degree 2 with neighbors v' and v'' , delete v and edges vv' , vv'' and replace them by a new edge $v'v''$.
- (3) Find a shortest cycle C in G , add C to the constructed solution \mathcal{C} and remove its edges from G .

It is easy to see that Steps 1 and 2 do not change the value of an optimal solution. Also, it is well-known that for every graph with minimum degree at least 3 the girth is $O(\log n)$ (e.g. see [Bollobás 2004]). Therefore, every cycle found by the algorithm has size $O(\log n)$ and therefore intersects cycles whose total weight is at most $O(\log n)$ in an optimal fractional solution. Hence the solution is within a factor of $O(\log n)$ of the optimal fractional solution. Here we show how a small change in this algorithm, together with a careful analysis yields an $O(\sqrt{\log n})$ -approximation for the EDC problem, hence proving Theorem 1.1.

Remark: It is also easily seen that a slight modification of Step 3 of the Modified Greedy algorithm yields an $O(\log n)$ -approximation for the problem of packing *vertex-disjoint cycles* (VDC). In fact this proves an upper bound on the ratio of the solution value of this algorithm to that of an optimal *fractional* vertex-disjoint cycle packing solution.

Our algorithm is as follows. At Phase 1, while $\text{girth}(G) \leq \sqrt{\log |V(G)|}$ we apply the Modified Greedy Algorithm (where $|V(G)|$ is the number of vertices in the *current* graph); the condition $\text{girth}(G) \leq \sqrt{\log |V(G)|}$ is checked after Step 2. Phase 2 starts when $\text{girth}(G) > \sqrt{\log |V(G)|}$ after Step 2 for the first time; then we repeatedly apply Step 3 only, which is the (ordinary) Greedy Algorithm.

2.1.1 The approximation ratio. We prove the following theorem that clearly implies Theorem 1.1.

THEOREM 2.1. *The algorithm computes a cycle packing of size $\Omega(\nu_e^*(G)/\sqrt{\log n})$.*

In the proof we use the following lemma, that provides an improved analysis of the performance of the greedy algorithm on graphs with large girth.

LEMMA 2.2. *Let H be a graph with n nodes, $m > n$ edges, and girth g , and let \mathcal{C}_H be the set of cycles found in H by the greedy algorithm. Then*

$$|\mathcal{C}_H| \geq \frac{(m-n)^2}{4m(m+n)} \cdot \frac{g}{\log(m-n)} \cdot \nu_e^*(H).$$

In particular, if $m \geq (1+\epsilon)n$ for an $\epsilon > 0$ then

$$|\mathcal{C}_H| \geq \frac{\epsilon^2}{4(1+\epsilon)(2+\epsilon)} \cdot \frac{g}{\log(\epsilon n)} \cdot \nu_e^*(H).$$

PROOF. Consider the steps of the Greedy Algorithm when $|E(H)| \geq (m+n)/2 = n + (m-n)/2$. Bollobás and Thomason [Bollobás and Thomason 1997] proved that if a graph H satisfies $|E(H)| \geq |V(H)| + f$ for a $f > 0$, then $\text{girth}(H) \leq 2(1 + |V(H)|/f) \log(2f)$. Thus during all these steps

$$\text{girth}(H) \leq \frac{2(m+n)}{m-n} \log(m-n).$$

The number of edges deleted during these steps is at least $(m-n)/2$. Also, $\nu_e^*(H) \leq m/g$. Indeed, if ψ is an optimal fractional edge-disjoint cycle packing, then $m = |E| \geq \sum_{e \in E} \sum_{C \ni e} \psi(C) = \sum_C \psi(C) \cdot |C| \geq g \sum_C \psi(C) = g\nu_e^*(G)$. Thus

$$|\mathcal{C}_H| \geq \frac{(m-n)^2}{4(m+n)\log(m-n)} \geq \frac{(m-n)^2}{4(m+n)\log(m-n)} \cdot \frac{g}{m} \nu_e^*(H).$$

The second statement follows from the first by observing that the function $f(m) = \frac{(m-n)^2}{4m(m+n)}$ is increasing for $m \geq n$. \square

Completing the proof of Theorem 2.1: Note that Steps 1 and 2 of the Modified Greedy Algorithm do not change the value of an optimal solution. Let \mathcal{C}_1 and \mathcal{C}_2 be the sets of cycles added to the packing during Phase 1 and Phase 2, respectively. Recall that in Phase 1 we execute the Modified Greedy Algorithm, and the length of every added cycle does not exceed $\sqrt{\log |V(G)|}$. Phase 2 starts when a cycle added to an approximate packing has length more than $\sqrt{\log |V(G)|}$, and executes the Greedy Algorithm. Fix an optimal fractional packing ψ^* , so $|\psi^*| = \nu_e^*(G)$. Let ψ_1^* be the restriction of ψ^* to the cycles that intersect some cycle from \mathcal{C}_1 , $\psi_2^* = \psi^* - \psi_1^*$. Since every cycle from \mathcal{C}_1 has length $\leq \sqrt{\log n}$, we have:

$$|\psi_1^*| \leq |\mathcal{C}_1| \sqrt{\log n}.$$

We claim that:

$$|\psi_2^*| \leq 60|\mathcal{C}_2| \sqrt{\log n}.$$

Thus

$$|\psi^*| = |\psi_1^*| + |\psi_2^*| \leq \sqrt{\log n}(|\mathcal{C}_1| + 60|\mathcal{C}_2|) \leq 60\sqrt{\log n}|\mathcal{C}|.$$

We prove that $|\psi_2^*| \leq 60|\mathcal{C}_2| \sqrt{\log n}$ using Lemma 2.2. Let H be the graph the second phase starts with. Then $\text{girth}(H) \geq \sqrt{\log |V(H)|}$, and H has at least $3|V(H)|/2$ edges (since H has minimum degree at least 3). Thus, by substituting $\epsilon = 1/2$ in the bound in Lemma 2.2 we get:

$$\begin{aligned} |\mathcal{C}_2| &\geq \frac{1}{60} \cdot \frac{\text{girth}(H)}{\log(|V(H)|/2)} \cdot \nu_e^*(H) \geq \frac{1}{60} \cdot \frac{\sqrt{\log |V(H)|}}{\log(|V(H)|/2)} \cdot \nu_e^*(H) \geq \\ &\frac{1}{60} \cdot \frac{\sqrt{\log |V(H)|}}{\log |V(H)|} \cdot \nu_e^*(H) = \frac{\nu_e^*(H)}{60\sqrt{\log |V(H)|}}. \end{aligned}$$

Clearly, $|V(H)| \leq n$. Also, $|\psi_2^*| \leq \nu_e^*(H)$, since ψ_2^* corresponds to a fractional packing in H . This implies $|\psi_2^*| \leq \nu_e^*(H) \leq 60|\mathcal{C}_2| \sqrt{\log |V(H)|} \leq 60|\mathcal{C}_2| \sqrt{\log n}$, as claimed.

