SIAM J. DISCRETE MATH. © 2018 Society for Industrial and Applied Mathematics
Vol. 32, No. 3, pp. 2134–2160

CONSTANT CONGESTION ROUTING OF SYMMETRIC DEMANDS IN PLANAR DIRECTED GRAPHS*

CHANDRA CHEKURI†, ALINA ENE‡, AND MARCIN PILIPCZUK§

Abstract. We study the problem of routing symmetric demand pairs in planar digraphs. The input consists of a directed planar graph $G = (V, E)$ and a collection of k source-destination pairs $\mathcal{M} = \{s_1t_1, \ldots, s_kt_k\}$. The goal is to maximize the number of pairs that are routed along disjoint paths. A pair s_it_i is routed in the symmetric setting if there is a directed path connecting s_i to t_i *and* a directed path connecting t_i to s_i. In this paper we obtain a randomized polylogarithmic approximation with constant congestion for this problem in planar digraphs. The main technical contribution is to show that a planar digraph with directed treewidth h contains a relaxed cylindrical grid (which can serve as a constant congestion crossbar in the context of a routing algorithm) of size $\Omega(h/\text{polylog}(h))$.

Key words. planar digraphs, disjoint paths, routing, directed treewidth

AMS subject classifications. 68Q25, 68R10, 68W25, 05C10, 05C20, 05C85

DOI. 10.1137/17M1150694

1. Introduction.

Disjoint path problems are well-studied routing problems with several applications and fundamental connections to algorithmic and structural results in combinatorial optimization and graph theory. Canonical problems here are the edge-disjoint paths problem (EDP) and the node-disjoint paths problem (NDP) in undirected graphs. In both these problems the input consists of an undirected graph $G = (V, E)$ and k node-pairs $\{s_1t_1, \ldots, s_kt_k\}$. In EDP the goal is to connect the pairs by edge-disjoint paths and in NDP the goal is to connect the pairs by node-disjoint paths. The decision versions of these problems are NP-Complete when k is part of the input [29]. The seminal work of Robertson and Seymour showed that both these problems are fixed parameter tractable when parameterized by k, the number of pairs [41]. In this paper we are concerned with an optimization version of the problems where the goal is to maximize the number of input pairs that can be routed via edge or node-disjoint paths. To avoid notational overload we will henceforth use EDP and NDP to refer to these maximization versions.

The approximability of EDP and NDP has been extensively studied but our understanding is still limited. The best known approximation for both these problems is $O(\sqrt{n})$ [11, 34]. (Here n is number of nodes in G.) Hardness of approximation results only ruled out an $O(\log^{1/2-\varepsilon} n)$ approximation [2] until very recently when Chuzhoy, Kim, and Nimavat showed a superpolylogarithmic lower bound [17] which holds even for planar graphs. Even in planar graphs the best approximation up to very recently was $O(\sqrt{n})$, with a slight recent improvement [18] for NDP. One of the

*Received by the editors October 5, 2017; accepted for publication (in revised form) May 14, 2018; published electronically August 21, 2018. An extended abstract of this work has been presented at ICALP 2016 [8].
http://www.siam.org/journals/sidma/32-3/M115069.html
Funding: The first author's work was partially supported by NSF grant CCF-1319376. The third author's work was partially supported by DIMAP and by Warwick-QMUL Alliance in Advances in Discrete Mathematics and its Applications.
†Department of Computer Science, University of Illinois, Urbana-Champaign, Urbana, IL 61801 (chekuri@illinois.edu).
‡Department of Computer Science, Boston University, Boston, MA 02215 (aene@bu.edu).
§Institute of Informatics, University of Warsaw, Warsaw, 02-097, Poland (malcin@mimuw.edu.pl).

reasons for this state of affairs is that the natural multicommodity flow relaxation has an integrality gap of $\Theta(\sqrt{n})$. On the other hand, two closely related relaxations of EDP and NDP have seen significant progress in the last decade. ANF is the relaxation of the disjoint paths problem where a subset of the input pairs $\mathcal{M}'$ is routed if there is a feasible multicommodity flow in the graph that routes one unit of flow for each pair in $\mathcal{M}'$. A second relaxation is to allow some small constant congestion c, i.e., instead of the pairs being routed on disjoint paths we allow up to c paths to use a given edge or node. ANF admits a polylogarithmic approximation [12, 10]. A series of breakthroughs [37, 1, 15] culminated in a polylogarithmic approximation for EDP with congestion 2 by Chuzhoy and Li [19]. These ideas have been extended to NDP as well [6, 5]. These results have been made possible by a number of nontrivial ideas and techniques at the intersection of algorithms, combinatorial optimization, and graph theory. In particular, the results have been enabled by and contributed to a deeper understanding of the structure of undirected graphs via the notion of treewidth. Treewidth is a well-known graph parameter that plays a fundamental role in the graph-minor theory of Robertson and Seymour; see [3, 5, 4, 14] for some of the recent results.

It is natural to study disjoint paths problems in *directed* graphs. Here the graph G is directed and the input pairs $\mathcal{M} = \{(s_1, t_1), \dots, (s_k, t_k)\}$ are ordered and we seek to find a maximum cardinality subset of $\mathcal{M}$ that can be connected by disjoint paths.[1] Unfortunately, it has been shown that disjoint paths problems are highly intractable in directed graphs. It is known that even the simpler case of ANF and with congestion c allowed is hard to approximate to within a factor of $n^{\Omega(1/c)}$ [16]; moreover, this holds in acyclic graphs.

A recent paper by a subset of the authors [7] initiated the study of maximum throughput routing problems in directed graphs where the demand pairs are *symmetric*. Here the graph G is directed but the input pairs are unordered as in the undirected setting. Routing a pair s_it_i requires finding a path that connects s_i to t_i *and* a path connecting t_i to s_i. We use Sym-Dir-EDP, Sym-Dir-NDP, and Sym-Dir-ANF to denote the analogues of EDP, NDP, and ANF, respectively, in this setting. A detailed motivation for the study of this model is given in [7]. Here we briefly outline some of the key points.

The model is motivated by both theoretical and practical considerations. On the theoretical side the model generalizes (modulo constant congestion) the edge and node disjoint paths problems in undirected graphs. Moreover, flow-cut gaps in this model have been studied in the past and have close connections to various problems including feedback edge/vertex set problems [33, 43, 23, 9]. From the more practical side there are several scenarios where the communication between users is symmetric while the underlying network that supports the communication may be asymmetric (hence modeled as a directed graph); see [28, 27], for instance.

Unlike the case of directed graph routing problems, the symmetric model exhibits tractability. In particular, the well-linked decomposition framework for undirected graphs extends to a large extent to this model [7]. To resolve the complexity of disjoint path problems in the symmetric model one needs to understand the structure of directed graphs as a function of their *directed treewidth* [25, 38], which we denote by $\mathrm{dtw}(G)$. As we mentioned, the interplay between routing problems and graph structure theory has been fruitful in the recent past for undirected graphs. There has been

[1] Although edge and node disjoint paths problems are equivalent in general directed graphs, this is not necessarily the case in restricted graph classes such as planar graphs.

recent significant progress on the graph theoretic side on directed treewidth; in particular Kawarabayashi and Kreutzer recently established the excluded grid theorem in directed graphs [30, 31].

The main technical contribution of [7] is to generalize the well-linked decomposition framework of [10] to the symmetric demands setting in directed graphs. As a consequence, [7] obtained a polylogarithmic approximation with constant congestion for Sym-Dir-ANF. The central open question they raised is the following: Is there a polylogarithmic approximation for Sym-Dir-NDP with constant congestion in general directed graphs? It was shown in [7] that this can be answered in the positive by addressing the following, which is the analogue of the question that was raised previously for undirected graphs [10]: If a directed graph G has directed treewidth h, does it have a constant congestion routing structure (crossbar) of size $\Omega(h/\text{polylog}(h))$? Note that grid-minor theorems establish such a connection between treewidth and routing structures; however, the quantitative relationship between the treewidth and the size of the grid is too weak to prove any meaningful approximation for the routing problem. On the other hand, the routing problem has the flexibility of allowing a large constant congestion which enables one to prove the existence of routing structures that are not as rigid as a grid; this relaxation has been the key to algorithmic success on routing. We also note that it is NP-complete to decide whether a single pair can be routed without congestion in the symmetric setting [24]; thus a congestion of at least 2 is necessary for a nontrivial approximation ratio.

In this paper we take a step toward the general problem by addressing the important special case of planar graphs. Our main algorithmic result is the following.

THEOREM 1.1. *There is a randomized polylogarithmic approximation both for* Sym-Dir-NDP *and* Sym-Dir-EDP *in planar directed graphs with congestion* 5.

The approximation algorithm in the preceding theorem is derived via a natural multicommodity flow relaxation for the problem. The main new technical ingredient in this paper is a graph theoretic result: if a planar digraph has directed treewidth h, then it has a relaxed cylindrical grid (defined formally later) of size $\Omega(h/\text{polylog}(h))$; such a grid can serve as a constant congestion crossbar for the purpose of a routing algorithm. We remark that an undirected planar graph with treewidth h has a grid-minor (which is a congestion 2 crossbar) of size $\Omega(h)$. In contrast the known relationship between treewidth and grid-minors in directed planar graphs is much weaker; recent work [30, 31] only shows that there is a directed-grid of size $f(h)$ for some weakly growing function of h. We hope that our crossbar result could be used as a starting point to improve the quantitative bound on the grid-minor theorem for planar digraphs.

1.1. Overview of the algorithm and technical contributions. Here we give a brief outline of the high-level details of the algorithm and some of our technical contributions. Let $(G, \mathcal{M})$ be an instance of Sym-Dir-NDP, where $G = (V, E)$ is a directed planar graph with unit node capacities, and $\mathcal{M} = \{s_1t_1, \ldots, s_kt_k\}$ is a collection of source-destination pairs. We refer to the nodes participating in $\mathcal{M}$ as terminals, and we use $\mathcal{T}$ to denote the set of terminals. It is convenient to assume that the pairs $\mathcal{M}$ form a matching on $\mathcal{T}$.

Well-linked sets. A key notion that we make use of is well-linkedness. In a directed graph $G = (V, E)$ a subset of nodes $X \subseteq V$ is said to be node-well-linked if for any two disjoint subsets Y and Z of X of equal size, there exist $|Y|$ node-disjoint paths

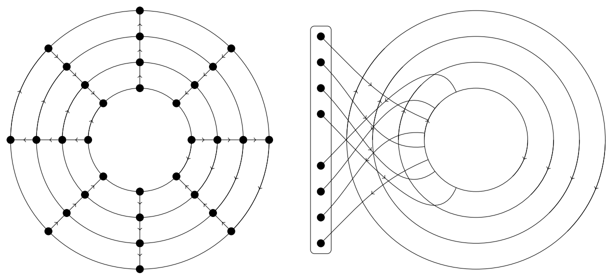

FIG. 1. *A cylinder (left) and a crossbar we obtain in our proof (right); the vertices in the rounded rectangle are a set of α-well-linked terminals.*

from Y to Z;[2] note that the definition is symmetric since we can swap Y and Z. We need a relaxation of node-well-linkedness. For some parameter $\alpha \in [0, 1]$, X is α-node-well-linked if for all disjoint $Y, Z \subset X$ of equal size there are $|Y|$ paths from Y to Z such that no node is in more than $\lceil 1/\alpha \rceil$ of these paths; in other words, the node-congestion caused by the paths is at most $\lceil 1/\alpha \rceil$. The case $\alpha = 1$ corresponds to node-well-linkedness. It is well-known that in both directed and undirected graphs node-well-linkedness is closely connected to treewidth. More precisely, a graph has treewidth k if and only if it has a node-well-linked set of size $\Theta(k)$; see [38]. Moreover, if X is α-node-well-linked in G, then the treewidth of G is $\Omega(\alpha|X|)$.

Throughout the paper we only work with well-linked sets and do not need the notion of directed treewidth. For this reason we refrain from stating and explaining the highly technical definition of directed treewidth; it is not straightforward to digest the formal definition. We refer the interested reader to [25, 38].

Algorithm. Here we outline the high-level steps of our algorithm.

1. Solve a multicommodity flow based LP relaxation that routes each pair $s_i t_i$ fractionally to an amount $x_i \in [0, 1]$ to maximize $\sum_{i=1}^{k} x_i$. See Figure 2 and the description in section 2.
2. Use the LP relaxation and the well-linked decomposition framework from [7] to reduce the problem, at the loss of a polylogarithmic factor in the approximation ratio, to instances in which the terminals $\mathcal{T}$ are α-node-well-linked for some *fixed* constant α.
3. Assuming that $\mathcal{T}$ is α-node-well-linked in G we have $\mathrm{dtw}(G) = \Omega(k)$, where $k = |\mathcal{T}|$. Using this fact show that G has a large routing structure and use this structure to route a large number of terminal pairs. Use the following steps.
   (a) From G obtain an *Eulerian multigraph* $H = (V, E_H)$ whose support is a subgraph of G such that (i) $\mathcal{T}$ is α'-node-well-linked in H for $\alpha' = \Omega(\frac{1}{\mathrm{polylog}(k)})$ and (ii) $\Delta(H)$, the maximum degree in H, is $\mathrm{polylog}(k)$.
   (b) Using the fact that H is Eulerian, has treewidth $\Omega(k/\mathrm{polylog}(k))$, and has maximum degree $\mathrm{polylog}(k)$, show that it has a cylinder-like routing structure of size $\Omega(k/\mathrm{polylog}(k))$. See Figure 1.
   (c) Route terminals to the routing structure and use the routing structure to connect a large number of input pairs.

[2] Throughout the paper when we say paths from Y to Z we mean a collection of paths with each node of Y as the start node of at most one of the paths, and each node of Z as the end node of at most one of the paths.

