Uncrossed Multiflows and Applications to Disjoint Paths

Chandra Chekuri
Department of Computer Science, University of Illinois, Urbana-Champaign, IL, USA

Guyslain Naves
Aix-Marseille Univ, CNRS, LIS, Marseille, France

Joseph Poremba[1]
Department of Computer Science, University of British Columbia, Vancouver, Canada

F. Bruce Shepherd
Department of Computer Science, University of British Columbia, Vancouver, Canada

Abstract

A multiflow in a planar graph is *uncrossed* if the curves identified by its support paths do not cross in the plane. Recently, uncrossed flows have played a role in approximation algorithms for maximum disjoint paths in "fully-planar" instances, where the combined supply-plus-demand graph is planar. They are also used in algorithms to find low-congestion unsplittable flows for both fully-planar and single-source instances. For these two instance classes, any fractional multiflow can be converted into one that is uncrossed, which these algorithms then exploit to obtain their results.

We investigate the utility of uncrossed flow more generally and ask three key questions. First, are there other interesting planar multiflow instances that admit uncrossed flows (beyond fully-planar and single-source)? We answer affirmatively, demonstrating a new family of "pairwise-planar" instances whose fractional flows can be uncrossed. This family subsumes fully-planar but includes substantially more, such as (2-connected) fully-compliant series-parallel instances and some instances that have large clique demand graphs. Second, can we always round a fractional uncrossed flow to a "good" integral flow? We again answer positively. For maximization problems, we show any fractional uncrossed flow can be rounded to an integral flow with a constant fraction of its value. For congestion problems (where we must fully route all given demands), we give a rounding procedure that yields an integral multiflow with edge congestion 2. Consequently, we obtain constant-factor approximation algorithms for maximum disjoint paths and minimum congestion integer multiflow for pairwise-planar instances, and show such instances have a constant integral flow-multicut gap. Finally we ask, given an arbitrary planar instance, can we determine if there exists a congestion-1 uncrossed fractional flow (congestion setting) or find the maximum value uncrossed fractional flow (maximization setting)? For congestion, we show this problem is NP-hard, but finding uncrossed edge-disjoint paths is polytime solvable if the demands span a bounded number of faces. For maximization, we present a strong (almost-polynomial) inapproximability result.

2012 ACM Subject Classification Mathematics of computing → Network flows; Mathematics of computing → Approximation algorithms

Keywords and phrases Network Flows, Disjoint Paths, Planar Graphs, Crossing, Flow-Multicut Gap, Approximation Algorithms, Combinatorial Optimization

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2026.12

Category APPROX

Related Version *Full Version*: https://arxiv.org/abs/2511.00254

Funding *Chandra Chekuri*: Supported in part by NSF grant CCF-2402667.
F. Bruce Shepherd: Supported by NSERC Discovery Grant RGPIN-2026-06972.

[1] Corresponding author

Figure 1 A fully-planar multiflow instance with two different multiflows routing its demands. Dashed edges represent demands, and coloured paths are support paths of the flow. The support paths on the left cross, while the multiflow on the right is uncrossed.

1 Introduction

In *multicommodity flow*, we are given an edge-capacitated *supply graph* (G, u) and a set of source-sink pairs called *demands* or *commodities*. The demands are represented by a *demand graph* H on the same node set as G. Unless otherwise specified, graphs are *undirected*. We seek to route flow for the demands simultaneously, while respecting the capacities of the supply graph. There are two flavours of the multiflow problem. In the *maximization* setting, the objective is to find a flow x that respects the capacities and maximizes the total sum of demands being routed. A feasible flow for (G, H, u) needs to respect the capacities, but has freedom in the quantities of each demand that it routes. In the *congestion* setting, we are given target amounts $d(s, t)$ for each $(s, t) \in E(H)$. To be feasible for (G, H, u, d), a flow must (in addition to respecting capacities) fully route this amount for each demand. The objective is to determine whether d can be satisfied in the capacities u, or to find the minimum scaling factor $\alpha \geq 1$ such that d can be satisfied in capacities αu.