Remark: Our algorithm is easily adjusted to the capacitated version of the problem, where we are also given integral capacities $\{c_e : e \in E\}$ on the edges; the goal is to find a maximum weight family \mathcal{C} of cycles so that for every edge $e \in E$ the capacity constraints $\sum\{w(C) : e \in C, C \in \mathcal{C}\} \leq c(e)$ are satisfied. We can imitate the capacitated case by the uncapacitated one if we replace every edge e by c_e parallel edges with the same ends as e , and “forbid” the arising cycles of the length 2. However, this will give only a pseudo-polynomial time algorithm. To get

a polynomial algorithm, let us show how to adjust Steps 1, 2, and 3 in the modified greedy algorithm to handle this case. Each time an edge of capacity zero arises, it is deleted. So assume that G has no zero capacity edges. Step 1 remains the same. In Step 2, the new edge $v'v''$ gets capacity $\min\{c_{vv'}, c_{vv''}\}$. In Step 3, after a shortest cycle C is found, we also find the minimum capacity edge e in C , add C to the constructed packing, and assign it weight $w(C) = c_e$. Then, in G , we reduce by c_e the capacities of the edges of C , and remove the arising zero capacity edges. Each one of the steps can be performed in polynomial time, and leads to a graph with less edges. Thus the running time is polynomial. It is easy to see that our analysis of the approximation ratio is valid for the capacitated case as well. The algorithm in Section 2.2.1 for directed graphs (to follow) admits a similar adjustment.

2.1.2 A tight example

THEOREM 2.3. *The approximation ratio of the Modified Greedy Algorithm is $\Omega(\sqrt{\log n})$.*

The proof borrows some ideas from a lower bound argument in [Caprara et al. 2003]. For the proof we will need the following technical lemma.

LEMMA 2.4. *Let G be a graph on n vertices of maximum degree at most 7. Let $V_0 \subseteq V(G)$. If $|V_0| \geq n/2$ then there exists a subset $U \subset V_0$ of size $|U| = \lceil \log n \rceil$, such that all vertices of U are at distance more than $\frac{1}{3} \log n$ from each other.*

PROOF. Note that every vertex $v \in G$ is at distance at most k from at most $7 \cdot 6^{k-1} < 7^k$ vertices from G . Define an auxiliary edge set E_0 on V_0 so that $(u, v) \in E_0$ if $\text{dist}_G(u, v) \leq \frac{1}{3} \log n$. Let $H = (V_0, E_0)$. Then H is a graph on at least $n/2$ vertices of maximum degree $\Delta(H) < 7^{\frac{1}{3} \log n} < n^{0.95}$, and has therefore an independent set U of size at least $|V(H)|/(1 + \Delta(H)) > \log n$. Each such independent set gives a required set of vertices in G .

A k -sunflower S^k is a cycle of length k (the *core cycle*) to each edge of which we attach a cycle of length $k + 1$ (a *petal*), so that the petals are vertex-disjoint outside the core cycle. The number of vertices of S^k is k^2 . Observe that the core is the shortest cycle in a k -sunflower, and removing its edges results in a cycle on k^2 vertices. We choose $k = \sqrt{(\log n)/3}$ and denote $t = k^2$ (we ignore floors and ceilings as they do not affect the asymptotic nature of our result).

Let now G_0 be a 3-regular graph on n vertices of girth more than $t = \frac{1}{3} \log n$. Such graphs exist for infinitely many values of n as proved by Erdős and Sachs [Erdős and Sachs 1963]. We start with $G = G_0$, set $W = \emptyset$, $i = 1$, and repeat $n/(2t)$ times the following procedure:

- (1) Find a subset $U_i \subset V \setminus W$ such that $|U_i| = t$ and all vertices of U_i are at distance more than $\frac{1}{3} \log n$ from each other in G ;
- (2) Insert a copy S_i of the k -sunflower in U_i , placing it arbitrarily within U_i ;
update $W \leftarrow W \cup U_i$; $i \leftarrow i + 1$.

Since the sets U_i are disjoint and the maximum degree of S^k is 4, the graph G has maximum degree at most 7 during the execution of the above procedure. Also, $|W| \leq \frac{n}{2t} \cdot t = \frac{n}{2}$, and therefore finding a required U_i at each step is possible due to Lemma 2.4. Let us denote by G^* the final graph of the above procedure.

CLAIM 2.1. *Let C be a cycle of length at most $\frac{1}{3} \log n$ in G^* . Then C is a cycle in one of the inserted k -sunflowers S_i .*

Proof: Since $\text{girth}(G_0) > \frac{1}{3} \log n$, C contains an edge $e \in E(G^*) - E(G_0)$. Let $i^* = \max\{i : E(C) \cap E(S_i) \neq \emptyset\}$. We claim that C is a cycle in S_{i^*} . Let G_{i^*} be the graph created during the above described procedure after having inserted the sunflower S_{i^*} . Obviously, $C \subset G_{i^*}$. If $E(C) \subset E(S_{i^*})$ we are done. Assume otherwise. Since U_{i^*} spans only the edges of S_{i^*} in G_{i^*} , at some point C leaves U_{i^*} and then returns back. Let $u_1, u_2 \in U_{i^*}$ be the vertices of U_{i^*} where C leaves and reenters U_{i^*} . By our choice of U_{i^*} , $\text{dist}_{G_{i^*}}(u_1, u_2) > \frac{1}{3} \log n$, implying $|C| > \frac{1}{3} \log n$, a contradiction. \square

Completing the proof of Theorem 2.3: We analyze the performance of our algorithm on G^* . By Claim 2.1, the shortest cycles in G^* are the $n/(2t) = O(n/\log n)$ core cycles of the inserted sunflowers, which are vertex-disjoint. Hence the algorithm starts by picking all of them. After all core cycles have been removed, none of the sunflowers contain a cycle of length at most $\frac{1}{3} \log n$, and applying Claim 2.1 again we infer that the modified greedy algorithm will be able to add at most $3|E(G^*)|/\log n = O(n/\log n)$ cycles, altogether ending up with $O(n/\log n)$ cycles. On the other hand, a feasible solution can be obtained by taking all petals of all inserted sunflowers, whose total number is $(n/(2t)) \cdot k = \Theta(n/\sqrt{\log n})$. It follows that the approximation ratio of our algorithm on G^* is

$$\Omega\left(\frac{\frac{n}{\sqrt{\log n}}}{\frac{n}{\log n}}\right) = \Omega(\sqrt{\log n}).$$

2.2 Cycle Packing in Directed Graphs

In this section we present the algorithms for EDC and S -EDC in directed setting.