The preceding algorithm follows the general framework that has been very successful in the undirected graph setting in the recent past. The first two steps follow the well-linked decomposition framework from [10] that has been extended to the symmetric demand instances in directed graphs by [7]. This framework allows one to reduce, via the LP relaxation, general instances to instances in which the terminals are node-well-linked. This incurs a polylogarithmic factor loss in the approximation. With this reduction in place we have the following property for our instance. The graph G has a terminal set $\mathcal{T}$ of size k and since $\mathcal{T}$ is α-node-well-linked for some fixed constant α, G has directed treewidth $\Omega(k)$. Now, the remaining task is to show a graph-theoretic result that any directed graph with treewidth k has a constant congestion *crossbar* routing structure of size $\Omega(k/\mathrm{polylog}(k))$. By crossbar we mean a directed graph H with an *interface* $I \subset V(H)$ with the following property: *any* matching on I can be routed in a symmetric fashion in H with constant congestion. The idea then is to route the terminals to the interface of the crossbar and use the crossbar to route the desired matching on the terminals.

In undirected planar graphs if G has treewidth k, then it has grid-minor of size $\Omega(k)$ [42], and this grid-minor can be used as a crossbar to route $\Omega(k)$ input pairs (see [10], for instance). What about directed graphs? Johnson et al. [25], who introduced the notion of directed treewith, conjectured that any directed graph with sufficiently large treewidth contains a cylindrical grid (see Figure 1) as a butterfly minor. The cylindrical grid can be used as a crossbar. In an unpublished manuscript, Johnson et al. [26] outlined a proof for the case of planar graphs. Kawarabayashi and Kreutzer [30] recently gave a different proof for the planar and minor-free case and very recently gave a proof for all graphs [31]. However, as we already mentioned, the quantitative relationship between the size of the cylindrical grid and treewidth is very weak. Hence, these results would not yield meaningful results for our routing problem. Here, we build on the high-level ideas in the work of Johnson et al. [26] to establish our main result, which gives a constant congestion crossbar of size $\Omega(k/\mathrm{polylog}(k))$, where k is the treewidth of G; our result applies only to planar graphs and establishing a similar result for general graphs is a challenging open problem.

A key insight from [26] is that given directed graph G one can create an Eulerian multigraph H of bounded degree whose support is a subgraph of G such that $\mathrm{dtw}(H) \geq f(\mathrm{dtw}(G))$ for some function f, where $\mathrm{dtw}(G)$ is the directed treewidth of G [25]. Eulerianness as well as small degree are critical for further manipulations. Our first contribution is to show that H can be chosen such that (i) $\mathrm{dtw}(H) = \mathrm{dtw}(G)/\mathrm{polylog}(\mathrm{dtw}(G))$ and (ii) the maximum degree in H, $\Delta(H) = O(\log^2 \mathrm{dtw}(G))$. For this purpose we need two ingredients. The first is an extension to directed graphs due to Louis [36], of the cut-matching game of Khandekar, Rao, and Vazirani [32]. The second is the well-linked decomposition framework of [7, 10].

THEOREM 1.2. *Suppose that there is a polynomial time algorithm for $\Omega(1)$-node-well-linked instances of* Sym-Dir-NDP *in planar directed Eulerian graphs of maximum degree Δ that achieves a $\beta(\Delta)$-approximation with congestion c. Then there is a polynomial time randomized algorithm that, with high probability, achieves a $\beta(O(\log^2 k)) \cdot O(\log^6 k)$ approximation with congestion c for arbitrary instances of* Sym-Dir-NDP *in planar directed graphs, where k is the number of pairs in the instance.*

Another key insight from [26] is to consider the undirected version of G, denoted by G^{UN}, to obtain a large undirected grid-minor using the fact that $\mathrm{tw}(G^{\mathrm{UN}}) = \Omega(\mathrm{dtw}(G))$. In particular, this allows the construction of several disjoint concentric directed cycles in G by exploiting the structure of the grid, Eulerianness, and planarity. We follow their ideas and show that the entire construction can be done in polynomial time to yield $\Omega(\mathrm{dtw}(H)/\Delta(H))$ concentric disjoint cycles.

The final step is to find many disjoint paths that cross the concentric cycles from the inner cycle to the outer cycle and many disjoint paths from the outer cycle to the inner cycle. We show that we can find such paths via some ideas in [26] but with the additional property that these paths originate at the terminals. The collection of concentric cycles with these crossing paths is our desired crossbar and we also obtain the required property that the terminals are linked to this crossbar. We note that [26] has to do considerable work to obtain a cylindrical grid while we are satisfied with the constant congestion properties of the cycles plus paths (see Figure 1).

In the end, we arrive at the following statement, whose proof is presented in section 4.

THEOREM 1.3. *There is a polynomial-time algorithm that given a directed Eulerian graph G of maximum in-degree at most Δ, a planar embedding of G, and an α-node-well-linked set X in G with $|X| = \Omega(\Delta^2/\alpha)$ finds a set of $\Omega(\alpha|X|/\Delta)$ concentric cycles going in the same direction (i.e., all clockwise or all counterclockwise), sets $Y^+, Y^- \subseteq X$ of size $|Y^+| = |Y^-| = \Omega(\alpha^2|X|/\Delta^2)$ each, and families $\mathcal{P}^+$ and $\mathcal{P}^-$ of node-disjoint paths, such that either*

(1) *none of the cycles enclose any vertex of $Y^+ \cup Y^-$, the family $\mathcal{P}^+$ consists of $|Y^+|$ node-disjoint paths from Y^+ to the innermost cycle, and the family $\mathcal{P}^-$ consists of $|Y^-|$ node-disjoint paths from the innermost cycle to Y^-, or*

(2) *all cycles enclose $Y^+ \cup Y^-$, the family $\mathcal{P}^+$ consists of $|Y^+|$ node-disjoint paths from Y^+ to the outermost cycle, and the family $\mathcal{P}^-$ consists of $|Y^-|$ node-disjoint paths from the outermost cycle to Y^-.*

Although we are inspired by [26], in the proof of Theorem 1.3 we use different methodology based on well-linked sets. We also point out that there are significant technical hurdles in working with directed graphs and treewidth. For instance, one can prove that if an undirected graph has treewidth k, then it has $\Omega(k/\log k)$ disjoint cycles. This is closely related to the well-known Erdős–Pósa theorem [22]. Relating treewidth and disjoint cycles in directed graphs is significantly harder and was resolved in [39] following an earlier result for planar graph [40] (and also via the more recent result [31]) but the quantitative relationship is weak and far from the known lower bounds.

Using Theorem 1.3, we show the following statement, which in turn, together with Theorem 1.2, immediately yields Theorem 1.1.

THEOREM 1.4. *There is an $\mathcal{O}(\Delta^2/\alpha^3)$ approximation with congestion* 5 *for* Sym-Dir-NDP *in instances for which the input digraph is planar and Eulerian with maximum degree Δ, and the terminals are α-node-well-linked for some $\alpha \leq 1$.*

Organization. After introducting notation and tools in section 2, we prove Theorem 1.2 in section 3 and Theorem 1.3 in section 4. Section 5 wraps up the proof of Theorem 1.1 by establishing Theorem 1.4.

2. Preliminaries.

In the remainder of this paper, we focus on the *node-disjoint paths* problem (Sym-Dir-NDP) in directed planar graphs with symmetric demands, and we give a polylogarithmic approximation with constant congestion for the problem. Furthermore, we show the approximation guarantee with regards to the value standard LP relaxation of the problem; in section 2.2 we discuss how we reduce Sym-Dir-NDP to Sym-Dir-EDP, proving both statements of Theorem 1.1.

$$
\begin{aligned}
&\text{(Sym-Dir-NDP LP)}\\
&\max \sum_{i=1}^{k} x_i, \\
&\sum_{p\in\mathcal{P}(s_i,t_i)} f(p) = x_i && \forall i\in[k],\\
&\sum_{p\in\mathcal{P}(t_i,s_i)} f(p) = x_i && \forall i\in[k],\\
&\sum_{p:\, v\in p} f(p) \le 1 && \forall v\in V(G),\\
&f(p)\ge 0 && \forall p\in\mathcal{P},\\
&x_i\in[0,1] && \forall i\in[k].
\end{aligned}
$$

FIG. 2. *Relaxation* Sym-Dir-NDP LP.

We start with some definitions and results from previous work. We follow the notation and exposition of [7]. Let $(G, \mathcal{M})$ be an instance of Sym-Dir-NDP, where $G = (V, E)$ is a directed planar graph with unit node capacities, and $\mathcal{M} = \{s_1t_1, \ldots, s_kt_k\}$ is a collection of source-destination pairs. We refer to the nodes participating in $\mathcal{M}$ as terminals, and we use $\mathcal{T}$ to denote the set of terminals.

2.1. LP relaxation. Our algorithm uses a standard multicommodity flow relaxation for the problem given in Figure 2. We use $\mathcal{P}(u, v)$ to denote the set of all paths in G from u to v for each ordered pair (u, v) of nodes. Our assumption that the pairs $\mathcal{M}$ form a matching ensures that the sets $\mathcal{P}(s_i, t_i)$, $\mathcal{P}(t_i, s_i)$, $\mathcal{P}(s_j, t_j)$, and $\mathcal{P}(t_j, s_j)$ are pairwise disjoint. Let $\mathcal{P} = \bigcup_{i=1}^{k}(\mathcal{P}(s_i, t_i) \cup \mathcal{P}(t_i, s_i))$. The LP has a variable $f(p)$ for each path $p \in \mathcal{P}$ representing the amount of flow on p. For each (unordered) pair $s_it_i \in \mathcal{M}$, the LP has a variable x_i denoting the total amount of flow routed for the pair. (In the corresponding IP, x_i denotes whether the pair is routed or not.) The LP imposes the symmetry constraint that there is a flow from s_i to t_i of value x_i *and* a flow from t_i to s_i of value x_i. Additionally, the LP has capacity constraints that ensure that the total amount of flow on paths using a given node is at most one.[3]

It is convenient to assume that the pairs $\mathcal{M}$ form a matching on $\mathcal{T}$ and each terminal is a leaf of G, i.e., it is attached to a single neighbor using an edge in each direction. As shown in [7], these properties can be ensured as follows. Given an instance $(G, \mathcal{M})$ with terminals $\mathcal{T}$, we create a new instance $(G', \mathcal{M}')$ by attaching a new leaf neighbor t' to every $t \in \mathcal{T}$ with arcs (t, t') and (t', t), and move the terminal t to t'. Given a solution to the LP relaxation on $(G, \mathcal{M})$, we can easily find a solution of at least half of the value by extending the flow along arcs (t, t') and (t', t); the loss of the flow is due to potential capacity violation at vertex t that is now counted twice along the flow paths. If we obtain an integral solution in $(G', \mathcal{M}')$ (i.e., a routing of

[3]There is a subtle issue here with regard to the capacity usage at the endpoints of a path. In the integral solution, a pair of paths, one from s_i to t_i and one from t_i to s_i, is regarded as using the vertex s_i only once and using the vertex t_i only once; in other words, such a pair can be seen as a simple cycle passing through s_i and t_i. To simulate it in the LP relaxation, we consider that the starting vertex belongs to a flow path, but the ending vertex does not belong to it. Alternatively, we can assume that a flow path uses only half of the capacities at its endpoints; these interpretations are equivalent due to the symmetry of the demands.

some pairs from $\mathcal{M}'$) with congestion c, by shortening the paths we obtain a routing with the same congestion in $(G, \mathcal{M})$.

2.2. Reduction between Sym-Dir-NDP and Sym-Dir-EDP. As discussed at the beginning of this section, in the remainder of this paper we develop an approximation algorithm for Sym-Dir-NDP whose approximation guarantee is actually with respect to the value of the aforementioned LP relaxation. We now discuss how to obtain from this result an analogous algorithm for Sym-Dir-EDP, with the same congestion guarantee.

Consider an instance $(G, \mathcal{M})$ of Sym-Dir-EDP with some fixed plane embedding of G. Similarly as in the case of Sym-Dir-NDP, we can assume that all terminals are leaves (of degree 1) in G. First, we subdivide every edge e of G with a new vertex x_e; the purpose of the new vertex is to keep capacity 1 for the edge e in the node-disjoint setting. Second, for every vertex v of the original graph G of degree $d \geq 3$, we replace v with a $d \times d$ bidirectional grid Γ_d and connect edges formerly incident to d to different vertices on one fixed side of the grid Γ_d. Given the fixed plane embedding of G, these edges can be connected to Γ_d in an order preserving planarity. Let G' be the resulting graph; we treat $(G', \mathcal{M})$ as an Sym-Dir-NDP instance.

Consider now a feasible solution to the natural LP relaxation of Sym-Dir-EDP on $(G, \mathcal{M})$ (i.e., an relaxation as we use for Sym-Dir-NDP, but with capacity constraints on edges instead of vertices). It is easy to see that this solution, scaled down by 1/2, can be modified to obtain a feasible solution to the LP relaxation of Sym-Dir-NDP on $(G', \mathcal{M})$ by appropriately routing the flow passing through v in G via the grid Γ_v in G'.

Indeed, to see this, without loss of generality, assume that the edges incident with v have been attached in G' to the top row of the grid Γ_v and an edge e incident with v in G has been attached to the endpoint y_e of a column C_e of Γ_d. For every such e, additionally pick a private row R_e of Γ_d, and redirect the flow incoming along e and leaving via e' in G via the column C_e, row R_e, and column $C_{e'}$. Since the flow in G' has been rescaled by 1/2, the total flow redirected along column C_e is at most 1/2 (because in G a flow of at most 1 passes e) and for the same reason the total flow redirected along row R_e is also at most 1/2.

Consequently, the optimum value of the LP relaxation of Sym-Dir-NDP on $(G', \mathcal{M})$ is at least half of the optimum value of the LP relaxation of Sym-Dir-EDP on $(G, \mathcal{M})$. We now use our algorithm for Sym-Dir-NDP on the instance $(G', \mathcal{M})$, obtaining a congestion-5 routing of a number of terminal pairs within polylogarithmic factor of the optimum value of the LP relaxation. Finally, it is straightforward to project this routing back to edge-disjoint paths (with the same congestion) in G.

2.3. Multicommodity flows and sparse node separators. We represent a multicommodity flow instance as a demand vector $\mathbf{d}$ that assigns a demand $d(u, v) \in \mathbb{R}_+$ to each ordered pair (u, v) of vertices of G. The instance is *symmetric* if $d(u, v) = d(v, u)$ for all pairs (u, v). A *product* multicommodity flow instance satisfies $d(u, v) = w(u)w(v)$ for each pair (u, v), where $w : V \to \mathbb{R}_+$ is a weight functions on the vertices of G. (Note that product multicommodity flows are symmetric.) We say that $\mathbf{d}$ is *routable* if there is a feasible multicommodity flow in G that routes $d(u, v)$ units of flow from u to v for each pair (u, v).