Both fractional and integer versions of these problems are fundamental in combinatorial optimization (cf. [28]). Due to applications, the literature often emphasizes problems where the supply graph is planar [20]. The integer problems are hard even for planar supply graphs with 3 demand edges [33]. There exist, however, good approximation algorithms in so-called *fully-planar* instances where the joint supply-demand graph $G + H$ is planar[2]. Often approximation algorithms follow a "relax-and-round" approach: relax the integer problem to its fractional version, find a fractional solution with "nice properties", and then round the nice fractional solution to an integral solution.

A recurring "nice property" for planar multiflow is that the support paths (i.e. paths routing non-zero flow) do not cross in the plane. A multiflow satisfying this property is *uncrossed* (see Figure 1). It is known how to construct uncrossed multiflows for single-source (H is a star) and fully-planar instances, the latter using an algorithm of Matsumoto et al. [21]. This was recently leveraged to give constant-factor approximation algorithms for *maximum edge-disjoint paths* (MEDP) [15] and *maximum node-disjoint paths* (MNDP) [25, 24] for fully-planar instances. Very recently uncrossed flows were used to round fractional flows to *unsplittable* flows, routing the same demand quantities with small additive error, in both single-source [31] and fully-planar contexts [12]. In an unsplittable flow, each demand is routed on a single path.

[2] In this paper, if we say a multiflow instance is *planar* we only mean that the supply graph G is planar. If we mean $G + H$ is planar, we will say fully-planar.

In the above results, the uncrossed property of the fractional solution plays a key role, but the arguments are still specialized to the structure of (G, H). For example, for the fully-planar context, the uncrossed flow can be shown to be *laminar*: if one considers the family of cycles obtained by joining demand edges of H to their support paths in G, the result is a nested family of cycles in the plane. However, this same property does not hold for all instances that admit uncrossed flows. For instance, if G is a cycle and H consists of mutually crossing demands, then no paths within G cross, but the path-plus-demand cycles are not nested. Our aim is to expand on a theory of uncrossed flows - when can we find them, and do they always admit good rounding algorithms, even outside the laminar/single-source cases? We have three motivating questions.

▶ **Question 1.** *Do there exist other (interesting) families of (G, H) satisfying the following property: for all capacities u and demand values d, if (G, H, u, d) has a feasible (fractional) multiflow x, then (G, H, u, d) has a feasible multiflow y that is uncrossed?*

We say (G, H) is an *uncrossable layout* if it satisfies the conditions of Question 1. For instance, if (G, H) is fully-planar or H is single-source, then (G, H) is an uncrossable layout.

▶ **Question 2.** *Suppose we are given a feasible fractional multiflow x that is uncrossed (not assuming that the multiflow x is laminar). Can we round x to a "good" integer multiflow y? Specifically:*

1. *(Congestion Setting) Given feasible uncrossed fractional multiflow x for (G, H, u, d), can we compute an integer multiflow y feasible for $(G, H, \alpha u, d)$, for some constant α?*
2. *(Maximization Setting) Given feasible uncrossed fractional multiflow x for (G, H, u), can we compute an integer multiflow y feasible for (G, H, u) whose value is $1/\beta$ times that of x, for some constant β?*

The answer to Question 1 and both parts of Question 2 is: yes! First, we define a new class of (G, H) and prove it gives uncrossable layouts.

▶ **Theorem 1.** *Let G be a planar graph with a fixed embedding. Let H be such that for any pair of demands $h_1, h_2 \in E(H)$, h_1 and h_2 can be embedded inside (possibly different) faces of G so they do not cross each other. Then (G, H) is an uncrossable layout.*

We say any (G, H) satisfying this property is *pairwise-planar* (see Figure 6 in Section 3 for a depiction). The pairwise-planar class includes fully-planar as a subset, but includes substantially more. For instance, it contains layouts where $G + H$ is highly non-planar (and thus, do not admit laminar flows). It also includes the 2-connected *fully-compliant*[3] family introduced in [7], which were not previously known to give uncrossed layouts to our knowledge. Pairwise-planar also differs from previous uncrossable layouts in an important structural sense. Namely, fully-planar and single-source (G, H) are *cut-sufficient* (essentially, they have a Max-Flow Min-Cut theorem in the congestion setting), while some pairwise-planar (G, H) are not.