2.2.1 Proof of Theorem 1.2. This algorithm uses ideas similar to those in [Chekuri and Khanna 2003] for edge-disjoint paths in directed acyclic graphs. It will be convenient to describe the algorithm with a certain parameter ℓ , which will be eventually set to $\ell = \sqrt{n}$. The algorithm starts with $\mathcal{C}_1, \mathcal{C}_2 = \emptyset$ and in the end outputs $\mathcal{C}_1 \cup \mathcal{C}_2$.

Phase 1:

As long as there is a directed cycle of length $\leq \ell$, find such a cycle, add it to \mathcal{C}_1 , and delete its edges from the graph.

Phase 2:

For each $v \in V$, compute a maximum size set $\mathcal{C}_2(v)$ of edge-disjoint directed cycles that contain v . Among the packings computed, let \mathcal{C}_2 be one of maximal size.

THEOREM 2.5. *For $\ell = \sqrt{n}$ the algorithm computes a packing $\mathcal{C}_1 \cup \mathcal{C}_2$ of size at least $\nu_e^*(G)/\sqrt{n}$.*

PROOF. As in the proof of Theorem 2.1, let us fix an optimal fractional packing ψ^* , let ψ_1^* be the restriction of ψ^* to cycles that intersect some cycle from \mathcal{C}_1 , $\psi_2^* = \psi^* - \psi_1^*$. Since every cycle from \mathcal{C}_1 has length $\leq \ell$ we have $|\psi_1^*| \leq \ell|\mathcal{C}_1|$. We claim that $|\mathcal{C}_2| \geq \ell|\psi_2^*|/n$. Thus by combining the bounds for $|\mathcal{C}_1|, |\mathcal{C}_2|$ and

substituting $\ell = \sqrt{n}$ we get:

$$\begin{aligned} |\mathcal{C}_1| + |\mathcal{C}_2| &\geq |\psi_1^*|/\ell + \ell|\psi_2^*|/n \\ &= (|\psi_1^*| + |\psi_2^*|)/\sqrt{n} = |\psi^*|/\sqrt{n}. \end{aligned}$$

To see that $|\mathcal{C}_2| \geq \ell|\psi_2^*|/n$, let G_2 be the graph at the beginning of Phase 2. For each $v \in V$ let $\psi_2^*(v)$ be the restriction of ψ_2^* to the cycles in G_2 containing v . Note that for every $v \in V$ we can compute $\mathcal{C}_2(v)$ using any max-flow algorithm and flow decomposition. By the integrality of an optimal flow from the Max-Flow Min-Cut Theorem, $|\mathcal{C}_2| \geq |\psi_2^*(v)|$ for every vertex v . Thus, since every cycle in G_2 has length $> \ell$, we have:

$$n|\mathcal{C}_2| \geq \sum_{v \in V} |\psi_2^*(v)| \geq \ell|\psi_2^*|.$$

□

2.2.2 Proof of Theorem 1.3. In this section we consider simple digraphs only. The greedy algorithm for the maximum S -cycle packing problem repeatedly chooses a shortest S -cycle and removes its edges from the graph. Similar ideas have been used earlier in [Hajiaghayi and Leighton 2006; Varadarajan and Venkataraman 2004].

THEOREM 2.6. *Given a subset S of vertices of a simple digraph G , the greedy algorithm finds a set of at least $\tau_e(G, S)/(5n^{2/3})$ edge-disjoint directed S -cycles in G .*

Proof: Let $f(n, \ell)$ be the maximum of $\tau_e(G)$ taken over all simple digraphs G on n vertices with $\text{girth}(G) > \ell$. It is easy to see that if \mathcal{C} is a cycle packing computed by the greedy algorithm on G , then $\tau_e(G) \leq \ell|\mathcal{C}| + f(n, \ell)$ for any positive integer ℓ . A similar statement holds for the analogous definition of $f(n, \ell)$ in the undirected case. In fact, a similar statement holds for the analogous vertex-disjoint (directed or undirected) cycle packing and cycle cover problems. In the undirected vertex-disjoint case Komlós [Komlós 1997] showed that $f(n, \ell) = \Theta(\frac{n}{\ell} \ln(n/\ell))$. In the directed vertex-disjoint case, Seymour [Seymour 1995] showed that $f(n, \ell) \leq 4\frac{n}{\ell} \ln(4n/\ell) \ln \log(4n/\ell)$. He also gave an example showing that $f(n, \ell) = \Omega(\frac{n}{\ell} \ln(n/\ell))$. In the edge-disjoint case, answering an earlier conjecture of Bollobás, Erdős, Simonovits, and Szemerédi [Bollobás et al. 1978], Komlós [Komlós 1997] established the asymptotically tight bound $f(n, \ell) = \Theta(\frac{n^2}{\ell^2})$ in undirected graphs.

We generalize this by defining $h(n, \ell)$ to be the maximum of $\tau_e(G, S)$ taken over all simple digraphs G on n vertices and $S \subseteq V(G)$ so that every S -cycle in G has length $> \ell$. Let $\tilde{\nu}(G, S)$ denote the size of an S -cycle packing computed by some run of the greedy algorithm.

LEMMA 2.7. *For any positive integer ℓ ,*

$$\tau_e(G, S) \leq \ell\tilde{\nu}(G, S) + h(n, \ell) \leq (\ell + h(n, \ell))\tilde{\nu}(G, S).$$

PROOF. Fix an optimal cover F with $|F| = \tau_e(G, S)$, and partition it into two sets F_1 and F_2 , where F_1 are the edges contained in S -cycles of length $\leq \ell$ of the S -packing computed. Then $|F_1| \leq \ell\tilde{\nu}(G, S)$, since every S -cycle of length $\leq \ell$

in the packing computed contains at least one edge from F_1 . On the other hand $|F_2| \leq h(n, \ell)$, by the optimality of $|F|$ and by the definition of $h(n, \ell)$. The result follows. \square

For digraphs, the bound $h(n, \ell) = O((n^2/\ell^2) \log^2(n/\ell))$ can be deduced from [Varadarajan and Venkataraman 2004, Theorem 1.1] where a more general problem was considered. We will show that $h(n, \ell) = \Theta(n^2/\ell^2)$ using the following lemma of Komlós [Komlós 1997].