We recall the following two quantities associated with a symmetric multicommodity flow instance: the maximum concurrent flow and the sparsest node separator. The *maximum concurrent flow* is the maximum value $\lambda \geq 0$ such that $\lambda\mathbf{d}$ is routable. A *node separator* is a set $C \subseteq V$ of nodes. The removal of a node separator gives us one or more strongly connected components; we say that an unordered pair uv is separated by C if u and v are *not* in the same strongly connected component of $G \setminus C$. The demand separated by C, denoted by $\mathrm{dem}_{\mathbf{d}}(C)$, is the total demand of all of the unordered pairs separated by C; more precisely, $\mathrm{dem}_{\mathbf{d}}(C) = \sum_{uv \text{ separated by } C} d(u,v)$. The *sparsity* of a node separator C is $\mathrm{cap}(C)/\mathrm{dem}_{\mathbf{d}}(C)$, where $\mathrm{cap}(C)$ is the sum of the capacities of the vertices in C. A *sparsest node separator* is a separator with minimum sparsity.

The minimum sparsity of a node separator is an upper bound on the maximum concurrent flow. The *flow-cut gap* in G is the maximum value—over all symmetric multicommodity flow instances $\mathbf{d}$ in G—of the ratio between the minimum sparsity of a node separator and the maximum concurrent flow. The flow-cut gap in any graph is $O(\log^3 k)$, where k is the number of commodities (each pair (u,v) with nonzero demand is a commodity) [33]. For product multicommodity flows, the flow-cut gap is $O(\log k)$ and there is a polynomial time algorithm that, given a product multicommodity flow instance $\mathbf{d}$ in G, constructs a node separator C whose sparsity is at most $O(\log k)\lambda$, where λ is the maximum concurrent flow for $\mathbf{d}$ [35]; we use such an algorithm in a black box fashion in the well-linked decomposition step described in the following subsection.

2.4. Well-linked sets and decompositions. Here we discuss several useful notions of well-linkedness in directed graphs. The corresponding definitions for the case of undirected graphs can be found in [10, 6]. There are several notions of well-linked sets that have been studied in the literature. In this paper, we work with *cut* well-linked sets; for simplicity, we refer to these sets as simply well-linked. (There is also a notion of *flow* well-linkedness, and the two notions are related via the flow-cut gap [7].)

We consider two variants of well-linked sets, *edge well-linked* sets and *node well-linked* sets. (The two notions are related via the maximum degree of the graph, as we remark below.)

A set $X \subseteq V$ is edge (resp., node) well-linked in G if, for any two disjoint subsets Y and Z of X of equal size, there exist $|Y|$ edge-disjoint (resp., node-disjoint) paths from Y to Z in G such that every node in Y is the start vertex of exactly one path and every node in Z is the end vertex of exactly one path. (Note that the well-linkedness guarantees that there exist paths routing Y to Z but we do not have any control over how Y is matched to Z.)

An equivalent definition of edge well-linkedness is the following. The set X is edge well-linked if, for any cut $(A, V \setminus A)$, $|\delta^{\mathrm{out}}(A)| \geq \min\{|X \cap A|, |X \cap (V \setminus A)|\}$. Here $\delta^{\mathrm{out}}(A)$ is the set of edges leaving A. If the nodes of X have degree 1 in G, one can give the following equivalent definition of well-linkedness. A node separation is a partition $V = A \uplus B \uplus C$ such that there is no edge between A and B. The set X is node well-linked if, for any node separation (A, B, C) satisfying $X \cap C = \emptyset$, we have $|C| \geq \min\{|X \cap A|, |X \cap B|\}$.

These equivalent definitions make it straightforward to extend the notions of well-linkedness to the fractional setting.

Let $\pi : X \to [0,1]$ be a weight function on X. The set X is π-edge-well-linked if, for any cut $(A, V \setminus A)$, $|\delta^{\mathrm{out}}(A)| \geq \min\{\pi(A), \pi(V \setminus A)\}$. If all of the nodes in X are leaves, X is π-node-well-linked in G if, for any node separation (A, B, C) with $X \cap C = \emptyset$, we have $|C| \geq \min\{\pi(A), \pi(B)\}$.

Clearly, if X is π-node-well-linked in G, then X is also π-edge-well-linked. Conversely, if X is π-edge-well-linked in G, then X is $\Omega(\pi/\Delta)$-node-well-linked in G, where Δ is the maximum degree of G.

The work of [7] gives the following well-linked decomposition and clustering procedure for directed graphs.

THEOREM 2.1 (see [7]). *Let* OPT *be the value of a solution to* Sym-Dir-NDP LP *for a given instance* $(G, \mathcal{M})$ *of* Sym-Dir-NDP. *Let* $\gamma = \gamma(G) \geq 1$ *be an upper bound on the worst case flow-cut gap for product multicommodity flows in* G. *There is a partition of* G *into node-disjoint induced subgraphs* $G_1, G_2, \ldots, G_\ell$ *and weight functions* $\pi_i : V(G_i) \to \mathbb{R}_+$ *with the following properties. Let* $\mathcal{M}_i$ *be the induced pairs of* $\mathcal{M}$ *in* G_i *and let* X_i *be the endpoints of the pairs in* $\mathcal{M}_i$. *We have*

(a) $\pi_i(u) = \pi_i(v)$ *for each pair* $uv \in \mathcal{M}_i$, *and the support of* π_i *is contained in* X_i *(that is,* $\pi_i(u) > 0$ *only if* $u \in X_i$*),*

(b) X_i *is* π_i*-node-well-linked in* G_i,

(c) $\sum_{i=1}^{\ell} \pi_i(X_i) = \Omega(\mathrm{OPT}/(\gamma \log \mathrm{OPT})) = \Omega(\mathrm{OPT}/\log^2 k)$.

Moreover, such a partition is computable in polynomial time if there is a polynomial time algorithm for computing a node separator with sparsity at most $\gamma(G)$ *times the maximum concurrent flow.*

THEOREM 2.2 (see [7]). *Let* X *be a* π*-node-well-linked set in* G *and let* $\mathcal{M}$ *be a perfect matching on* X *such that* $\pi(u) = \pi(v)$ *for each pair* $uv \in \mathcal{M}$. *There is a matching* $\mathcal{M}' \subseteq \mathcal{M}$ *on a set* $X' \subseteq X$ *such that* X' *is* $1/32$*-node-well-linked in* G *and* $|\mathcal{M}'| = 2|X'| = \Omega(\pi(X))$. *Moreover, there is a polynomial-time algorithm that given* X *and* $\mathcal{M}$ *constructs* X' *and* $\mathcal{M}'$.

2.5. The cut-matching game. Another key tool that we use is the cut-matching game of KRV [32]; Louis [36] provides an extension of the cut-matching game to directed graphs.

We define the directed cut expansion of a cut $(A, V \setminus A)$ in a graph $G = (V, E)$ to be $\Phi(A) := \frac{|\delta^{\mathrm{out}}(A)|}{\min\{|A|, |V \setminus A|\}}$. A directed multigraph $G = (V, E)$ is a directed α-edge-expander if $\Phi(A) \geq \alpha$ for each nontrivial cut $A \subseteq V$, $A \neq \emptyset, V$. A set M of edges is a directed matching on V if $|\delta_M^{\mathrm{out}}(v)| \cup |\delta_M^{\mathrm{in}}(v)| \leq 1$ for every vertex $v \in V$. We say that M is a directed matching from Y to Z if M is a directed matching and each edge of M is directed from a node in Y to a node in Z. A directed matching from Y to Z is perfect if each node of $Y \cup Z$ is incident to exactly one edge of M.

In the cut-matching game, there is a set V of nodes, where $|V|$ is even, and there are two players, the cut player and the matching player. The goal of the cut player is to construct a directed edge-expander in as few iterations as possible, whereas the goal of the matching player is to prevent the construction of the edge-expander for as long as possible. The two players start with a graph $\mathcal{X}$ with node set V and an empty edge set. The game then proceeds in iterations, each of which adds a set of edges to $\mathcal{X}$. In iteration j, the cut player chooses a partition (Y_j, Z_j) of V such that $|Y_j| = |Z_j|$ and the matching player chooses a directed perfect matching M_j that matches the nodes of Y_j to the nodes of Z_j. The edges of M_j are then added to $\mathcal{X}$. Louis, building on the work of [32], showed that there is a strategy for the cut player that guarantees that after $O(\log^2(|V|))$ iterations the graph $\mathcal{X}$ is a directed $1/2$-edge-expander. We note that, given the symmetry in the definition of the expansion, the cut-player in round j can give a partition (Y_j, Z_j) and also (Z_j, Y_j), and ask the matching player to provide two directed matchings. In fact Louis's cut-player has this property. This ensures that the graph is Eulerian after each round.

THEOREM 2.3 (cut-matching game [36, 32]). *There is a randomized algorithm for the cut player such that, no matter how the matching player plays, after* $\gamma_{\mathrm{cmg}}(|V|) :=$

$O(\log^2(|V|))$ *iterations, the graph* $\mathcal{X}$ *is an Eulerian directed* $1/2$*-edge-expander with high probability.*

2.6. Planar graphs. By Π we denote the Euclidean plane. A *planar* graph is a graph that can be drawn on a plane; a *plane* graph is a planar graph, given together with its (one, fixed) planar embedding which maps vertices of the graph to points in the plane and edges to curves that can only intersect at the vertices.

Curves. For a closed Jordan curve γ and a point $p \in \Pi \setminus \gamma$, by $\zeta_p(\gamma) \in \mathbb{Z}$ we denote the element of the fundamental group of $\Pi \setminus \{p\}$, where γ belongs (with the convention that a clockwise cycle around p is the $+1$ element).

If $\zeta_p(\gamma) \neq 0$, then we say that γ *strictly encloses* a point p; note that this definition is consistent with the usual understanding of enclosure if γ is without self-intersections. We say that γ *encloses* a point $p \in \Pi$ if $p \in \gamma$ or γ strictly encloses p.

This notion naturally generalizes to (strict) enclosure of vertices, edges, and faces; in every such case, we require that every point of an edge or a face is (strictly) enclosed by a curve. Furthermore, we treat a face as an open subset of the plane, without its boundary vertices and edges. Also, given a closed walk or a cycle in a graph, when we say that the walk or a cycle (strictly) encloses some object, we mean that the closed curve along the walk or cycle (strictly) encloses it. Furthermore, for a closed curve γ without self-intersections, we say that a point $p \in \Pi$ is *to the left/right* of γ if $p \notin \gamma$ and p belongs to the connected part of $\Pi \setminus \gamma$ that is to the left or right of γ, respectively.

A sequence $C_1, C_2, \ldots, C_r$ of cycles in a directed or undirected plane graph is called *concentric* if they are pairwise vertex-disjoint and C_i encloses C_j if and only if $i \leq j$.

For some reasonings about cuts in planar graphs, it is helpful to look at the dual of the graph. For us, it is most convenient to formalize such reasonings using Jordan curves. A Jordan curve γ is in *general position* with respect to the plane graph G if it has finite number of intersections with G, its starting point and ending point do not belong to G, and whenever a point p lies both on γ and in the interior of an edge $e \in E(G)$, then γ traverses the edge e at this point. (That is, in a small neighborhood of p, the edge e splits the neighborhood into two parts, where one part contains points on γ immediately before p, and the second part contains points on γ immediately after p.) *A face-edge curve* in a plane digraph G is a Jordan curve in general position that does not traverse any vertex of G.

For a curve γ in general position with respect to G, we introduce the following notions. Assume γ intersects an edge e while going from a face f to a face f'. If e has the face f on the right and the face f' on the left, then we say that e *crosses* γ *from left to right* and, otherwise, if e has the face f on the left and the face f' on the right, then we say that e *crosses* γ *from right to left*. By $\mathrm{cross}^{L\to R}(\gamma)$ and $\mathrm{cross}^{R\to L}(\gamma)$ we denote the number of times an edge crosses γ from left to right and from right to left, respectively; note that in these numbers we may count one edge multiple times, one for each moment γ crosses the edge.

Balance. For a vertex v in a digraph G, the *imbalance* of v is the number $\mathrm{imb}_G(v) := |\delta_G^{\mathrm{in}}(v)| - |\delta_G^{\mathrm{out}}(v)|$, i.e., the difference between the in-degree and out-degree of v in G. A graph is *balanced* if $\mathrm{imb}_G(v) = 0$ for every $v \in V(G)$, and *Eulerian* if it is additionally weakly connected. (A directed graph is weakly connected if its underlying undirected graph is connected.) Furthermore, let the *imbalance* of a curve γ in a general position with respect to G be $\mathrm{imb}(\gamma) = \mathrm{cross}^{L\to R}(\gamma) - \mathrm{cross}^{R\to L}(\gamma)$.

The following lemma can be proved in a standard way by induction: the lemma is immediate for a curve γ enclosing once a single vertex v (i.e., with $\zeta_v(\gamma) = \pm 1$), while the statement supports an inductive step following the structure of the fundamental group of the plane punctured at the vertices of $V(G)$.

LEMMA 2.4. *Let γ be a closed face-edge curve in a plane digraph G. Then*

$$\mathrm{imb}(\gamma) = \sum_{v \in V(G)} \zeta_v(\gamma) \cdot \mathrm{imb}_G(v). \tag{2.1}$$

Proof. First, note that we can assume that γ has a finite number of intersections; if this is not the case, then we can slightly perturb γ without changing any of the quantities in (2.1).

We prove the statement by induction on the number of self-intersections, the number of enclosed vertices, and the number of arc crossings, lexicographically.

For the base cases, note that the statement is straightforward for curves completely contained in a face of G, curves without self-intersections that intersect only one arc twice in opposite directions, and curves γ that enclose a single vertex v such that $\zeta_v(\gamma) = \pm 1$ and γ intersects every nonloop arc incident to v exactly once.