As for Question 2, we prove the following results. We begin with the maximization setting.

▶ **Theorem 2.** *Let (G, H, u) be a planar maximization instance where u is integral. Let f be a feasible fractional flow that is uncrossed. There exists a feasible integral flow f' achieving an $\Omega(1)$ fraction of the value of f. The integral flow can be computed in polynomial time.*

[3] (G, H) is fully-compliant if G is series-parallel and $G + h$ is series-parallel for every $h \in E(H)$.

We prove this result by establishing an "approximate integer decomposition property", exploiting colouring results on a restricted class of string graphs (path intersection graphs). This integer decomposition also implies $\Omega(1)$ rounding holds for the *weighted* maximization version, where each $(s,t) \in E(H)$ has an associated weight $w(s,t)$, and a flow achieves $w(s,t)$ value for each unit of (s,t)-flow that it routes. Additionally, this result extends to *node-capacitated* multiflows (i.e. u is a capacity function on nodes rather than edges).

For the congestion setting, we give a factor 2 rounding for converting a fractional uncrossed multiflow into an integral multiflow. Moreover, for an extra additive penalty we can round to an unsplittable flow.

▶ **Theorem 3.** *Let (G,H,u,d) be a planar congestion instance where u,d are integral. Let f be a feasible fractional multiflow that is strongly uncrossed*[4]*. Then there exists both:*

1. *an integral multiflow f' that is feasible for (G,H,u',d) where $u'(e) \le 2u(e)$ for each $e \in E(G)$, and*
2. *an unsplittable multiflow f'' that is feasible for (G,H,u'',d) where $u''(e) < 2u(e) + d_{\max}$ for every $e \in E(G)$, where $d_{\max} = \max_{h \in E(H)} d(h)$.*

The integral and unsplittable multiflows can be computed in polynomial time.

Using Theorems 1, 2, and 3, we obtain constant-factor approximation algorithms for integer multiflow problems with pairwise-planar inputs.

▶ **Corollary 4.** *There exist polynomial time constant-factor approximation algorithms for maximum edge-disjoint paths and maximum node-disjoint paths (and more generally, maximum integer multiflow), as well as minimum congestion integer multiflow, for inputs where (G,H) is pairwise-planar.*

Another interesting corollary concerns the *integral flow-multicut gap*, which is the ratio of the minimum capacity of a multicut (a set $F \subseteq E(G)$ whose removal destroys all paths connecting demands of H) to the maximum value of an *integral* multiflow. Using their flow rounding algorithms, [17] [15] show the gap is constant for fully-planar (G,H), but the gap can be as large as $\Theta(|E(H)|)$ even for planar supply graphs [14]. Combining our results with a theorem of [30], we obtain a constant gap for pairwise-planar instances.

▶ **Corollary 5.** *For any maximization instance (G,H,u) where (G,H) is pairwise-planar, the minimum capacity of a multicut is at most $O(1)$ times the maximum value of an integral multiflow.*

The third key question requires slightly more motivation. If (G,H) is not an uncrossable layout, it only means the structure of (G,H) alone is insufficient to guarantee uncrossed solutions. However for specific capacities u and demands d we may still be able to find an uncrossed flow and round it. Thus it would be nice to have algorithms to determine whether (in the congestion setting) we can find a feasible fractional flow that is uncrossed, or (in the maximization setting) find the largest feasible flow that is uncrossed.

▶ **Question 3.** *Suppose we are given an* arbitrary *planar multiflow instance. Can we solve the fractional multiflow problem if we constrain the feasible space to only allow uncrossed solutions? Specifically:*

[4] We defer the more technical definition of strongly uncrossed to Section 2.4. Importantly, the uncrossable layouts mentioned in this paper, including pairwise-planar, admit strongly uncrossed flows.

1. *(Congestion Setting) Given (G, H, u, d), can we determine whether there exists a feasible fractional multiflow that is uncrossed?*
2. *(Maximization Setting) Given (G, H, u), can we compute a feasible fractional uncrossed multiflow whose value is (approximately) maximum amongst feasible uncrossed solutions (*maximum uncrossed multiflow*)?*

These questions concern *fractional* multiflows and may feel like linear optimization problems. However the requirement of uncrossed flow paths makes these computationally much more challenging. Interestingly, the positive result of Theorem 2 can be combined with results of [8, 9] to establish strong almost-polynomial inapproximability for maximum uncrossed fractional multiflow (for both edge and node capacities).