LEMMA 2.8 [KOMLÓS 1997], LEMMA 3. *Let a_0, a_1, \dots, a_t be a sequence of non-negative real numbers, and denote $s_k = \sum_{i=0}^k a_i$. Then there exists $k \in \{0, \dots, t-1\}$ such that $a_k a_{k+1} < \frac{2e}{t^2} s_k s_t$.*

COROLLARY 2.9. *Let a_0, a_1, \dots, a_t be a sequence of integers, and denote $s_k = \sum_{i=1}^k a_i$ and $p = \lceil t/2 \rceil$. Suppose that $s_p \leq s_t/2$. Then there exists $k \in \{0, \dots, p-1\}$ such that:*

$$a_k a_{k+1} < \frac{2e}{p^2} s_k s_p \leq \frac{4e}{t^2} s_k s_t.$$

LEMMA 2.10. *Let S be a subset of vertices of a simple digraph G on n vertices so that every S -cycle in G has length $> \ell$. Then there exists an S -cycle edge-cover F with $|F| \leq 4e(n/\ell)^2$. Moreover, such F can be found in polynomial time.*

PROOF. The proof is by induction on n . If G has no S -cycles, in particular if it has ℓ vertices or less, the statement is obvious. We can also assume that G is strongly connected; otherwise, validity of the result for every strongly connected component of G implies the result for G .

Since every S -cycle in G has length $> \ell$, there are vertices u, v with $u \in S$ and $v \in V(G)$ such that every (u, v) -dipath has length $\geq \ell$, and hence there is a partition of $V(G)$ into nonempty sets X_0, \dots, X_t , where $t \geq \ell$, such that no edge of G has tail in X_i and head in X_j , for $j \geq i+2$. Let $a_i = |X_i|$ for $i = 0, \dots, t$, and let s_k and p be as in Corollary 2.9. Notice that $s_t = n$. We may assume that $s_p \leq n - s_p$, since otherwise we may consider the reversed sequence of a_0, \dots, a_t . By Corollary 2.9, there exists $k \in \{0, \dots, p-1\}$ such that:

$$a_k a_{k+1} < \frac{4e}{t^2} s_k n.$$

Let F' be the edge cut consisting of the set of edges going from X_k to X_{k+1} (if we consider the reversed sequence, then we take also the “reversed” cut). Then, since G is simple,

$$|F'| \leq a_k a_{k+1} < \frac{4e}{t^2} s_k n.$$

We delete F' and apply the inductive hypothesis to the subgraphs G_1 and G_2 of G induced by the corresponding parts $V_1 = X_1 \cup \dots \cup X_k$ and $V_2 = X_{k+1} \cup \dots \cup X_t$. Clearly, any S -cycle in $G - F'$ is entirely contained either in G_1 or in G_2 .

To summarize, we can find a cut F' that divides G into two subgraphs G_1 and G_2 , where G_i has n_i vertices, such that $n_1 + n_2 = n$ and $n_1 \leq n/2 \leq n_2$, and such that $|F'| \leq \frac{4e}{t^2} n_1 n$. We need to prove that:

$$|F'| + 4e \left(\frac{n_1^2}{\ell^2} + \frac{n_2^2}{\ell^2} \right) \leq 4e \frac{n^2}{\ell^2}.$$

Indeed,

$$\begin{aligned} |F'| + 4e \left(\frac{n_1^2}{\ell^2} + \frac{n_2^2}{\ell^2} \right) &\leq \frac{4e}{\ell^2} (n_1 n + n_1^2 + n_2^2) \\ &< \frac{4e}{\ell^2} (2n_1 n_2 + n_1^2 + n_2^2) = 4e \frac{n^2}{\ell^2}. \end{aligned}$$

□

The bound in Lemma 2.10 is tight up to a constant factor even for $S = V$, as can be seen in the following example. Start with a directed cycle on \sqrt{n} vertices and then replace each vertex u with $\ell = \lceil \sqrt{n} \rceil$ copies u_1, \dots, u_ℓ . Also replace each original edge uv with ℓ^2 edges $u_i v_j$, $1 \leq i, j \leq \ell$. Clearly in this graph every cycle has length at least ℓ and every cut must contain at least $\ell^2 = \Omega(n)$ edges.

By Lemmas 2.7 and 2.10 we deduce:

COROLLARY 2.11. *Let S be a subset of vertices of a simple digraph G on n vertices. Then for any integer ℓ ,*

$$\tau_e(G, S) \leq (\ell + 4e(n/\ell)^2) \tilde{\nu}(G, S).$$

In particular for $\ell = 2e^{1/3}n^{2/3}$ we have $\tau_e(G, S) \leq 3e^{1/3}n^{2/3}\tilde{\nu}(G, S) < 5n^{2/3}\tilde{\nu}(G, S)$ and this also completes the proofs of Theorems 2.6 and 1.3.

Remark: In [Chekuri and Khanna 2003] it was shown that the greedy algorithm for the *undirected* Edge-Disjoint Paths problem has approximation ratio $O(n^{2/3})$. The method presented in this section can be used to provide a different proof for the same result.

3. INTEGRALITY GAP AND HARDNESS OF DIRECTED DISJOINT CYCLE PACKING

In this section, we study lower bounds for the integrality gap and hardness of approximation for packing disjoint cycles. For directed graphs, an easy reduction shows that the problems of packing edge-disjoint cycles (EDC) and vertex-disjoint cycles (VDC) are equally hard. Hence, our lower bounds for integrality gap and hardness of approximation carry over to the VDC problem as well. We present this reduction here (Lemma 3.1) for the sake of completeness.

Recall that Theorem 1.2 proves a \sqrt{n} upper bound for the integrality gap of directed EDC. Although we are unable to prove that $\Omega(\sqrt{n})$ is also a lower bound for the integrality, we conjecture this is the case. This conjecture is supported by a construction (Proposition 3.2) showing that $\Theta(\sqrt{n})$ is a lower bound for the integrality gap of the *odd directed cycle packing* problem (namely, the maximum number of edge-disjoint directed cycles of odd length). Then we prove that the (standard) directed EDC problem has an integrality gap of $\Omega(\frac{\log n}{\log \log n})$. This is proved in Theorem 3.3 below. Then we show how to use the ideas of this proof to prove Theorem 1.4, i.e. that $\nu_e(G)$ is quasi-NP-hard to approximate within a factor of $O(\log^{1-\epsilon} n)$, for any $\epsilon > 0$. The idea of starting with a grid-like graph in the proofs of Proposition 3.2 and Theorem 3.3 was inspired by the work of Guruswami et al. [Guruswami et al. 2003] to prove hardness of directed edge-disjoint path problem.