Let now γ be as in the lemma statement and assume that γ does not fall under any of the base cases.

If γ has a self-intersection, we can split γ at a self-intersection point into two closed curves γ_1 and γ_2. Then, both γ_1 and γ_2 have strictly less self-intersections than γ, and (2.1) holds for them. If we add these two equations for γ_1 and γ_2, we obtain (2.1) for γ. Thus, henceforth we assume that γ is without self-intersections.

If γ visits the same face twice, let γ' be a simple curve connecting two points in two subsequent visits on the face in question that does not intersect neither G nor γ except for the endpoints. Let γ_1 and γ_2 be the two parts of γ, separated by the endpoints of γ'. For $i = 1, 2$, let γ_i' be a closed curve formed by γ_i and possibly reversed curve γ'. Then, note that each curve γ_i' is without self-intersections, encloses a subset of the vertices enclosed by γ, and has strictly less intersections with G than γ. Consequently, the statement (2.1) holds for γ_i'. If we add up these equations for γ_1' and γ_2', the contribution of γ' cancels (it is reversed in exactly one of the curves γ_i'), and we obtain (2.1) for γ.

Otherwise, note that by the assumption that γ does not fall under any of the base cases and visits every face at most once, γ intersects at least one arc, and intersects every arc at most once. We pick an arc e intersected by γ. By the assumed properties of γ, exactly one endpoint of e is enclosed by γ; let it be v. Construct a curve γ' that starts at a point of γ on one side of the intersection of γ and e, goes around the vertex v sufficiently close to v such that it intersects every arc incident to v except for e exactly once, and then meets γ on the other side of the intersection of γ and e.

We have again a curve γ' that is enclosed by γ that does not intersect γ except for the endpoints. Let γ_i and γ_i' be defined as previously for $i = 1, 2$, and without loss of generality assume that γ_1 is the "short" part of γ that crosses only e. Now, γ_1' falls under the base case of the induction (it encloses only v), while γ_2' encloses strictly less vertices than γ, as it no longer encloses v. Consequently, we can apply the inductive hypothesis for both γ_1' and γ_2', and derive (2.1) for γ as previously. This finishes the proof of the lemma. ∎

COROLLARY 2.5. *Every closed face-edge curve in a plane balanced digraph has zero imbalance.*

Flow/cut duality. We need the following flow/cut duality.

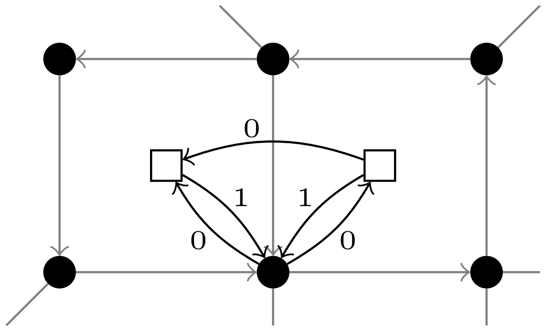

FIG. 3. *An illustration of the auxiliary graph in the proof of Lemma* 2.6. *The black dots are vertices of the primal graph, the white squares correspond to faces of the primal graph. The gray arrows are the edges of the primal graph, and the black arrows are the edges of the auxiliary graph.*

LEMMA 2.6. *Given a plane digraph G, two distinguished faces f^{in} and f^{out}, and an integer k, one can in linear time find either*

(a) *a family of directed vertex-disjoint cycles $C_1, C_2, \ldots, C_k$, all having f^{in} to the right and f^{out} to the left, or*

(b) *a curve γ in general position with respect to G, that starts in f^{in}, ends in f^{out}, intersects at most k vertices, and satisfies* $\mathrm{cross}^{L\to R}(\gamma) = 0$.

Proof. Define the following auxiliary directed plane digraph H (see Figure 3). The vertex set of H consists of the set of faces and vertices of G. For every edge e of G, we construct an edge of weight 0 pointing from the face to the left of e to the face to the right of e. Furthermore, for every vertex v of G and every face f incident to v, we construct an edge (v, f) of weight 0 and edge (f, v) of weight 1. We construct the graph H and perform a single-source shortest path search from f^{in}; since the weights are 0 and 1, these steps can be done in linear time by contracting the zero-weight edges and running breadth first search. When we speak later in this proof about distances between faces or vertices of G, we mean distances in the auxiliary graph H.

Let d be the distance from f^{in} to f^{out}. We will show that if $d < k$, then we can find the desired curve γ, and if $d \geq k$, we can find the desired k cycles. Observe that if a face f is within distance exactly δ from f^{in}, then it is easy to construct a curve γ in general position with respect to G that starts in f^{in}, ends in f, intersects exactly δ vertices, and satisfies $\mathrm{cross}^{L\to R}(\gamma) = 0$. Indeed, consider a simple path P in H of weight δ that starts in f^{in} and ends in f. Let $f^{\mathrm{in}} = f_0, f_1, \ldots, f_r = f$ be consecutive faces visited by P. We construct a curve γ that visits the same faces in the same order as follows. Fix an index $0 \leq i < r$. By the construction of H, the path P goes from f_i to f_{i+1} either via a direct edge of weight 0 or via a vertex v_i incident both to f_i and f_{i+1} in G. In the former case, the definition of H implies that there is an arc e_i in G that has f_i to the left and f_{i+1} to the right. We let γ cross e_i to go from f_i to f_{i+1}; note that this keeps $\mathrm{cross}^{L\to R}(\gamma) = 0$. In the latter case, the definition of H implies that P goes along arcs (f_i, v_i) and (v_i, f_{i+1}) of H, out of which one arc is of weight 1 and one arc is of weight 0. We let γ traverse v_i to go from f_i to f_{i+1}; since the weight of P is δ, the curve γ traverses exactly δ vertices. Finally, since P is simple, γ is without self-intersections. Consequently, if $d < k$, we obtain a curve γ as in case (2) of the lemma statement.

Therefore we may assume that $d \geq k$. We show that one can construct a family of d vertex-disjoint cycles, all having f^{in} to the right and f^{out} to the left. To this end, we show that for every $1 \leq i \leq d$, one can construct one such cycle using only vertices within distance exactly i from f^{in}. Fix such an index i. Let V_i be the set of vertices of G within distance exactly i from f^{in} and let G_i be the subgraph of G with $V(G_i) = V_i$ and $e \in E(G_i)$ if and only if one of the faces incident to e has distance (from f^{in}) less than i and the other has distance at least i.

Consider an edge $e \in E(G_i)$ that has on one side a face $f_{\geq}$ within distance at least i from f^{in}, and on one side a face $f_{<}$ within distance less than i. We make the following observations.

Since H has a zero weight edge from the left of e to the right of e, $f_{\geq}$ is on the left of e and $f_{<}$ is on the right of e.

Let v be the head of e. Since we can reach $f_{<}$ from $f_{\geq}$ in H using the edges $f_{<} \to v$ of cost 1 and $v \to f_{\geq}$ of cost 0, it follows that the distance from f^{in} to $f_{<}$ is equal to $i-1$ and the distance from f^{in} to $f_{\geq}$ is equal to i. Additionally, the distance from f^{in} to v is equal to i: the distance is at least i due to the cost-0 edge $(v, f_{\geq})$, and it is at most i due to the cost-1 edge $(f_{<}, v)$. Therefore $v \in V_i$.

Let f be the face of G_i that contains f^{out}. We claim that every face of G that is contained in f is within distance at least i from f^{in}. Let f' be any face of distance less than i, and let P be any undirected path in the dual of G from f^{out} to f'. Since f^{out} has distance at least i, and f' has distance less than i, there exists an arc e^* on P whose primal copy e belongs to G_i. Since the choice of P is arbitrary, f' does not lie in the same face of G_i as f^{out}. Thus, in particular, f^{in} is not contained in f. Furthermore, every edge e incident to the face f has f to its left (since $e \in E(G_i)$ and f contains faces of G that are within distance at least i from f^{in}), and the face of G to the right of e is within distance less than i from f^{in}.

Moreover, observe that if a vertex $v \in V(G)$ is within distance at most j from f^{in} for some j, then so is every face incident to v. Consequently, f is either isomorphic to an open disc, or is isomorphic to a complement of a closed disc (i.e., if we treat the embedding of G as an embedding on a sphere, f is always isomorphic to an open disc). Let C be the (undirected) cycle around f in the graph G_i. Since every face of G that is contained in f is within distance at least i from f^{in}, by the definition of G_i we have that C is in fact a directed cycle that has f on its left and f^{in} on its right. This finishes the proof of the lemma. ∎

2.7. A lemma on Eulerian digraphs. We conclude with the following lemma that encapsulates the main property of Eulerian digraphs that make them similar to undirected graphs. The lemma has been used previously, e.g., in [26]; we include its proof for the sake of completeness.

LEMMA 2.7. *Let G be an Eulerian digraph of maximum in-degree Δ, let G^{UN} be the undirected graph underlying G, let $A, B \subseteq V(G)$, and let ℓ be an integer. If there exist $(\Delta+1)\ell+1$ (undirected) vertex-disjoint paths from A to B in G^{UN}, then there exist $\ell+1$ directed ones in G as well.*

Proof. Assume there do not exist $\ell+1$ directed vertex-disjoint paths from A to B in G. By Menger's theorem, there exist sets $A', B' \subseteq V(G)$ with $A \subseteq A'$, $B \subseteq B'$, $A' \cup B' = V(G)$, $|A' \cap B'| \leq \ell$, and no arc of G has its tail in $A' \setminus B'$ and its head in $B' \setminus A'$. Let d be the number of arcs with a tail in $B' \setminus A'$ and a head in $A' \setminus B'$. Observe that if there exist $(\Delta+1)\ell+1$ undirected vertex-disjoint paths from A to B in G^{UN}, then $d \geq \Delta\ell+1$, as only ℓ of these paths can go through $|A' \cap B'|$.

Thus, there are at least $\Delta\ell+1$ arcs going from B' to $A' \setminus B'$. However, since the maximum in-degree of G is Δ, there are at most $\Delta|A' \cap B'| \leq \Delta\ell$ arcs with tail in $A' \setminus B'$ and head in B'. This is in contradiction with the assumption that G is Eulerian, as in an Eulerian digraph the number of arcs from $A' \setminus B'$ to B' and from B' to $A' \setminus B'$ should be the same. ◻

3. Reduction to Eulerian graphs with small degree. In this section, we show that Sym-Dir-NDP in directed planar graphs can be reduced to Sym-Dir-NDP in directed planar multigraphs that are Eulerian and of maximum degree $O(\log^2 k)$, where k is the number of pairs in the Sym-Dir-NDP instance.

The reduction combines the well-linked decomposition framework of [7, 10] and the cut-matching game of [36, 32] given in the preliminaries section.

Let $(G, \mathcal{M})$ be an instance of Sym-Dir-NDP on a directed planar graph $G = (V, E)$. First, we solve the LP relaxation (Sym-Dir-NDP LP) and obtain a fractional solution $(f, \mathbf{x})$. Using Theorem 2.1 together with the algorithm of [35] for computing a sparse node separator, we construct in polynomial time a decomposition $(G_1, \pi_1), \ldots, (G_\ell, \pi_\ell)$. Using Theorem 2.2 on each instance (G_i, π_i), we construct in polynomial time an instance $(G_i, \mathcal{M}'_i)$ in which the terminals of $\mathcal{M}'_i$ are $\Omega(1)$-node-well-linked. In the following, we consider each of these subinstances separately and, for each instance $(G_i, \mathcal{M}'_i)$, we construct an instance $(G'_i, \mathcal{M}''_i)$ such that G'_i is an Eulerian multigraph on the same vertex set as G_i and $\mathcal{M}''_i \subseteq \mathcal{M}'_i$.

Consider an instance $(G, \mathcal{M})$ in which the terminals of $\mathcal{M}$ are $\Omega(1)$-node-well-linked. Let $k = |\mathcal{M}|$. We use the cut-matching game to construct a directed edge-expander $\mathcal{X} = (\mathcal{T}, F)$, where $\mathcal{T}$ is the set of terminals of $\mathcal{M}$, and an embedding of $\mathcal{X}$ into G. For the cut player, we use the strategy guaranteed by Theorem 2.3. We implement the matching player as follows. Consider an iteration j and let (Y_j, Z_j) be the partition of $\mathcal{T}$ chosen by the cut player. Since $\mathcal{T}$ is $\Omega(1)$-node-well-linked in G, there is a collection of paths $\mathcal{P}_j$ from Y_j to Z_j and a collection of paths $\mathcal{Q}_j$ from Z_j to Y_j such that each node of G appears in $O(1)$ of the paths in $\mathcal{P}_j \cup \mathcal{Q}_j$. The paths $\mathcal{P}_j$ define a directed matching M_j between Y_j and Z_j, where an edge $(u, v) \in M_j$ corresponds to a path in $\mathcal{P}_j$ from u to v. Similarly $\mathcal{Q}_j$ defines a directed matching M'_j between Z_j and Y_j. Assuming that the cut-player gives both (Y_j, Z_j) and (Z_j, Y_j) in round j we can return M_j and M'_j as the response of the matching player. In the general setting where the cut-player may only give the partition (Y_j, Z_j), we can let M_j be the response of the matching player but add the paths in $\mathcal{P}_j \cup \mathcal{Q}_j$ even though $\mathcal{Q}_j$ is not immediately relevant. The advantage of adding both sets of paths is that we maintain Eulerianness of the graph.

We run the cut-matching game with the above strategies for the cut and matching player for $\gamma_{\mathrm{cmg}}(k) = O(\log^2 k)$ rounds, where $k = |\mathcal{M}|$, in order to obtain a graph $\mathcal{X} = (\mathcal{T}, F)$. The paths $\{\mathcal{P}_j \cup \mathcal{Q}_j : 1 \le j \le \gamma_{\mathrm{cmg}}(k)\}$ give us an embedding of $\mathcal{X}$ into G with $O(\log^2 k)$ node congestion that maps each vertex of $\mathcal{X}$ to its corresponding node in G and it maps each edge of $\mathcal{X}$ to a path in G.