▶ **Theorem 6.** *For planar G, maximum uncrossed fractional multiflow is not approximable to within a factor of $2^{\Omega(\log^{1-\epsilon} n)}$ for any $\epsilon > 0$ assuming that $NP \not\subseteq DTIME(n^{polylog\ n})$. Moreover, there is no $n^{O\left(\frac{1}{(\log\log n)^2}\right)}$-approximation polynomial-time algorithm assuming that for some constant $\delta > 0$, $NP \not\subseteq DTIME(2^{n^\delta})$. These results hold even when G is maximum degree 3 (or even G is a wall graph) and the capacities are unit.*

For the congestion setting, we prove the problem posed by Question 3 is NP-complete.

▶ **Theorem 7.** *It is NP-complete to determine whether planar congestion instance (G, H, u, d) has a feasible fractional uncrossed multiflow, even with unit-valued capacities and demands.*

We remark that since this problem is not a linear program anymore, it is not even obvious that the problem is in NP, however we establish this as well. On the positive side, we show that finding uncrossed edge-disjoint paths can be (polytime) reduced to a node-disjoint paths instance. Hence we may find uncrossed solutions if $E(H)$ is incident with a bounded number of faces using a result of Schrijver [27].

▶ **Theorem 8.** *For instances where G is planar, the edges of H are incident with a bounded number of faces, and the capacities and demands are unit-valued, there is a polynomial time algorithm to find a feasible integral uncrossed multiflow for (G, H, u, d) or declare none exists.*

1.1 Organization of the Paper

Section 2 lays the groundwork for the theory of uncrossed flows, including detailed definitions. Section 3 discusses pairwise-planar instances, proves they admit uncrossed flows, and shows a constant integral flow-multicut gap. Section 4 covers our maximization results, both rounding (Theorem 2) and inapproximability (Theorem 6). The rounding result for congestion (Theorem 3) appears in Section 5. This manuscript includes the important aspects of each proof, though occasionally some details are deferred to the full version. The NP-completeness result (Theorem 7) and the integral uncrossed flow algorithm for a bounded number of H-incident faces (Theorem 8) are deferred to the full version.

1.2 Related Work

Uncrossed flows have played a role in previous routing algorithms in planar graphs, such as Schrijver's Homotopy Method [26] for node-disjoint paths. Most relevant are recent algorithms for maximum disjoint paths in fully-planar instances ($G + H$ planar). In [17, 15], constant-factor approximation algorithms are given for maximum edge-disjoint paths, then extended to the node-disjoint setting in [25, 24]. These algorithms glue uncrossed flow paths to their respective demand edges to obtain an uncrossed (laminar) family of cycles in a planar graph. Our rounding algorithms are agnostic in the sense that they just operate on uncrossed flow paths, not assuming that demands lie within a face or that demands of a common face are non-crossing (indeed for our pairwise-planar instances, it is even possible for demands within a face to form a clique).

Uncrossed flows for fully-planar instances are typically constructed by a primal algorithm [21] that computes an uncrossed flow from scratch. We build on a different approach, taking a feasible flow (with many crossings) and iteratively decreasing total crossing (used for example in [18] to find cycles in higher-genus surfaces that cross at most once). We discuss obstacles faced in extending this to pairwise-planar in Section 3.

In [15] the initial flow is first rounded to a half-integral flow (using linear programming theory such as total unimodularity), then this half-integral flow is rounded to an integral flow using a 4-colouring of the (planar) intersection graph of the cycles. Extending to the node-disjoint setting, [25] uses a different argument, iteratively constructing a high-value solution by showing one can always find an "efficient" cycle that only intersects the other cycles on a small set of its nodes. Our rounding technique for Theorem 2 instead uses an approach based on establishing an "approximate integer decomposition property" [3, 5]. A key ingredient comes from looking at a subset of *string graphs*, which are the intersection graphs from curves in the plane. Specifically, we are interested in those that arise from curves that do not cross, studied for example in [11], and for which an important colouring result was proven by Fox and Pach [13][5].