LEMMA 3.1. *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 $|G'| = \text{poly}(|V(G)|)$, 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. Reduction from VDC to EDC: 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' . It is easy to see that G has a collection of integral (or fractional) vertex-disjoint cycles in G with size k if and only if G' has a collection of k integral (or fractional) edge-disjoint cycles.

Reduction from EDC to VDC: 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 incoming edges y_1x, y_2x, \dots, y_px and outgoing edges xz_1, xz_2, \dots, xz_q add the following edges to G' : $v_{y_i x}v_{xz_j}$ 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 G' has k vertex-disjoint integral (or fractional) cycles. \square

PROPOSITION 3.2. *For infinitely many n , there exists a digraph G on n vertices, in which every pair of odd cycles has a common edge, and yet $\nu_{\text{odd}}^*(G) = \Omega(\sqrt{n})$, where $\nu_{\text{odd}}^*(G)$ is the fractional odd cycle packing number of G .*

PROOF. Let N be an odd positive integer, and consider the digraph D_N whose vertices are (i, j) for $i, j = 1, \dots, N$. We think of the first coordinates as rows and the second ones as columns. The edges of D_N emanate from (i, j) to $(i + 1, j)$ for $i = 1, \dots, N - 1$ and $j = 1, \dots, N$ and from (i, j) to $(i, j + 1)$ for $i = 1, \dots, N$ and $j = 1, \dots, N - 1$. There are also edges from (i, N) to $(N + 1 - i, 1)$.

We first show that D_N does not have two *vertex-disjoint* odd directed cycles. Clearly, every cycle of D_N is composed of segments, where each segment starts in the first column, passes through every column sequentially (and sometimes goes down in the rows) until it reaches the last column. Segments are separated by the edges connecting the vertices in the last column to the vertices in the first column. Thus, each segment has a unique start vertex from the first column, and a unique end vertex from the last column. The length of a segment is the number of vertices it contains. Thus, the length of a cycle is the sum of the lengths of its segments. We partition the vertices of D_N into two types, even and odd. Even vertices are those whose coordinates have the same parity. An even (odd) segment of a cycle is a segment of even (odd) length. Notice that since N is odd, the endpoints of even segments belong to different types, while the endpoints of odd segments belong to the same type. Also notice that the end vertex of a segment has the same type as the start vertex of the following segment. It follows that odd cycles must have an even number of even segments and, trivially, an odd number of odd segments. Thus, odd cycles have an odd number of segments. Notice that every cycle (whether even or odd) that does not contain a vertex from the middle row must have an even number of segments (as the segments alternate below and above the middle row). Thus, we have shown that every odd cycle must contain a vertex from the middle row. In particular, every odd cycle has a segment starting in $(i, 1)$ and ending in (j, N) where $i \leq (N + 1)/2$ and $j \geq (N + 1)/2$. Now, let C and C' be two odd cycles. We may assume that C has a segment S starting in $(i, 1)$ and ending in

(j, N) where $i \leq (N + 1)/2$ and $j \geq (N + 1)/2$, and C' has a segment S' starting in $(k, 1)$ and ending in (ℓ, N) where $k \leq (N + 1)/2$ and $\ell \geq (N + 1)/2$. Assume, w.l.o.g., that $i \leq k$. If $i = k$ or $\ell \leq j$ we are done since in this case segments S and S' intersect. Thus, we may assume $i < k$ and $j < \ell$. But in this case we have, as before, that if C' does not contain any vertex of S then the segments of C' must alternate below and above the segment S , and hence C' must have an even number of segments, contradicting the fact that C' is an odd cycle.

To estimate from below the fractional odd cycle packing number of G , for each $1 \leq i \leq (N + 1)/2$, define the cycle C_i as follows:

$$\begin{aligned} C_i = & ((i, 1), (i, 2), \dots, (i, i), (i + 1, i), (i + 2, i), \dots, \\ & (N + 1 - i, i), (N + 1 - i, i + 1), \\ & (N + 1 - i, i + 2), \dots, (N + 1 - i, N), (i, 1)) \end{aligned}$$

(i.e. C_i starts at $(i, 1)$, goes horizontally till (i, i) , then drops vertically to $(N + 1 - i, i)$ and then again goes horizontally till $(N + 1 - i, N)$ and finally returns to $(i, 1)$). It is easy to see that each vertex of D_N belongs to at most two cycles C_i , and therefore, giving value $\psi(C_i) = 0.5$ to each cycle C_i , we obtain a fractional odd cycle packing of value $(N + 1)/4$. Now, by replacing each vertex v of D_N with the path v_{in}, v_{mid}, v_{out} and replacing each edge (u, v) with the edge (u_{out}, v_{in}) we obtain a new graph D'_N with $3N^2$ vertices. Any set of edge-disjoint directed cycles in D'_N is also vertex-disjoint, and corresponds to a set of vertex-disjoint directed cycles in D_N . Furthermore, any odd (even) cycle in D_N corresponds to an odd (even) cycle in D'_N . Thus, by letting $n = 3N^2$ the desired construction follows. \square

THEOREM 3.3. *The directed EDC problem has an integrality gap of $\Omega(\frac{\log n}{\log \log n})$.*

We give a 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 [Guruswami et al. 2003]. 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 $u_{ii}h_{(i+1)(i+1)}$, and for every $1 \leq i \leq r$ it has $s_i h_{i1}$, 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 the edge-disjoint property, we “split” every vertex h_{ij} into two copies h_{ij}^1 and h_{ij}^2 , where the incoming edges of h_{ij} are now going into h_{ij}^1 and the outgoing edges of h_{ij} are going out of h_{ij}^2 and put the edge $h_{ij}^1 h_{ij}^2$ in (see Figure 1). Let’s call this graph D_r and the subgraph induced by four vertices $h_{ij}^1, h_{ij}^2, u_{ij}, v_{ij}$ an *intersection module* of D_r . Again, it is easy to see that there cannot be two *edge-disjoint* paths

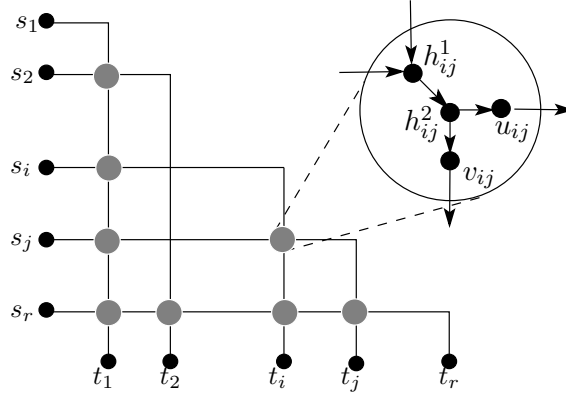


Fig. 1. Construction of D_r : each gray circle corresponds to an intersection module