Let H be the multigraph on the same vertex set as G obtained by taking the union of the paths $\{\mathcal{P}_j \cup \mathcal{Q}_j : 1 \le j \le \gamma_{\mathrm{cmg}}(k)\}$; if an edge appears in more than one path, we make multiple copies of the edge.

LEMMA 3.1. *The multigraph H is Eulerian and it has maximum degree $O(\gamma_{\mathrm{cmg}}(k)) = O(\log^2 k)$. Moreover, with high probability, $\mathcal{T}$ is $\Omega(1/\log^4 k)$-node-well-linked in H.*

Proof. The first two properties are immediate from the construction of $\mathcal{X}$. Now suppose that $\mathcal{X}$ is a directed 1/2-edge-expander. (Recall that this event holds with high probability.) Note that this implies that $\mathcal{T}$ is 1/2-edge-well-linked in $\mathcal{X}$. Since $\mathcal{X}$

is embedded into H with $O(\log^2 k)$ node (and thus edge) congestion, it follows that $\mathcal{T}$ is $\Omega(1/\log^2 k)$-edge-well-linked in H. Since the maximum degree in H is $O(\log^2 k)$, $\mathcal{T}$ is $\Omega(1/\log^4 k)$-node-well-linked in H. ∎

We use Theorem 2.2 on $(H, \mathcal{M})$ to obtain a subset $\mathcal{M}' \subseteq \mathcal{M}$ of size $\Omega(k/\log^4 k)$ such that the terminals $\mathcal{T}'$ participating in $\mathcal{M}'$ are $\Omega(1)$-node-well-linked in H. Since we have only duplicated edges of G when constructing H but not the nodes, a routing in H with node congestion c translates to a routing in G with node congestion c. Therefore we have the following reduction theorem to Eulerian instances of bounded degree.

THEOREM 3.2. *There is a polynomial time randomized algorithm that takes as input an instance $(G, \mathcal{M})$ of* Sym-Dir-NDP *in which the vertices of $\mathcal{M}$ are $\Omega(1)$-node-well-linked and constructs an instance $(H, \mathcal{M}')$ such that the latter has the following properties with high probability:*

- *H is an Eulerian multigraph of maximum degree $O(\log^2 k)$, where $k = |\mathcal{M}|$, with $V(H) \subseteq V(G)$ and $E(H)$ consists of copies of edges of G.*
- *$\mathcal{M}'$ is a subset of $\mathcal{M}$ of size $\Omega(k/\log^4 k)$ and the vertices of $\mathcal{M}'$ are $\Omega(1)$-node-well-linked in H.*

By combining the preceding theorem with the reduction to well-linked instances (Theorem 2.1), we obtain Theorem 1.2. Note that we lose an $O(\log^2 k)$ factor in the approximation from the well-linked decomposition and another $O(\log^4 k)$ factor from Theorem 3.2.

4. The crossbar construction for Eulerian graphs. The goal of this section is to prove Theorem 1.3: in a planar directed Eulerian graph with maximum in-degree Δ and an α-node-well-linked set X one can find a routing structure (crossbar) of size $\Omega(\alpha^2|X|/\Delta^2)$.

We first remark that the two outcomes of Theorem 1.3 are the same if one considers embeddings on the sphere, while on the plane they differ only by the choice of the outer face of the embedding. Furthermore, note that all paths in $\mathcal{P}^+$ and in $\mathcal{P}^-$ intersect every constructed concentric cycle due to planarity.

In the proof of Theorem 1.3, without loss of generality we assume that every vertex of X is incident to one outgoing arc and one incoming arc, and these two arcs have the same second endpoint: We can achieve it by creating a pendant vertex x' for every $x \in X$, connected to x with arcs (x, x') and (x', x); note that the well-linkedness of X may drop to $\alpha/(\alpha + 1)$ in this manner. Consequently, every cycle and path has no vertices in X (except for possibly some endpoints); henceforth we will implicitly use this property multiple times.

Obtaining an undirected grid. We start by applying the construction for *undirected* planar graphs from [10]. Let G^{UN} be the underlying undirected graph of G. Clearly, X is α-node-well-linked in G^{UN} and thus G^{UN} has (undirected) treewidth $\Omega(\alpha|X|)$. Hence we can obtain a large grid minor linked to the terminals X using the following theorem of [10]. In what follows, it is notationally more convenient to work with *subdivided walls as subgraphs*, instead of minors. A $t \times t$ wall and a subdivided wall are shown in Figure 4.

Formally, for an odd integer t, a $t \times t$ wall is defined as follows. Its vertex set consists of $2t^2 - 2$ vertices $v_{i,j}$ for $1 \leq i \leq t$, $1 \leq j \leq 2t$, $(i, j) \notin \{(1, 2t), (t, 1)\}$. For fixed $1 \leq i \leq t$, the vertices $v_{i,j}$ are arranged in the increasing order of indices j into a path, called the ith row of the wall. Furthermore, for every vertex $v_{i,j}$ with $i \neq t$ and $i + j$ even there is an edge $v_{i,j}v_{i+1,j}$.

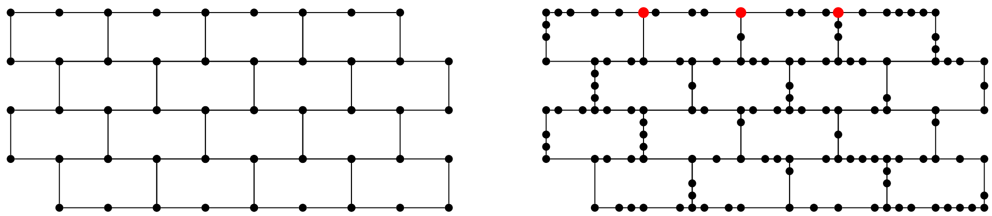

FIG. 4. *A wall (left) and a subdivided wall (right). The red vertices denote the interface of the wall.*

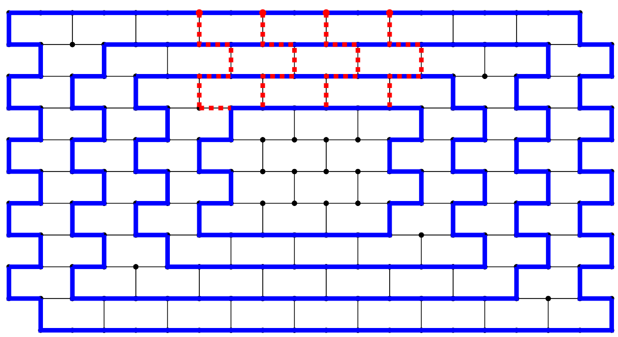

FIG. 5. *Illustration of Corollary* 4.2. *The cycles are blue, while the paths are dashed red.*

The $(t-2)$ vertices of degree three in the top row of a $t \times t$ wall Γ are called the *interface* of the wall, denoted I_Γ.

THEOREM 4.1 (see [10, Theorem 4.5]). *There is a polynomial-time algorithm that given an undirected planar graph H, $\alpha \in (0,1]$, and an α-node-well-linked set X in H, finds an integer $t = \Omega(\alpha|X|)$, a subdivided $t \times t$ wall Γ in H, and a family of t node-disjoint paths connecting X and the interface of Γ.*

In our construction we do not need the entire structure of a subdivided wall, but only part of it, as in the following immediate corollary (see Figure 5).

COROLLARY 4.2. *One can in polynomial time find an integer $r = \Omega(\alpha|X|)$ and a sequence of node-disjoint concentric undirected cycles $C_1, C_2, \ldots, C_r$ in G^{UN} with C_1 being the outermost and C_r being the innermost cycle, with the additional property that for every $1 \le i \le r$ there exists r vertex-disjoint paths in G^{UN} from X to $V(C_i)$.*

Isles S^{out} and S^{in}. Let us fix a choice of r and cycles $C_1, C_2, \ldots, C_r$ stemming from Corollary 4.2. For a while, we work only with the undirected graph G^{UN}. Our goal is to strengthen the requirement of the existence of many undirected paths between X and the innermost and outermost cycles by getting more properties about their endpoints, so that we can use an argument similar to the one of [26] to reason about the existence of *directed* concentric cycles with similar connectivity toward X.

To this end, we identify two small connected parts of G^{UN}, S^{out}, and S^{in}, one around C_1 and one around C_r. The parts will be large enough so that there is a substantial number of vertex-disjoint directed paths between them and X, but small enough so that they are placed very locally in the graph, and their boundary is small. This last property ensures that after deletion of these parts, the graph is close to Eulerian, and we can make use of Lemma 2.4.

For a vertex set $Q \subseteq V(G^{\mathrm{UN}})$, a vertex $v \notin Q$, and an integer $\ell \geq 2\Delta$, we say that a vertex set S is a (v, Q, ℓ)-*isle* if $v \in S$, $G^{\mathrm{UN}}[S]$ is connected, $S \cap Q = \emptyset$, and $|N_{G^{\mathrm{UN}}}(S)| \leq \ell$.[4] We will rely on the following greedy procedure, that is inspired by the enumeration algorithm for important separators in parameterized complexity (cf. [13] and [20, Chapter 8]).

LEMMA 4.3. *Given a set $Q \subseteq V(G^{\mathrm{UN}})$, a vertex $v \notin Q$, and an integer $\ell \geq 2\Delta$, one can in $O(\ell^3 n)$ time find an inclusionwise maximal (v, Q, ℓ)-isle.*

Proof. We perform the following iterative procedure. Start with $S = \{v\}$; clearly, S is a (v, Q, ℓ)-isle, as $v \notin Q$ by assumption and the maximum in-degree of G is Δ. In an iterative step, we assume that S is a (v, Q, ℓ)-isle, and our goal is to check if S is an inclusionwise maximal one, or produce a (v, Q, ℓ)-isle S' with $S \subsetneq S'$.

To this end, consider every $w \in N_{G^{\mathrm{UN}}}(S) \setminus Q$; note that, by the connectivity of S' and S, there exists such w that is contained in $S' \setminus S$ for every isle S' we are looking for. Collapse in G^{UN} the set $S \cup \{w\}$ into a single vertex s and add a supersource vertex t adjacent to all vertices of Q. Let G' be the resulting (undirected) graph. Find a minimum $s-t$ vertex cut Z in G' of size at most ℓ, or conclude that such a minimum cut is of size larger than ℓ; using the Ford–Fulkerson algorithm, we can do so after $\mathcal{O}(\ell)$ augmentations, using a total time of $\mathcal{O}(\ell n)$. Moreover, within this time we can find the minimum cut *closest to* t, that is, the unique one with an inclusionwise maximal set of vertices remaining in the connected component with the vertex s (cf. [20]).

If such a cut is found, let S' be the subset of vertices of G corresponding to the connected component of $G' \setminus Z$ containing the vertex s. Clearly, $N_{G^{\mathrm{UN}}}(S') = Z$, and S' is a (v, Q, ℓ)-isle containing S and w. Otherwise, we conclude that no (v, Q, ℓ)-isle containing both S and w exists, since for every such isle S', the set $N_{G^{\mathrm{UN}}}(S')$ is an $s - t$ cut in G' of size at most ℓ.

The computation for fixed S and w takes $\mathcal{O}(\ell n)$ time. Since S is an (v, Q, ℓ)-isle, there are at most ℓ vertices w to try. Due to the fact that we always take the $s - t$ cut closest to t, the size of the set $N_{G^{\mathrm{UN}}}(S)$ strictly grows at every iteration (possibly except the first one, when $S = \{v\}$). Consequently, there are at most $\ell + 1$ iterations of the procedure, and the running time bound follows. ∎

We pick an arbitrary vertex v^{out} on C_1 and an arbitrary vertex v^{in} on C_r and use Lemma 4.3 for each of these vertices, the set $Q := X$, and threshold $\ell := \lfloor r/(4\Delta+2) \rfloor$; recall that $|X| = \Omega(\Delta^2/\alpha)$ by the assumptions of Theorem 1.3, and thus we may assume $\ell \geq 2\Delta$. Let S^{out} and S^{in} be the two isles obtained. Since $\ell < r$, and every cycle C_i is connected with r vertex-disjoint paths to X, no cycle C_i is contained in either S^{out} or S^{in}. Since an isle is connected, we obtain the following.

LEMMA 4.4. *The isle S^{out} does not contain any vertex that is enclosed by $C_{\ell+1}$, and the isle S^{in} does not contain any vertex that is not strictly enclosed by $C_{r-\ell}$.*

Proof. The proofs for S^{in} and S^{out} are symmetrical, so we just focus here on the case of S^{out}. Assume to the contrary that S^{out} contains a vertex enclosed by $C_{\ell+1}$. Since $v^{\mathrm{out}} \in S^{\mathrm{out}}$ and by the connectity of S^{out}, S^{out} contains a vertex from every cycle C_i, $1 \leq i \leq \ell + 1$. Since $|N_{G^{\mathrm{UN}}}(S^{\mathrm{out}})| \leq \ell$, for some $1 \leq i \leq \ell + 1$ we have that $V(C_i)$ is completely contained in S^{out}. However, recall that there are $r > \ell$ vertex-disjoint paths in G^{UN} connecting C_i with X. This contradicts the facts that $S^{\mathrm{out}} \cap X = \emptyset$ and $|N_{G^{\mathrm{UN}}}(S^{\mathrm{out}})| \leq \ell$. ∎

[4]We use the following notation with respect to neighborhoods. Let G be an undirected graph, $x \in V(G)$, and $S \subseteq V(G)$. Then $N_G(x)$ is the set of neighbors of x in G, $N_G[x] = \{x\} \cup N_G(x)$, $N_G[S] = \bigcup_{x \in S} N_G[x]$, $N_G(S) = N_G[S] \setminus S$, $N_G^2[S] = N_G[N_G[S]]$, and $N_G^2(S) = N_G(N_G[S])$.

By Lemma 4.4, the isles S^{out} and S^{in} are somewhat local in the graph: they do not go too deep into the set of cycles $C_1, C_2, \ldots, C_r$. On the other hand, recall that they are *inclusionwise maximal* isles; by the next lemma, this ensures that they are connected by a large number of vertex-disjoint undirected paths to the set X. Let $W^{\text{out}} = N^2_{G^{\text{UN}}}[S^{\text{out}}]$ and $W^{\text{in}} = N^2_{G^{\text{UN}}}[S^{\text{in}}]$.