The fully-planar maximum disjoint paths algorithms were extended [25, 18] to uncrossable families of cycles in higher-genus surfaces, yielding approximation algorithms for maximum disjoint paths when $G + H$ is embedded in a surface of genus g; the approximation ratio is polynomial in the genus g. By contrast our work extends uncrossing techniques to a broader family of multiflow instances (pairwise-planar) while remaining *in the plane*. Some pairwise-planar instances have $G + H$ of arbitrarily high genus (e.g. H contains a large clique), but we show nevertheless that they admit uncrossed flows in the plane in order to achieve constant-factor approximations. Results in [25] use the context of uncrossed cycle families, as defined by Goemans and Williamson [16], which is a combinatorial rather than topological definition of crossing.

The works of [17], [15] use their rounding algorithms to establish that the integral flow-multicut gap of fully-planar instances is constant. The integral flow-multicut gap can be as large as $\Theta(|E(H)|)$ for general planar G [14], so further restrictions on (G, H) are needed to establish constant bounds. Other classes known to give constant integral flow-multicut gaps include: when G is a tree [14], when $G + H$ is series-parallel [10] (note that these are less general than fully-compliant), when G has fixed cyclomatic number [1], when $G + H$ has fixed genus [18], and when G has fixed tree-width [4]. A common ingredient in these results is a bound on the *fractional* flow-multicut gap for $K_{r,r}$-minor-free G by [30, 19].

Other works using uncrossed flow come from the study of unsplittable flow in the congestion model (where specified demand quantities must be fully routed). For directed single-source planar instances (H is a star) [31], it is shown how to convert a feasible fractional uncrossed flow into an unsplittable flow that exceeds capacities by a small amount and has no increase in cost. For (undirected) fully-planar instances, [12] shows how to convert a laminar flow into an unsplittable flow y with small capacity violation (no cost guarantee).

[5] This manuscript is unpublished, but the result is mentioned in [32] and their writeup was shared with us [23].

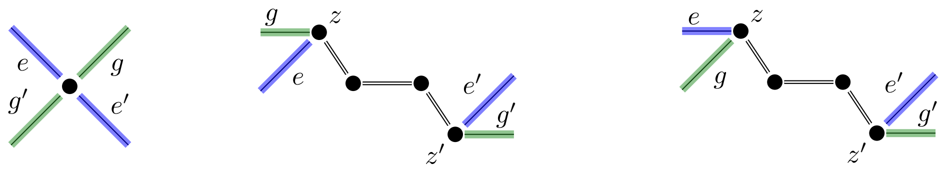


Figure 2 (Left) Two edge-disjoint paths that cross at a common node. (Center) Two paths that cross at a common subpath. (Right) Two paths that share a common subpath, but are uncrossed. Uncoloured edges are shared between the two paths.

2 Uncrossed Multiflows

2.1 Multiflow Preliminaries

All graphs are undirected in this paper. Consider a supply graph $G = (V, E(G))$ and demand graph $H = (V, E(H))$. Endpoints of edges in H are *terminals*, and the edges themselves are *demands* or *commodities*. Together, (G, H) define a *layout*[6]. For nodes $s, t \in V$, let $\mathcal{P}_{st}$ denote the set of st-paths. By *path* we always mean simple path. A *multiflow* is a function $x : \mathcal{P} \to \mathbb{R}_{\geq 0}$, where $\mathcal{P} = \bigcup_{(s,t)\in E(H)} \mathcal{P}_{st}$. Multiflows are fractional unless explicitly stated as integral. A multiflow can also be viewed as a collection of single-commodity flows $(x^{st} : (s,t) \in E(H))$. For $e \in E(G)$, we let $x(e) = \sum_{P\in\mathcal{P}:e\in P} x(P)$ (the amount of flow on e). For $(s,t) \in E(H)$ we let $|x^{st}| = \sum_{P\in\mathcal{P}_{st}} x(P)$ (the amount of demand (s,t) that is routed).