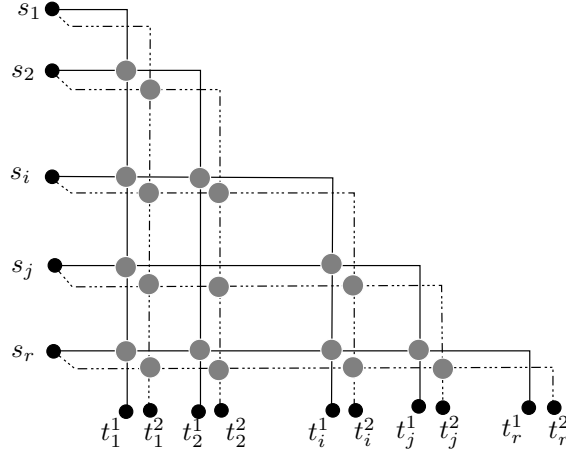
from s_i 's to t_i 's (because we can route at most one path through every intersection module). Note that:

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

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 $t_i s_i$, for $1 \leq i \leq r$. Unfortunately, this new graph will have an integral solution of 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 does not create the desired gap. The problem appears here because the cycles are not bound 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 contains 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 D_r we construct another graph H_r in the following way. Consider D_r with input vertices s_1, \dots, s_r and output vertices t_1, \dots, 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, \dots, s_r be the new set of (unified) input vertices and t_1^1, \dots, t_r^1 and t_1^2, \dots, t_r^2 be the set of output vertices. Let us 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 , $1 \leq i \leq k$. 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_r with $2r$ pairs, either contains only one fully routed block or two partially routed blocks. Furthermore, there is a half-integral solution in which every block is fully routed (with value $\frac{1}{2}$


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

on each path).

We will use the following technical lemma in our construction.

LEMMA 3.4. *For given positive integers r , k , and g with $r < k$, there is an explicit construction of 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, [Lazebnik et al. 1997] 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 us assume that k is a power of 2.

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 at least $2g$. Now we define a 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 at least g and has size $O(\frac{k^{2g}}{r})$. \square

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 hypergraphs exist by Lemma 3.4. The underlying structure of the main graph for the integrality gap is \mathcal{H} . Let p and q be the numbers of vertices and hyperedges of \mathcal{H} , respectively. Take a set $P_r = \{D_r^1, \dots, D_r^p\}$ containing p copies of D_r (constructed earlier), one corresponding to each vertex in \mathcal{H} . Also take a set $Q_k = \{R_k^1, \dots, R_k^q\}$ 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

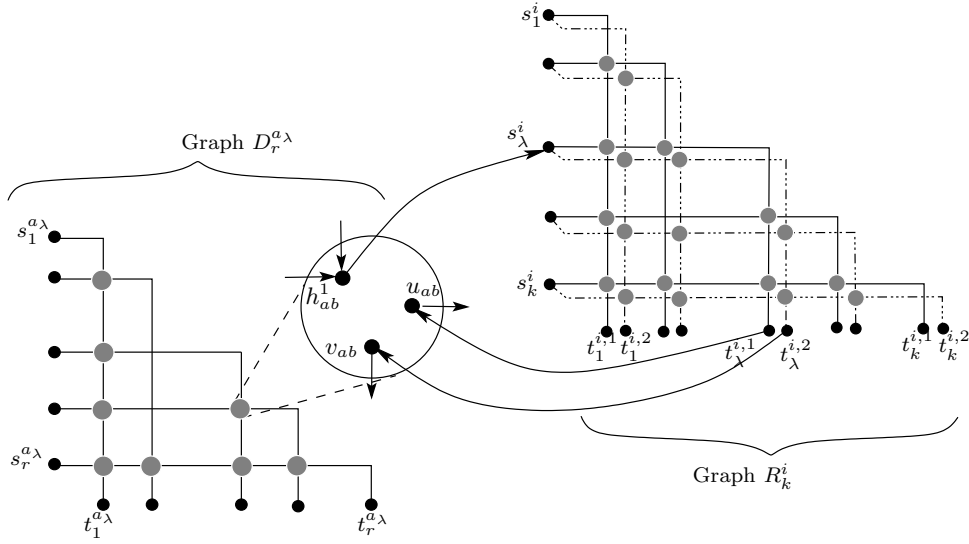


Fig. 3. Replacing intersection modules of $D_1^{a_1}, \dots, D_r^{a_k}$ with blocks of R_k^i

intersection modules of D_r is $\binom{r}{2} = r'$; the same as the 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 that corresponds to hyperedge e_i . Note that R_k^i has k blocks; let us denote these blocks by b_1^i, \dots, b_k^i , where b_λ^i consists of triple $s_\lambda^i, t_\lambda^{1,i}, t_\lambda^{2,i}$. Also, assume that edge e_i consist of vertices v_{a_1}, \dots, v_{a_k} . Thus the copies of D_r in P_r corresponding to these vertices are $D_r^{a_1}, \dots, D_r^{a_k}$; let's denote this set by S_i , i.e. $S_i = \{D_r^{a_j} \in P_r \mid \text{the vertex of } \mathcal{H} \text{ corresponding to } D_r^{a_j} \text{ belongs to } e_i, 1 \leq j \leq k\}$. Pick the first green block of R_k^i , say $s_\lambda^i, t_\lambda^{1,i}, t_\lambda^{2,i}$ (for some $1 \leq \lambda \leq k$) 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 $1 \leq a, b \leq r$) 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: $h_{ab}^1 s_\lambda^i, t_\lambda^{1,i} u_{ab}$, and $t_\lambda^{2,i} v_{ab}$ (see Figure 3). We will consider these new three edges (instead of the three edges that were removed from $D_r^{a_\lambda}$) as part of $D_r^{a_\lambda}$. Do this for all the blocks of R_k^i . 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. $D_r^{a_1}, \dots, D_r^{a_k}$); one for every block of R_k^i . 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 $O(r^2 p + k^2 q)$ vertices. Note that, since each graph $R_k^i \in Q_k$ 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_{ab}^1 to u_{ab} in the intersection module of $D_r^{\alpha\lambda}$, we have to go from s_λ^i to $t_\lambda^{1,i}$ in a “block” of R_k^i . For the moment, assume that:

- . (i) among all the blocks, only an ϵ -fraction could be (partially or fully) routed, and for the other $(1 - \epsilon)$ -fraction no routing existed at all.
- . (ii) all the blocks were 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 .