LEMMA 4.5. *In G^{UN}, there are $\ell + 1$ node-disjoint undirected paths connecting W^{out} and X and $\ell + 1$ node-disjoint undirected paths connecting W^{in} and X.*

Proof. By symmetry, we can focus on the case of W^{out}. The intuition is as follows: if there do not exist sufficiently many node-disjoint paths, then the corresponding cut would allow us to construct a strictly larger isle, a contradiction to the maximality of S^{out}. In some sense, $N_{G^{\text{UN}}}(S^{\text{out}})$ is the "last bottleneck" of size at most ℓ between v^{out} and X, and, after passing it, we should have more than ℓ paths between X and $N^2_{G^{\text{UN}}}[S^{\text{out}}] = W^{\text{out}}$.

Formally, assume the contrary of the lemma statement; by Menger's theorem, there exist vertex sets $A, B \subseteq V(G^{\text{UN}})$ such that $A \cup B = V(G^{\text{UN}})$, $|A \cap B| \leq \ell$, $W^{\text{out}} \subseteq A$, $X \subseteq B$, and no edge of G^{UN} has one endpoint in $A \setminus B$ and the second endpoint in $B \setminus A$.

Recall that $S^{\text{out}} \cap X = \emptyset$ by the definition of an isle, while $N^2_{G^{\text{UN}}}[S^{\text{out}}] = W^{\text{out}} \subseteq A$. Hence we may assume that $(N_{G^{\text{UN}}}[S^{\text{out}}] \setminus X) \subseteq (A \setminus B)$, as removing all vertices of $N_{G^{\text{UN}}}[S^{\text{out}}] \setminus X$ from B would not invalidate any of the properties of the pair (A, B). Recall also that $G^{\text{UN}}[S^{\text{out}}]$ is connected; let S_A be the vertex set of the connected component of $G^{\text{UN}} \setminus (A \cap B)$ containing S^{out}. Clearly, $S_A \subseteq A \setminus B$, so $S_A \cap X = \emptyset$. Furthermore, $N_{G^{\text{UN}}}(S_A) \subseteq A \cap B$, so $|N_{G^{\text{UN}}}(S_A)| \leq \ell$. As $S^{\text{out}} \subseteq S_A$, by the maximality of S^{out}, we infer that $S_A = S^{\text{out}}$. Since $N_{G^{\text{UN}}}[S^{\text{out}}] \setminus X \subseteq S_A$, we infer that $N_{G^{\text{UN}}}(S^{\text{out}}) \subseteq X$. However, this is a contradiction, as G^{UN} is connected and $S^{\text{out}} \subsetneq V(G^{\text{UN}}) \setminus X$. ∎

Finding directed concentric cycles. We now use the undirected cycles $C_1, C_2, \ldots, C_r$ to find a large number of node-disjoint *directed* concentric cycles separating S^{in} and S^{out}. Recall that $\ell := \lfloor r/(4\Delta + 2) \rfloor$.

LEMMA 4.6. *One can in polynomial time find $\lceil \ell/2 \rceil$ node-disjoint directed concentric cycles, all going in the same direction (all clockwise or all counterclockwise), such that all vertices of S^{in} are strictly enclosed by the innermost cycle, and none of the vertices of S^{out} are enclosed by the outermost cycle, or vice versa, with the roles of S^{in} and S^{out} swapped.*

Proof. Denote $G' = G \setminus (N_{G^{\text{UN}}}[S^{\text{out}}] \cup N_{G^{\text{UN}}}[S^{\text{in}}])$. Let f^{out} and f^{in} be the faces of G' that contain S^{out} and S^{in}, respectively; by Lemma 4.4, the cycle $C_{\lceil r/2 \rceil}$ remains in G' and $f^{\text{out}} \neq f^{\text{in}}$. Furthermore, the vertices of $N^2_{G^{\text{UN}}}(S^{\text{out}})$ lie on the face f^{out} of G', and the vertices of $N^2_{G^{\text{UN}}}(S^{\text{in}})$ lie on the face f^{in}. We apply Lemma 2.6 twice to the graph G' and the requirement of ℓ cycles, once for the pair of faces $(f^{\text{out}}, f^{\text{in}})$ and once for the pair $(f^{\text{in}}, f^{\text{out}})$. If at least one of the applications returns a family of cycles, then we are done, as every cycle encloses either S^{in} or S^{out}. Thus, we are left with the case when both the applications return a curve; let us denote these curves γ_1 and γ_2, respectively.

Before we proceed to the formal calculations leading to a contradiction, let us give some intuition. The curves γ_1 and γ_2 are very skewed in terms of the directions of edges crossing it: only edges in one direction are allowed, while in the second direction only ℓ vertices are allowed, and every vertex is of maximum in-degree Δ. The locality of isles S^{out} and S^{in} (Lemma 4.4) implies that γ_1 and γ_2 cross most of the cycles C_i;

consequently, they need to cross much more than $\ell\Delta$ arcs in one direction. However, the graph G' is very close to an Eulerian one, as we have a bound of ℓ on the size of the boundary of S^{out} and S^{in}. This leads to a contradiction with Lemma 2.4 for a closed curve being essentially a concatenation of γ_1 and γ_2.

Formally, let us first modify the curve γ_1 to obtain a face-edge curve γ_1' as follows: whenever γ_1 crosses a vertex v, we move it slightly to avoid v, at the cost of intersecting some of the arcs incident to v. Since the maximum in-degree and out-degree of G (and thus G') is at most Δ, we have that $\text{cross}^{L\to R}(\gamma_1') \leq \Delta(\ell-1)$. Similarly, we obtain a curve γ_2' with $\text{cross}^{L\to R}(\gamma_2') \leq \Delta(\ell-1)$. Since γ_1' starts in f^{out} and ends in f^{in}, while γ_2' starts in f^{in} and ends in f^{out}, we can obtain a closed face-edge curve γ' by concatenating γ_1' and γ_2' so that the following holds:

- we do not introduce any new crossing with the embedding of G';
- for the arcs of $E(G)\setminus E(G')$, γ' intersects only arcs between $N_{G^{\text{UN}}}[S^{\text{out}}] \cup N_{G^{\text{UN}}}[S^{\text{in}}]$ and $V(G')$, and every such arc is intersected at most once.

This curve γ' visits both f^{out} and f^{in}. Let us estimate $\text{cross}^{L\to R}(\gamma')$ with respect to G using the observations above. We have a contribution of at most $2\Delta(\ell-1)$ from the intersections with G'. For the edges of $E(G)\setminus E(G')$, the bound on the maximum out- and in-degree, as well as the bound $|N_{G^{\text{UN}}}(S^{\text{out}})|, |N_{G^{\text{UN}}}(S^{\text{in}})| \leq \ell$, ensures that each of the connections between γ_1' and γ_2' in f^{out} and f^{in} contribute at most $\Delta\ell$ each to $\text{cross}^{L\to R}(\gamma')$. Together, we obtain

$$\text{cross}^{L\to R}(\gamma') \leq 2\Delta(\ell-1) + 2\Delta\ell. \tag{4.1}$$

By Lemma 4.4, the undirected cycles $C_{\ell+2}, C_{\ell+3}, \ldots, C_{r-\ell-1}$ remain in G', and both γ_1' and γ_2' need to cross at least one edge of each of these cycles, since γ_1 and γ_2 connect f^{in} and f^{out}. Consequently,

$$\text{cross}^{L\to R}(\gamma') + \text{cross}^{R\to L}(\gamma') \geq 2(r-2\ell-2). \tag{4.2}$$

By combining (4.1) and (4.2), and by the choice of ℓ, we obtain that

$$-\text{imb}(\gamma') = \text{cross}^{R\to L}(\gamma') - \text{cross}^{L\to R}(\gamma') \geq 2(r-2\ell-2-2\Delta(\ell-1)-2\Delta\ell) > 0. \tag{4.3}$$

This is a contradiction with Corollary 2.5, as G is balanced. ∎

Finishing the crossbar construction. Recall that $W^{\text{out}} = N^2_{G^{\text{UN}}}[S^{\text{out}}]$ and $W^{\text{in}} = N^2_{G^{\text{UN}}}[S^{\text{in}}]$. Let $C_1', \ldots, C_{\lceil \ell/2\rceil}'$ be the concentric directed cycles found by Lemma 4.6; by symmetry, assume they all enclose S^{in}. Let $q := \lceil \ell/4 \rceil$. We consider two cases, depending on whether at least half of the vertices of X are enclosed by C_q' or not. The two cases correspond to the two symmetric outcomes of Theorem 1.3. In what follows, we describe only the first case, when at least half of the vertices of X are enclosed by C_q', and we use cycles $C_1', C_2', \ldots, C_q'$ and vertex-disjoint paths from X to W^{out}; the second case is completely symmetric but uses cycles $C_{q+1}', C_{q+2}', \ldots, C_\ell'$ and paths from X to W^{in}.

Consider the set of $\ell+1$ paths in G^{UN} connecting W^{out} and X, whose existence is guaranteed by Lemma 4.5, and let X^{out} be the set of the endpoints of the paths. The vertices in X^{out} may not be enclosed by C_q'. Our goal is to find a different set of vertices that are enclosed by C_q' such that they have disjoint paths to W^{out}; we use well-linkedness of X for this purpose. As $\ell+1 \leq |X|/2$ and the set X is α node-well-linked, for every set $X' \subseteq X$ of $\ell+1$ vertices enclosed by C_q', there exist $\alpha(\ell+1)$ node-disjoint paths connecting X^{out} and X'. By combining these paths with the paths connecting W^{out} and X^{out}, we obtain a flow that sends $\alpha(\ell+1)/2$ amount of flow in G^{UN} with unit node capacities from X' to W^{out}. By the integrality of flow, there is an integral of the same value. Therefore there exists a set $Y \subseteq X$ of size at least $\alpha(\ell+1)/2$, whose vertices are all enclosed by C'_q, and such that there exist $|Y|$ node-disjoint paths in G^{UN} connecting Y and W^{out}.

By Lemma 2.7, there exist $(\alpha\ell-2)/(2(\Delta+1))$ node-disjoint directed paths from Y to W^{out} (we let $Y^+ \subseteq Y$ denote the end points of these paths) and the same amount of node-disjoint directed paths from W^{out} to Y (we let $Y^- \subseteq Y$ denote the end points of these paths). Recall that $\ell = \Theta(\alpha|X|/\Delta)$, and thus $(\alpha\ell-2)/(2(\Delta+1)) = \Theta(\alpha^2|X|/\Delta^2)$. As no vertex of W^{out} is strictly enclosed by C'_1, these paths, together with the cycles $C'_1, C'_2, \dots, C'_q$, form the desired structure. This concludes the proof of Theorem 1.3.

5. Routing using the crossbar. In this section, we show how to use the crossbar guaranteed by Theorem 1.3 in order to route a large subset of the pairs and prove Theorem 1.4.

We first give an informal sketch of the argument and then provide the formal proof. We start with a basic property of the crossbar structure.

LEMMA 5.1. *Let $A \subseteq Y^+$ and $B \subseteq Y^-$ such that $|A| = |B|$ and let M be any directed matching from A to B. The pairs in M can be routed in G with node-congestion* 3.

Proof. We can assume without loss of generality that $A = Y^+$ and $B = Y^-$. Let $h = |A| = |B|$ and let the pairs of the matching be $(s_1,t_1),\dots,(s_h,t_h)$. For each i let $P_i^+ \in \mathcal{P}^+$ be the the path from s_i to the innermost cycle and let $P_i^- \in \mathcal{P}^-$ be the path from the innermost cycle to t_i. Consider the graph induced by $E(P_i^+) \cup E(C_i) \cup E(P_i^-)$, where C_i is the ith cycle in the crossbar; this graph contains a path from s_i to t_i since both P_i^+ and P_i^- intersect C_i. This is our routing for the pair (s_i,t_i). Since each of the path collections $\mathcal{P}^+$, $\mathcal{P}^-$ and the concentric cycles of the crossbar are separately node-disjoint we see that the routing we prescribed for the pairs causes a node congestion of at most 3. ∎

The preceding lemma is at the heart of the routing but we still need to take care of a few things. We can assume via a simple argument that there are at least $|Y^+|/2$ pairs from our initial pairs $\mathcal{M}$ whose end points are in Y^+. Let $\mathcal{M}_1 \subset \mathcal{M}$ be these pairs and we will assume for ease of notation that these pairs are $s_1t_1,\dots,s_ht_h$ and that Y^+ contains $s_1,s_2,\dots,s_h$. (This is is just a relabeling.) There are two main issues. The mate of s_i—namely, t_i—may not be in Y^-. Second, even if t_i is in Y^-, the preceding lemma only helps in routing s_i to t_i, and we still need to route t_i to s_i. These two issues can be overcome via the well-linkedness of X. Using the α-well-linkedness of X we can find disjoint paths from a subset of $\{t_1,t_2,\dots,t_h\}$ to Y^-; note that we have no control over where these paths end up in Y^- and also which of the nodes in $\{t_1,t_2,\dots,t_h\}$ are connected. In this process, we may retain only an α-fraction of the pairs from $\mathcal{M}_1$, since X is α-well-linked. After this we have the sources in Y^+ and sinks in Y^- and we can use the preceding lemma to find paths from the sources to the sinks with node congestion 3. To find a routing from the sinks to the sources we can again use well-linkedness of X to route the sources to Y^- and the sinks to Y^+ and then apply the preceding lemma again. Each invocation of well-linkedness results in a loss of an α-factor in the number of pairs and an additive 1 in the congestion bound. This basic scheme gives a routing with congestion 9 and routes $\Omega(\alpha^3|Y^+|)$ pairs.

Below we build upon the high-level scheme and provide details of more careful schemes that result in improved congestion bounds. We provide two proofs. In our opinion, the first one is more natural and easier to understand, but it yields a worse congestion guarantee of 6. The second one is slightly more involved, but it achieves the promised congestion of 5.

Both proofs use the following flow-augmentation procedure.