A *maximization instance* has the form (G, H, u), where $u : E(G) \to \mathbb{R}_{\geq 0}$ are *capacities*. A multiflow x is *feasible* for the instance if for every $e \in E(G)$, $x(e) \leq u(e)$. The objective is to maximize the value of a feasible multiflow, where the value of a multiflow x is $|x| = \sum_{P\in\mathcal{P}} x(P) = \sum_{(s,t)\in E(H)} |x^{st}|$. If $u = \vec{1}$ and we restrict to integral x, this problem is the *maximum edge-disjoint paths* problem (MEDP). A *weighted* maximization instance also comes equipped with per-unit profits $w : E(H) \to \mathbb{R}_{\geq 0}$, in which case the value of x is instead defined $|x|_w = \sum_{(s,t)\in E(H)} w(s,t)|x^{st}|$.

A *congestion instance* has the form (G, H, u, d), where $u : E(G) \to \mathbb{R}_{\geq 0}$ are capacities and $d : E(H) \to \mathbb{R}_{\geq 0}$ are *demand values*. A multiflow x is *feasible* for the instance if for every $e \in E(G)$ we have $x(e) \leq u(e)$ and for every $(s,t) \in E(H)$ we have $|x^{st}| = d(s,t)$. As a decision problem, the goal is to determine whether (G, H, u, d) has a feasible multiflow. As an optimization problem, the objective is to find the minimum $\alpha \geq 1$ such that $(G, H, \alpha u, d)$ has a feasible multiflow.

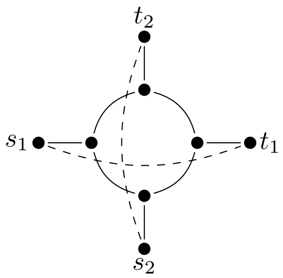
In either setting, we say an instance is *planar* if G is planar. We emphasize that $G + H$ is not necessarily planar. We always assume a fixed embedding of planar graph G.

2.2 Uncrossed Paths and Multiflows

First, we define what it means for paths to cross. Edge-disjoint paths can cross at a common node, but in general we need to consider common subpaths (see Figure 2).

▶ **Definition 9** (Crossing). *Let Z be a maximal shared subpath of paths P and Q that does not include any of their endpoints. Say the ends of Z are z and z'. The edges of $(P \cup Q) \setminus Z$ incident with z or z' have a clockwise ordering about Z in the plane. The paths* cross *at*

[6] Sometimes in the literature, (G, H) is called a *multiflow topology*. We avoid this term to avoid confusion with the topology of the plane.



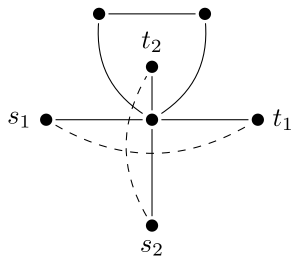


Figure 3 Problematic instances for uncrossability. Capacities and demand values are unit. (Left) A congestion instance with a feasible multiflow but no uncrossed multiflow. (Right) The *Keyhole Instance*, a congestion instance with a feasible uncrossed trail-multiflow but no uncrossed path-multiflow.

Z if the edges of P and Q alternate clockwise about Z. That is, if $P = P_1, e, Z, e', P_2$ and $Q = Q_1, g, Z, g', Q_2$, the edges occur e, g, e', g' or e, g', e', g clockwise about Z. Two paths are uncrossed *if they do not cross at any such subpath*[7].

The *support paths* of a multiflow x are the paths with $x(P) > 0$. The following is the natural extension of "uncrossed" to the multiflow setting (though we discuss a stronger condition in Section 2.4).

▶ **Definition 10** (Uncrossed Multiflow). *A multiflow is* uncrossed *if its support paths are pairwise uncrossed.*

We are interested in when a feasible multiflow x can be converted into an uncrossed one that routes the same demands. Our discussion here focuses on the congestion setting, but still applies to the maximization setting (since we can define target demand values d equal to whatever quantities x routes). As shown in Figure 3, even if (G, H, u, d) is feasible, it may not have an uncrossed feasible multiflow[8]. However several classes do admit uncrossed solutions. We say (G, H) is an *uncrossable layout* if for all u, d, if (G, H, u, d) has a feasible multiflow, then (G, H, u, d) has a feasible uncrossed multiflow. Matsumoto et al. [21] give an algorithm that, given a feasible fully-planar ($G + H$ planar) instance, produces a feasible uncrossed flow. Thus fully-planar layouts are uncrossable. It is also well-known that single-source layouts (H is a star) are uncrossable. Another uncrossable layout is given when G is a cycle with parallel edges and H is arbitrary, which we call *ring* layouts. In Section 3 we introduce *pairwise-planar* layouts and prove they are uncrossable.