This would imply that $G_{r,k,g}$ has no more than ϵrp cycles. The reason is that each cycle must contain a feedback edge, and so goes from some s_α in a copy of D_r in P_r to t_α 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_r^i 's in P_r . For instance, a cycle C may start (as a path) at some vertex s_α in a copy D_r^i in P_r (D_r^i corresponds to vertex $v_i \in \mathcal{H}$) and then at some vertex $h_{ab}^1 \in D_r^i$ the path enters the start point of a block b_x in a graph R_k^j (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_k^j) it goes (within R_k^j) to an end point of another block b_y of R_k^j . We call this situation a *jump between blocks* of R_k^j . This way, the path may end-up in another copy $D_r^{i'}$ (before going back to s_α). 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 $D_r^i \in P_r$) to $v_{i'}$ (graph $D_r^{i'} \in P_r$) in \mathcal{H} through hyperedge e_j (through graph $R_k^j \in Q_k$ by starting at the start point of one block and going down to an end point of another block of R_k^j). But if this happens, since the start point (s_α) is in D_r^i , this path must eventually come back to t_α in D_r^i (because $t_\alpha \in D_r^i$ is the only vertex that has an edge to s_α). However, because \mathcal{H} has girth at least 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 3.5. $\nu_e^*(G_{r,k,g}) = \Omega(rp)$, 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 $1 \leq i \leq r$) 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. \square

LEMMA 3.6. $\nu_e(G_{r,k,g}) = O(\frac{q}{r} + \frac{r^2 p}{g})$.

PROOF. By Fact 2, for every graph $R_k^i \in Q_k$, there are 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 3.5, 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 $O(\frac{q}{r} + \frac{r^2 p}{g})$. \square

Proof of Theorem 3.3: Recall that the number of vertices p and the number of hyperedges q of \mathcal{H} are $O(\frac{k^{2g}}{r})$ 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 $G_{r,k,g}$ is $O(r^2 p + k^2 q)$ which is $O(k^{2k+1})$. By Lemmas 3.5 and 3.6, 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 3.3. \square

Combining Lemma 3.1 and Theorem 3.3 and noting that the constructions in Lemma 3.1 have polynomial size, we obtain:

COROLLARY 3.7. *Directed VDC has 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 [Ma and Wang 2000] which is based on the hardness of the label cover problem.

THEOREM 3.8. [Ma and Wang 2000] *For any $\epsilon > 0$, directed EDP cannot be approximated within ratio $2^{\log^{1-\epsilon} n}$ unless $\text{NP} \subseteq \text{DTIME}(2^{\text{poly} \log(n)})$.*

A careful analysis of proof of Theorem 3.8 reveals that in fact their proof implies the following stronger version:

THEOREM 3.9. *Given an instance I of directed EDP, which consists of an acyclic digraph G (on n vertices) and k source-sink pairs $(s_1, t_1), \dots, (s_k, t_k)$ in G , where $k \in \Omega(n^\delta)$ for some absolute $\delta > 0$, then for any $\epsilon > 0$ it is quasi-NP-hard to decide between the following two cases:*

- (1) *All pairs (s_i, t_i) can be routed by disjoint paths, or*
- (2) *At most a fraction $2^{-\log^{1-\epsilon} n}$ of the pairs can be routed.*

We call the instance a *Yes* instance if all the pairs can be routed (case 1 above) and a *No* instance if at most a fraction $2^{-\log^{1-\epsilon} n}$ of the pairs can be routed (case 2 above).

Proof of Theorem 1.4: Let I_{EDP} be an instance of (directed) EDP as in Theorem 3.9 which consists of a directed acyclic graph G and k pairs (s_i, t_i) , $1 \leq i \leq k$. 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 ($1 \leq i \leq k$). Denote this new graph by H_k , with $2k$ source-sink pairs s_i, t_i^1 and s_i, t_i^2 , $1 \leq i \leq k$. 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 the EDP problem with instance H_k and the $2k$ pairs s_i, t_i^1 and s_i, t_i^2 ($1 \leq i \leq k$) 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 an $\binom{r}{g}$ -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 a graph $G_{r,k,g}$ whose underlying structure is \mathcal{H} in the same manner we did in Theorem 3.3 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 $P_r = \{D_r^1, \dots, D_r^p\}$ and $Q_k = \{R_k^1, \dots, R_k^q\}$. We define *short* and *long* cycles in $G_{r,k,g}$ in the same way as we did in Theorem 3.3.

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 $k = \Omega(|I_{EDP}|^\delta)$, this fraction, denoted α , is at most $2^{-\left(\frac{\log k}{\delta}\right)^{1-\epsilon}}$. 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 3.3 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 $O\left(\frac{r^2 p}{g}\right)$. All together, the number of short and long cycles is $O\left(\frac{2\alpha kq}{r} + \frac{r^2 p}{g}\right)$,

which is $O(rp(2\alpha + \frac{r}{g}))$, because $q = \frac{r^2 p}{k}$.

The above arguments, together with Theorem 3.9 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, \mathcal{H} has $p = O(k^{\log^c k})$ vertices and $q = \frac{r^2 p}{k} = O(k^{\log^c k - 1})$ edges. So, if N denotes the number of vertices of $G_{r,k,g}$ (i.e. the size of the construction), then it will be at most $O(k^{\log^c k})$ (for the vertices in graphs in P_r) plus $O(k^{\frac{2}{5} + \log^c k})$ (for the vertices in graphs in Q_k). So overall, $N = O(k^{\frac{2}{5} + \log^c k})$, 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 $\Omega(\log^{\frac{c}{c+1}} N)$. \square

By Lemma 3.1 the same hardness result holds for VDC. Under a stronger complexity assumption that for some sufficiently small $\sigma > 0$, $\text{NP} \not\subseteq \text{DTIME}(2^{n^\sigma})$, we can improve the hardness result to $\Omega\left(\frac{\log n}{(\log \log n)^2}\right)$. To do this we start with an instance \mathcal{L} of label cover of size n and use ℓ parallel repetitions to get a hardness of factor $2^{O(\ell)}$ for the instance \mathcal{L}^ℓ (using the PCP theorem [Arora et al. 1998; Arora and Safra], together with Raz's [Raz 1998] parallel repetition theorem). Note that the size of this instance is $|\mathcal{L}^\ell| = n^{O(\ell)}$. Combining this with the construction of [Ma and Wang 2000] we get a hardness of factor $2^{O(\ell)}$ for the EDP problem. Following the same construction as in the proof of Theorem 1.4 with constant r , $\ell = c \cdot \log n$ for sufficiently large constant c , and $g = \Theta(n^\epsilon)$ (for an $\epsilon < \sigma$) we get a hardness of factor n^ϵ and N (the size of $G_{r,k,g}$) is in $O(n^{n^\epsilon \log n})$. Writing the hardness factor in terms of N , we get a gap of $\Omega\left(\frac{\log N}{(\log \log N)^2}\right)$.