LEMMA 5.2 (see [11, Theorem 2.1]). *Let G be a directed graph with integer edge capacities. Given a flow h in G that goes from set $X \subseteq V(G)$ to a single vertex $u \in V(G)$, such that for every $v \in X$ the amount of flow originating in v is $\mathbf{h}(v)$, and a vertex $v_0 \in X$ such that $\mathbf{h}(v_0)$ is not an integer, one can in polynomial time compute a flow h' in G, sending $\mathbf{h}'(v)$ amount of flow from every $v \in X$ to u, such that*

1. $|h'| \geq |h|$,
2. $\mathbf{h}'(v) = \mathbf{h}(v)$ *for every $v \in X$, where $\mathbf{h}(v)$ is an integer,*
3. $\mathbf{h}'(v_0) = \lceil \mathbf{h}(v_0) \rceil$,
4. $\mathbf{h}'(v) \leq \mathbf{h}(v)$ *for every $v \in X \setminus \{v_0\}$.*

As node- and edge-capacitated flows are equivalent in the directed case, Lemma 5.2 applies also to node-capacitated flows. Furthermore, observe that the conditions of Lemma 5.2 imply that

$$\begin{aligned} \sum_{v \in X \setminus \{v_0\}} (\mathbf{h}(v) - \mathbf{h}'(v)) &= (|h| - |h'|) - (\mathbf{h}(v_0) - \mathbf{h}'(v_0)) \\ &\leq \mathbf{h}'(v_0) - \mathbf{h}(v_0) = \lceil \mathbf{h}(v_0) \rceil - \mathbf{h}(v_0). \end{aligned} \tag{5.1}$$

5.1. The first proof. If $|X| = \mathcal{O}(\Delta^2/\alpha)$, it suffices just to route one terminal pair. This is always possible since X is in a strongly connected component and we can route one pair with congestion 2. Otherwise, we apply Theorem 1.3 to obtain concentric directed cycles $C'_1, C'_2, \ldots, C'_q$, sets Y^+, Y^-, and path families $\mathcal{P}^+$ and $\mathcal{P}^-$ satisfying $|Y^+| = |Y^-| = \Omega(\alpha^2|X|/\Delta^2)$ and $q = \Omega(\alpha|X|/\Delta)$. By symmetry, we assume that the first outcome of Theorem 1.3 happens: no cycle C'_i encloses any vertex of $Y^+ \cup Y^-$, and the paths from $\mathcal{P}^+ \cup \mathcal{P}^-$ have one of their endpoints on the innermost cycle C'_q.

Recall that $\mathcal{M}$ is a matching on the terminals. Therefore, we can construct, in a greedy manner, a subset $\mathcal{M}_1$ of $|Y^+|/2 = |Y^-|/2$ terminal pairs, such that for every pair $\{s,t\}$ in $\mathcal{M}_1$ we have $s \in Y^+$; we will henceforth refer to the terminal s as the *s-terminal* of the pair $\{s,t\}$, and to t as the *t-terminal*. Let $X_1 = V(\mathcal{M}_1)$, and partition $X_1 = X_{\mathtt{s}} \uplus X_{\mathtt{t}}$ by putting the s-terminal of every terminal pair into $X_{\mathtt{s}}$ and the t-terminal into $X_{\mathtt{t}}$.

Since X is α-node-well-linked, there exists a flow f^+ sending $\alpha|Y^+|$ amount of flow from X_1 to Y^+, such that every vertex of X_1 sends α amount of flow and every vertex of Y^+ receives α amount of flow. Similarly, there exists a flow f^- from Y^- to X_1, such that every vertex of Y^- sends α amount of flow, and every vertex of X_1 receives α amount of flow. The partition $X_1 = X_{\mathtt{s}} \uplus X_{\mathtt{t}}$ naturally induces a split of the flows into $f^+ = f^+_{\mathtt{s}} + f^+_{\mathtt{t}}$ and $f^- = f^-_{\mathtt{s}} + f^-_{\mathtt{t}}$; as $X_{\mathtt{s}} \subseteq Y^+$, we assume that $f^+_{\mathtt{s}}$ is a trivial flow with zero-length flow paths.

We now apply the rounding procedure of Lemma 5.2. We observe that if we add a supersource being an out-neighbor of every vertex in Y^+, then the (naturally extended) flow $f^+_{\mathtt{t}}$ satisfies the assumptions of Lemma 5.2; a similar statement holds for Y^- and the reversed flows $f^-_{\mathtt{s}}$ and $f^-_{\mathtt{t}}$ in the reversed graph G.

Using Lemma 5.2, we iteratively modify the flows $f^+_{\mathtt{t}}$, $f^-_{\mathtt{s}}$, and $f^-_{\mathtt{t}}$ as follows, maintaining the following invariant:

$$\begin{aligned}&\text{for every terminal pair } \{s,t\} \in \mathcal{M}_1,\\ &\text{the same amout of flow is sent by } t \text{ in } f^+_{\mathtt{t}},\\ &\text{received by } s \text{ in } f^-_{\mathtt{s}}, \text{ and received by } t \text{ in } f^-_{\mathtt{t}}.\end{aligned} \tag{5.2}$$

We note that we apply the lemma in three separate copies of G. If this amount of flow is integral for every pair $\{s,t\} \in \mathcal{M}_1$, we stop the rounding algorithm. Otherwise, we pick a pair $\{s,t\}$ for which it is not integral, and apply Lemma 5.2 separately to s in $f^-_{\mathtt{s}}$ and to t in $f^+_{\mathtt{t}}$ and in $f^-_{\mathtt{t}}$. Let $g^-_{\mathtt{s}}$, $g^+_{\mathtt{t}}$, and $g^-_{\mathtt{t}}$ be the computed flows. To maintain the invariant (5.2), we artificially restrict the flows such that for every $\{s',t'\} \in \mathcal{M}_1$, the amounts of flow originating in t' in $g^+_{\mathtt{t}}$, received by s' in $g^-_{\mathtt{s}}$ and received by t' in $g^-_{\mathtt{t}}$, are equal to the minimum of these three numbers, and proceed further with the flows $g^+_{\mathtt{t}}$, $g^-_{\mathtt{s}}$, and $g^-_{\mathtt{t}}$.

The procedure stops after at most $|\mathcal{M}_1|$ steps, as in every step the number of terminal pairs $\{s,t\}$, where the amount of flow originating in t in $f^+_{\mathtt{t}}$ is integral, strictly increases. Furthermore, consider a step and let γ be the amount of flow in $f^+_{\mathtt{t}}$ originating in t. Then, applying Lemma 5.2 caused a decrease of at most $(1-\gamma)$ of the total flow value originating from sources other than t, due to (5.1). Due to the artificial restriction of other flows to maintain the invariant (5.2), this decrease also applies to $f^-_{\mathtt{s}}$ and $f^-_{\mathtt{t}}$. Since we apply Lemma 5.2 three times, the total decrease in each of these flows is at most $3(1-\gamma)$. Consequently, if $h^+_{\mathtt{s}}$, $h^+_{\mathtt{t}}$, $h^-_{\mathtt{s}}$, and $h^+_{\mathtt{t}}$ are the flows after the aforementioned rounding procedure terminates, then $|h^+_{\mathtt{s}}| \geq |f^+_{\mathtt{s}}|/3 = \Omega(\alpha|Y^+|) = \Omega(\alpha^3|X|/\Delta^2)$.

Let $\mathcal{M}_2 \subseteq \mathcal{M}_1$ be the set of pairs $\{s,t\}$ for which $h^+_{\mathtt{t}}$ sends a positive amount of flow from t to Y^+. We infer that the flows $h^+_{\mathtt{t}}$, $h^-_{\mathtt{s}}$, and $h^-_{\mathtt{t}}$ correspond to three families of node-disjoint paths: $\mathcal{P}^+_{\mathtt{t}}$ going from the t-terminals of $\mathcal{M}_2$ to Y^+, $\mathcal{P}^-_{\mathtt{s}}$ going from Y^- to the s-terminals of $\mathcal{M}_2$, and $\mathcal{P}^-_{\mathtt{t}}$ going from Y^- to the t-terminals of $\mathcal{M}_2$. Note that paths between the families may intersect. For every terminal pair $\{s,t\} \in \mathcal{M}_2$, pick two private cycles C'_i and C'_j, and

- route the path from s to t via a path in $\mathcal{P}^+$ starting in s, the cycle C'_i, and a concatenation of a path of $\mathcal{P}^-$ and path in $\mathcal{P}^-_{\mathtt{t}}$, ending in t,
- route the path from t to s via a path in $\mathcal{P}^+_{\mathtt{t}}$ starting in t, continuing along a path in $\mathcal{P}^+$, the cycle C'_j, and a concatenation of a path in $\mathcal{P}^-$ and a path in $\mathcal{P}^-_{\mathtt{s}}$ ending in s.

Clearly such a routing routes all pairs in $\mathcal{M}_2$. To see that it has the promised congestion 6, note that any node u is contained in at most one path from each of the families $\mathcal{P}^+$, $\mathcal{P}^-$, $\mathcal{P}^+_{\mathtt{t}}$, $\mathcal{P}^-_{\mathtt{s}}$, $\mathcal{P}^-_{\mathtt{t}}$, and in at most one of the concentric cycles. Furthermore, $|\mathcal{M}_2| = |h^+_{\mathtt{t}}| = \Omega(\alpha^3|X|/\Delta^2)$. This concludes the proof of Theorem 1.4 (with congestion 6).

5.2. The second proof. First, let us enhance Lemma 5.2 to the following form. This is in fact very similar to the use of Lemma 5.2 in the previous proof, but here we prefer to make it more formal.

LEMMA 5.3 (see [11, section 3.2]). *Assume we are given a set X and a sequence of r tuples (G_i, t_i, f_i) for $1 \leq i \leq r$, where G_i is an edge- or node-capacitated graph with integral capacities, $t_i \in V(G_i)$, $X \subseteq V(G_i)$ (i.e., all graphs G_i have vertices X in common) and f_i is a flow in G_i from X to a single sink t_i, such that for every*

$x \in X$ the same amount of flow originates in x in every flow f_i. Then one can in polynomial time find integral flows h_i from X to t_i in G_i such that for every $x \in X$ the same amount of flow originates in x in every flow h_i, and $|h_1| \geq |f_1|/r$.

Proof. As node and edge-capacitated flows are equivalent in the directed case, Lemma 5.2 applies also to node-capacitated flows.

Using Lemma 5.2, we iteratively modify the flows f_i as follows, maintaining the invariant that for every $x \in X$ the same amount of flow originates in f_i for every i. If this amount of flow is integral for every $x \in X$, we stop the rounding algorithm. Otherwise, we pick a node $x \in X$ for which it is not integral, and apply Lemma 5.2 separately for every $1 \leq i \leq r$ to the vertex x in the flow f_i in G_i. Let g_i for $1 \leq i \leq r$ be the computed flows. To maintain the invariant, we artificially restrict the flows such that for every $y \in X$, the amounts of flow sent by y in g_i are the same for every i; we just take the minimum of these r numbers. We proceed further with the flows g_i.

The procedure clearly stops after at most $|X|$ steps, as in every step the number of vertices x where the amount of flow originating in x in f_1 is integral strictly increases. Furthermore, if in a step in f_1 there was γ amount of flow originating in x, then by (5.1) the total loss in the amount of flow for vertices $y \in X \setminus \{x\}$ in each of the r considered flows is at most $r(\lceil\gamma\rceil - \gamma)$. Consequently, if h_i for $1 \leq i \leq r$ are the flows after the aforementioned rounding procedure terminates, then $|h_i| \geq |f_i|/r$. This finishes the proof of Lemma 5.3. ∎

We now go back to the proof of Theorem 1.4. Again, if $|X| = \mathcal{O}(\Delta^2/\alpha)$, it suffices just to route one terminal pair. Otherwise, we apply Theorem 1.3 to obtain concentric directed cycles $C'_1, C'_2, \ldots, C'_q$, sets Y^+, Y^-, and path families $\mathcal{P}^+$ and $\mathcal{P}^-$ intersecting every cycle C'_i. We have $|Y^+| = |Y^-| = \Omega(\alpha^2|X|/\Delta^2)$ and $q = \Omega(\alpha|X|/\Delta)$.

Classify every terminal into one of the three caterogies: (a) not enclosed by $C_{\lceil q/3\rceil}$; (b) enclosed by $C_{\lceil q/3\rceil}$, but not enclosed by $C_{\lceil 2q/3\rceil}$; (c) enclosed by $C_{\lceil 2q/3\rceil}$. Classify every terminal pair according to the classification of its terminals; as the terminal pairs are unordered, we have six categories. Let $\mathcal{M}_1$ be the subset of size $|Y^+|$ of the category with the largest number of terminal pairs; recall that $|Y^+|$ is much smaller than $|\mathcal{M}|$. In every terminal pair of $\mathcal{M}_1$, distinguish an *s-terminal* and a *t-terminal* of the pair, such that all s-terminals of $\mathcal{M}_1$ have the same category, and all t-terminals of $\mathcal{M}_1$ have the same category. Let $X_{\mathtt{s}}$ and $X_{\mathtt{t}}$ be the sets of the s- and t-terminals of $\mathcal{M}_1$, respectively.

Let $\hat{q} = \lceil q/3\rceil - 2$, and pick a subsequence $D_1, D_2, \ldots, D_{\hat{q}}$ of the cycles C'_i such that either all s-terminals are enclosed by the innermost cycle $D_{\hat{q}}$, or neither of them is enclosed by the outermost cycle D_1, and a symmetric condition holds for t-terminals. (We allow, e.g., that $D_{\hat{q}}$ encloses all s-terminals but D_1 does not enclose any t-terminal.) Note that one can always choose either the first $\hat{q}$, the middle $\hat{q}$, or the last $\hat{q}$ of the cycles C'_i as the sequence $D_1, D_2, \ldots, D_{\hat{q}}$, due to the fact that all s-terminals are of the same category and all t-terminals are of the same category.