4. CONCLUDING REMARKS

Although there is a large gap between the upper bound (approximation algorithm of Theorem 1.2) and the lower bound (hardness result of Theorem 1.4), closing this gap seems a challenging problem. In fact there are some similarities between the problem of cycle-packing and the well-studied edge-disjoint paths problems (for undirected graphs). For the latter problem, despite several attempts there is still a similar gap between the best known approximation algorithm [Chekuri et al. 2006] (with ratio $O(\sqrt{n})$) and the best known hardness result [Andrews et al. 2005] (with ratio $O(\log^{\frac{1}{2}-\epsilon})$).

For the problem of undirected EDC, very recently Friggstad and Salavatipour [Friggstad and Salavatipour 2006] have proved a hardness of $\Omega(\log^{\frac{1}{2}-\epsilon} n)$, unless $\text{NP} \subseteq \text{DTIME}(n^{\text{poly} \log(n)})$. This shows that the approximation ratio of the greedy algorithm of Theorem 1.1 is almost tight. It would be interesting to find the threshold of approximability of this problem.

Acknowledgment

The authors thank Noga Alon and Guy Kortsarz for useful discussions. We also thank the referee's for their comments.

REFERENCES

- ANDREWS, M., CHUZHUY, J., AND ZHANG, L. 2005. Hardness of the undirected edge-disjoint paths problem with congestion. In *FOCS '05: Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*. IEEE Computer Society, Washington, DC, USA, 226–244.
- ARORA, S., LUND, C., MOTWANI, R., SUDAN, M., AND SZEGEDY, M. 1998. Proof verification and the hardness of approximation problems. *J. ACM* 45, 3, 501–555.
- ARORA, S. AND SAFRA, S. Probabilistic checking of proofs: A new characterization of np. *J. ACM* 45, 1, 70–122.
- BAFNA, V., BERMAN, P., AND FUJITO, T. 1995. Constant ratio approximation of the weighted feedback vertex set problem for undirected graphs. In *In Proc. of 6th International Symposium Algorithms and Computation ISAAC*. Lecture Notes in Computer Science. Springer-Verlag, 142–151.
- BALISTER, P. 2003. Packing digraphs with directed closed trails. *Comb. Probab. Comput.* 12, 1, 1–15.
- BECKER, A. AND GEIGER, D. 1994. Approximation algorithms for the loop cutset problem. In *Proceedings of the Tenth Annual Conference on Uncertainty in Artificial Intelligence, July 29-31, 1994, Seattle, Washington*. 60–68.
- BOLLOBÁS, B. 2004. *Extremal Graph Theory*. Dover Publications, Incorporated.
- BOLLOBÁS, B., ERDŐS, P., SIMONOVITS, M., AND SZEMERÉDI, E. 1978. Extremal graphs without large forbidden subgraphs. *Ann. Discrete Math.* 3, 29–41.
- BOLLOBÁS, B. AND THOMASON, A. 1997. On the girth of hamiltonian weakly pancyclic graphs. *J. Graph Theory* 26, 3, 165–173.
- CAPRARA, A., PANCONESI, A., AND RIZZI, R. 2003. Packing cycles in undirected graphs. *J. Algorithms* 48, 1, 239–256.
- CHEKURI, C. AND KHANNA, S. 2003. Edge disjoint paths revisited. In *SODA '03: Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 628–637.
- CHEKURI, C., KHANNA, S., AND SHEPHERD, B. 2006. An $o(\sqrt{n})$ approximation and integrality gap for disjoint paths and ufp. *Theory of Computing* 2, 137–146.
- CHUDAK, F. A., GOEMANS, M. X., HOCHBAUM, D. S., AND WILLIAMSON, D. P. 1998. A primal-dual interpretation of two 2-approximation algorithms for the feedback vertex set problem in undirected graphs. *Oper. Res. Lett.* 22, 4-5, 111–118.
- DOR, D. AND TARSI, M. 1992. Graph decomposition is npc - a complete proof of holyer's conjecture. In *STOC '92: Proceedings of the twenty-fourth annual ACM symposium on Theory of computing*. ACM Press, New York, NY, USA, 252–263.
- ERDŐS, P. AND SACHS, H. 1963. Reguläre graphen gegebener taillenweite mit minimaler knotenzahl. *Wiss. Z. Martin-Luther Univ. Halle-Wittenberg Math.-Natur. Reihe* 12.
- EVEN, G., NAOR, J., SCHIEBER, B., AND SUDAN, M. 1998. Approximating minimum feedback sets and multicuts in directed graphs. *Algorithmica* 20, 2, 151–174.
- FRIGGSTAD, Z. AND SALAVATIPOUR, M. 2006. unpublished manuscript.
- GURUSWAMI, V., KHANNA, S., RAJARAMAN, R., SHEPHERD, B., AND YANNAKAKIS, M. 2003. Near-optimal hardness results and approximation algorithms for edge-disjoint paths and related problems. *J. Comput. Syst. Sci.* 67, 3, 473–496.
- HAJIAGHAYI, M. T. AND LEIGHTON, T. 2006. On the max-flow min-cut ratio for directed multi-commodity flows. *Theor. Comput. Sci.* 352, 1, 318–321.
- KOMLÓS, J. 1997. Covering odd cycles. *Combinatorica* 17, 3, 393–400.
- LAZEBNIK, F., USTIMENKO, V. A., AND WOLDAR, A. J. 1997. New upper bounds on the order of cages. *Electr. J. Comb.* 4, 2.
- MA, B. AND WANG, L. 2000. On the inapproximability of disjoint paths and minimum steiner forest with bandwidth constraints. *J. Comput. Syst. Sci.* 60, 1, 1–12.
- RAZ, R. 1998. A parallel repetition theorem. *SIAM J. Comput.* 27, 3, 763–803.
- SEYMOUR, P. D. 1995. Packing directed circuits fractionally. *Combinatorica* 15, 2, 281–288.

VARADARAJAN, K. AND VENKATARAMAN, G. 2004. Graph decomposition and a greedy algorithm for edge-disjoint paths. In *Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*. 379–380.