Let $D^{\mathtt{s}}$ be the cycle D_i that is "opposite" to the s-terminals of $\mathcal{M}_1$, that is, $D^{\mathtt{s}} = D_{\hat{q}}$ if no s-terminal of $\mathcal{M}_1$ is enclosed by D_1, and $D^{\mathtt{s}} = D_1$ otherwise. Similarly define $D^{\mathtt{t}}$ as the cycle opposite the t-terminals. Since X is α-node-well-linked, there exists a flow sending $\alpha|Y^+|$ amount of flow from $X_{\mathtt{s}}$ to Y^+, such that every vertex of $X_{\mathtt{s}}$ sends α amount of flow and every vertex of Y^+ receives α amount of flow. By combining this flow with subpaths of the paths $\mathcal{P}^+$, we obtain a flow $f_{\mathtt{s}}^+$ sending $(\alpha/2)|Y^+|$ amount of flow from $X_{\mathtt{s}}$ to $D^{\mathtt{s}}$, such that every vertex $X_{\mathtt{s}}$ sends $(\alpha/2)$ amount of flow and every vertex of $D^{\mathtt{s}}$ receives at most $(\alpha/2)$ amount of flow. Similarly we construct flows: $f_{\mathtt{s}}^-$ from $D^{\mathtt{s}}$ to $X_{\mathtt{s}}$, $f_{\mathtt{t}}^+$ from $X_{\mathtt{t}}$ to $D^{\mathtt{t}}$, and $f_{\mathtt{t}}^-$ from $D^{\mathtt{t}}$ to $X_{\mathtt{t}}$.

We apply Lemma 5.3 to four tuples (G_i, t_i, f_i): we add a supersink being an out-neighbor of every vertex in $D^{\mathtt{s}}$ and extend the flow $f_{\mathtt{s}}^+$ to reach this supersink, and we perform a similar construction for $f_{\mathtt{t}}^+$ to a supersink being an out-neighbor of every vertex $D^{\mathtt{t}}$, the reversed $f_{\mathtt{s}}^-$ in the reversed graph G to a supersink near $D^{\mathtt{s}}$, and the reversed $f_{\mathtt{t}}^-$ in the reversed graph G to a supersink near $D^{\mathtt{t}}$. Note that formally the sources in the first two flows are different than in the second two, but for the purpose of Lemma 5.3 we treat as the set X the set of the s-terminals in the first two flows, and the set of the t-terminals in the second two flows, with the natural correspondance between these two sets given by $\mathcal{M}_1$.

Let $h_{\mathtt{s}}^+$, $h_{\mathtt{t}}^+$, $h_{\mathtt{s}}^-$, and $h_{\mathtt{t}}^+$ be the output flows. We have $|h_{\mathtt{s}}^+| \geq |f_{\mathtt{s}}^+|/4 = \Omega(\alpha|Y^+|) = \Omega(\alpha^3|X|/\Delta^2)$. Let $\mathcal{M}_2 \subseteq \mathcal{M}_1$ be the set of pairs $\{s,t\}$ for which $h_{\mathtt{s}}^+$ sends positive amount of flow from t to Y^+. We infer that the flows $h_{\mathtt{s}}^+$, $h_{\mathtt{t}}^+$, $h_{\mathtt{s}}^-$, and $h_{\mathtt{t}}^-$ correspond to four families of node-disjoint paths: $\mathcal{P}_{\mathtt{s}}^+$ going from the s-terminals of $\mathcal{M}_2$ to $D^{\mathtt{s}}$, $\mathcal{P}_{\mathtt{t}}^+$ going from the t-terminals of $\mathcal{M}_2$ to $D^{\mathtt{t}}$, $\mathcal{P}_{\mathtt{s}}^-$ going from $D^{\mathtt{s}}$ to the s-terminals of $\mathcal{M}_2$, and $\mathcal{P}_{\mathtt{s}}^-$ going from $D^{\mathtt{t}}$ to the t-terminals of $\mathcal{M}_2$. Note that paths between the families may intersect. By the choice of the cycles $D^{\mathtt{s}}$ and $D^{\mathtt{t}}$, every path in each of these four families intersects every cycle D_i. For every terminal pair $\{s,t\} \in \mathcal{M}_2$, pick two private cycles D_i and D_j, and

- route the path from s to t via a path in $\mathcal{P}_{\mathtt{s}}^+$ starting in s, the cycle D_i, and a path of $\mathcal{P}_{\mathtt{t}}^-$ ending in t,
- route the path from t to s via a path in $\mathcal{P}_{\mathtt{t}}^+$ starting in t, the cycle D_j, and a path in $\mathcal{P}_{\mathtt{s}}^-$ ending in s.

Clearly such a routing routes all pairs in $\mathcal{M}_2$. To see that it has the promised congestion 5, note that any node u is contained in at most one path from each of the families $\mathcal{P}_{\mathtt{s}}^+$, $\mathcal{P}_{\mathtt{t}}^+$, $\mathcal{P}_{\mathtt{s}}^-$, $\mathcal{P}_{\mathtt{t}}^-$, and in at most one of the concentric cycles. Furthermore, $|\mathcal{M}_2| = |h_{\mathtt{s}}^+| = \Omega(\alpha^3|X|/\Delta^2)$. This concludes the proof of Theorem 1.4.

6. Concluding remarks. Our main technical contribution in this paper is to show that a planar directed graph has a constant congestion routing structure of size $\Omega(h/\mathrm{polylog}(h))$, where $h = \mathrm{dtw}(G)$. This structural result was motivated by the algorithmic problem of routing symmetric demands in directed graphs. Recent results, in the undirected graph setting, have demonstrated effectively the inherent synergy between approximation algorithms for routing problems and structural results in graph theory related to treewidth. The work in [7] and here are steps toward extending this synergy to directed graphs. The directed graph setting is significantly more challenging, however, and progress in this direction could yield several new benefits. We raise some open problems below.

- Does a planar directed graph with treewidth h have a constant congestion crossbar of size $\Omega(h)$? This would strengthen our result. In particular, is there a cylindrical grid minor of size $\Omega(h)$?
- Can we obtain a polylogarithmic approximation in planar graphs with congestion 2, as in the undirected case?
- The techniques in this paper could likely be extended to directed graphs that can be embedded on a bounded genus surface, and more generally to directed graphs whose undirected support graph is from a proper minor-closed family. The ideas of well-linked decomposition and degree-reduction do not rely on planarity. Moreover, there is a linear relationship between treewidth and the size of a grid-minor in undirected graphs from a proper minor-closed family [21].

- Does a general directed graph with treewidth h have a constant congestion crossbar of size $\Omega(h/\text{polylog}(h))$? Is there a cylindrical grid minor of size $\Omega(h^\delta)$ for some fixed $\delta > 0$?

REFERENCES

[1] M. Andrews, *Approximation algorithms for the edge-disjoint paths problem via Raecke decompositions*, in Proceedings of IEEE FOCS, 2010, pp. 277–286.

[2] M. Andrews, J. Chuzhoy, V. Guruswami, S. Khanna, K. Talwar, and L. Zhang, *Inapproximability of edge-disjoint paths and low congestion routing on undirected graphs*, Combinatorica, 30 (2010), pp. 485–520.

[3] C. Chekuri and J. Chuzhoy, *Large-treewidth graph decompositions and applications*, in Procedings of ACM STOC, 2013, pp. 291–300.

[4] C. Chekuri and J. Chuzhoy, *Degree-3 treewidth sparsifiers*, in Proceedings of ACM-SIAM SODA, 2015, pp. 242–255.

[5] C. Chekuri and J. Chuzhoy, *Polynomial bounds for the grid-minor theorem*, J. ACM, 63 (2016), pp. 40:1–40:65; also available online from http://dl.acm.org/citation.cfm?id=2820609.

[6] C. Chekuri and A. Ene, *Poly-logarithmic approximation for maximum node disjoint paths with constant congestion*, in Proceedings of ACM-SIAM SODA, 2013, pp. 326–341.

[7] C. Chekuri and A. Ene, *The all-or-nothing flow problem in directed graphs with symmetric demand pairs*, Math. Program., 154 (2015), pp. 249–272.

[8] C. Chekuri, A. Ene, and M. Pilipczuk, *Constant congestion routing of symmetric demands in planar directed graphs*, in Proceedings of the 43rd International Colloquium on Automata, Languages, and Programming, 2016, Rome, Italy, I. Chatzigiannakis, M. Mitzenmacher, Y. Rabani, and D. Sangiorgi, eds., pp. 7:1–7:14; also available online from https://doi.org/10.4230/LIPIcs.ICALP.2016.7.

[9] C. Chekuri, S. Kannan, A. Raja, and P. Viswanath, *Multicommodity flows and cuts in polymatroidal networks*, SIAM J. Comput., 44 (2015), pp. 912–943.

[10] C. Chekuri, S. Khanna, and F. Shepherd, *Multicommodity flow, well-linked terminals, and routing problems*, in Procedings of ACM STOC, 2005, pp. 183–192.

[11] C. Chekuri, S. Khanna, and F. Shepherd, *An $O(\sqrt{n})$ approximation and integrality gap for disjoint paths and unsplittable flow*, Theory Comput., 2 (2006), pp. 137–146.

[12] C. Chekuri, S. Khanna, and F. B. Shepherd, *The all-or-nothing multicommodity flow problem*, SIAM J. Comput., 42 (2013), pp. 1467–1493.

[13] J. Chen, Y. Liu, S. Lu, B. O'Sullivan, and I. Razgon, *A fixed-parameter algorithm for the directed feedback vertex set problem*, J. ACM, 55 (2008).

[14] J. Chuzhoy, *Excluded grid theorem: Improved and simplified*, in Proceedings of ACM STOC, 2015, pp. 645–654.

[15] J. Chuzhoy, *Routing in undirected graphs with constant congestion*, SIAM J. Comput., 45 (2016), pp. 1490–1532, https://doi.org/10.1137/130910464.

[16] J. Chuzhoy, V. Guruswami, S. Khanna, and K. Talwar, *Hardness of routing with congestion in directed graphs*, in Proceedings of ACM STOC, 2007, pp. 165–178.

[17] J. Chuzhoy, D. H. Kim, and R. Nimavat, *New hardness results for routing on disjoint paths*, in Proceedings of ACM STOC, 2017, pp. 86–99.

[18] J. Chuzhoy, D. H. K. Kim, and S. Li, *Improved approximation for node-disjoint paths in planar graphs*, in Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, Cambridge, MA, 2016, D. Wichs and Y. Mans, eds., pp. 556–569, https://doi.org/10.1145/2897518.2897538.

[19] J. Chuzhoy and S. Li, *A polylogarithmic approximation algorithm for edge-disjoint paths with congestion* 2, J. ACM, 63 (2016), 45.

[20] M. Cygan, F. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh, *Parameterized Algorithms*, Springer, New York, 2015.

[21] E. D. Demaine and M. Hajiaghayi, *Linearity of grid minors in treewidth with applications through bidimensionality*, Combinatorica, 28 (2008), pp. 19–36.

[22] P. Erdos and L. Pósa, *On independent circuits contained in a graph*, Canad. J. Math., 17 (1965), pp. 347–352.

[23] G. Even, J. Naor, S. Rao, and B. Schieber, *Divide-and-conquer approximation algorithms via spreading metrics*, J. ACM, 47 (2000), pp. 585–616.

[24] S. Fortune, J. Hopcroft, and J. Wyllie, *The directed subgraph homeomorphism problem*, Theoret. Comput. Sci., 10 (1980), pp. 111–121.

[25] T. Johnson, N. Robertson, P. D. Seymour, and R. Thomas, *Directed tree-width*, J. Combin. Theory Ser. B, 82 (2001), pp. 138–154.
[26] T. Johnson, N. Robertson, P. D. Seymour, and R. Thomas, *Excluding a Grid Minor in Digraphs*, manuscript, http://arxiv.org/abs/1510.00473, 2001.
[27] S. Kamath, S. Kannan, and P. Viswanath, *Network capacity under traffic symmetry: Wireline and wireless networks*, IEEE Trans. Inform. Theory, 60 (2014), pp. 5457–5469.
[28] S. Kannan and P. Viswanath, *Capacity of multiple unicast in wireless networks: A polymatroidal approach*, IEEE Trans. Inform. Theory, 60 (2014), pp. 6303–6328.
[29] R. M. Karp, *On the complexity of combinatorial problems*, Networks, 5 (1975), pp. 45–68.
[30] K. Kawarabayashi and S. Kreutzer, *An excluded grid theorem for digraphs with forbidden minors*, in Proceedings of ACM-SIAM SODA, 2014, pp. 72–81.
[31] K. Kawarabayashi and S. Kreutzer, *The directed grid theorem*, in Proceedings of ACM STOC, 2015.
[32] R. Khandekar, S. Rao, and U. Vazirani, *Graph partitioning using single commodity flows*, J. ACM, 56 (2009), p. 19.
[33] P. N. Klein, S. Plotkin, S. Rao, and E. Tardos, *Approximation algorithms for Steiner and directed multicuts*, J. Algorithms, 22 (1997), pp. 241–269.
[34] S. G. Kolliopoulos and C. Stein, *Approximating disjoint-path problems using packing integer programs*, Math. Program., 99 (2004), pp. 63–87.
[35] T. Leighton and S. Rao, *Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms*, J. ACM, 46 (1999), pp. 787–832.
[36] A. Louis, *Cut-Matching Games on Directed Graphs*, CoRR, abs/1010.1047, 2010.
[37] S. Rao and S. Zhou, *Edge disjoint paths in moderately connected graphs*, SIAM J. Comput., 39 (2010), pp. 1856–1887.
[38] B. Reed, *Introducing directed tree width*, Electron. Notes Discrete Math., 3 (1999), pp. 222–229.
[39] B. Reed, N. Robertson, P. D. Seymour, and R. Thomas, *Packing directed circuits*, Combinatorica, 16 (1996), pp. 535–554.
[40] B. A. Reed and F. B. Shepherd, *The Gallai–Younger conjecture for planar graphs*, Combinatorica, 16 (1996), pp. 555–566.
[41] N. Robertson and P. D. Seymour, *Graph minors* XIII. *The disjoint paths problem*, J. Combin. Theory Ser. B, 63 (1995), pp. 65–110, https://doi.org/10.1006/jctb.1995.1006.
[42] N. Robertson, P. D. Seymour, and R. Thomas, *Quickly excluding a planar graph*, J. Combin. Theory Ser. B, 62 (1994), pp. 323–348.
[43] P. D. Seymour, *Packing directed circuits fractionally*, Combinatorica, 15 (1995), pp. 281–288.