2.3 Another View on Crossing: Parallelizations

There are more intuitive ways to think about uncrossed paths that we frequently use.

[7] We emphasize that paths can only cross at maximal common subpaths that do *not* contain their endpoints. Both paths must have edges before and after the common subpath in order to cross. For example, if P contains an endpoint of Q and Z is the maximal common subpath containing that endpoint, then P and Q do not cross at Z.

[8] In fact, it is not always possible to find an uncrossed flow with bounded congestion. One may modify the canonical grid instance [14] so it has a half-integral routing of demands, but any uncrossed multiflow incurs polynomial congestion.

Figure 4 (Left) An uncrossed parallelization of the uncrossed paths from Figure 2. (Right) Disk-expansion of the node z after uncrossed parallelization.

Widened Graph. Suppose we "widen" G, making its nodes into small disks and its edges into small channels. A family of pairwise uncrossed paths can have its paths drawn as curves simultaneously, each curve following the node disks and edge channels corresponding to its path, so that the curves are disjoint.

Parallelizations. To avoid dealing with the formal topology of widened graphs, we use an equivalent notion based on "separating" the paths at each edge. Consider a family of paths $\mathcal{P}$ and for each $e \in E(G)$ let $\mathcal{P}_e = \{P \in \mathcal{P} : e \in E(P)\}$. Let G' be obtained by replacing each e with $|\mathcal{P}_e|$ parallel copies of itself. Intuitively, the parallel edges represent the curves within the channel of e.

Now, for each $e \in E(G)$ we decide on some one-to-one mapping ϕ_e from $\mathcal{P}_e$ to the parallel class of e in G'. For each $P \in \mathcal{P}$, we obtain a path P_ϕ in G' whose edges are $\{\phi_e(P) : e \in P\}$. This induces a family of edge-disjoint paths $\mathcal{P}_\phi$ in G'. We call $\mathcal{P}_\phi$ a *parallelization* of $\mathcal{P}$. If we (arbitrarily) orient each $e \in E(G)$, it gives a linear ordering $\leq_e$ of $\mathcal{P}_e$. Different choices for $(\phi_e : e \in E(G))$ yield different parallelizations. The family $\mathcal{P}$ is pairwise uncrossed if and only if there is some parallelization that is pairwise uncrossed at every node of G' (rather than needing to consider longer shared subpaths). Using parallelizations helps us simplify subsequent definitions by assuming paths are edge-disjoint. Given a collection of paths, one can compute in polynomial time an uncrossed parallelization or find a crossing pair of paths.

Disk-Expansion. After parallelization, we sometimes go a step closer to the widened graph idea by "expanding" a node v. We draw a small disk with v at its centre (small enough that its boundary intersects only $\delta(v)$, and hits each edge once). Where an edge copy intersects the boundary, we place a new node. We terminate the edges at these new nodes. For an uncrossed parallelization, it is then possible to (simultaneously) draw edges in the open interior of the disk between nodes that correspond to consecutive edges of the same path, such that the drawing is planar. An example is shown in Figure 4.

2.4 Strongly Uncrossed Multiflows

Consider an uncrossed multiflow. For any $(s, t) \in E(H)$, its support paths have a clockwise ordering about s (and t) given by the parallelization. We call the regions of the plane bounded between two consecutive paths *sectors* for this commodity (defined more formally in Section 5).

The proof of Theorem 3 uses that sectors of different commodities are non-crossing as sets - that is, we want that if $A, B \subseteq \mathbb{R}^2$ are sectors of different commodities, then either $A \cap B = \emptyset$ or one contains the other. One might think this non-crossing property follows

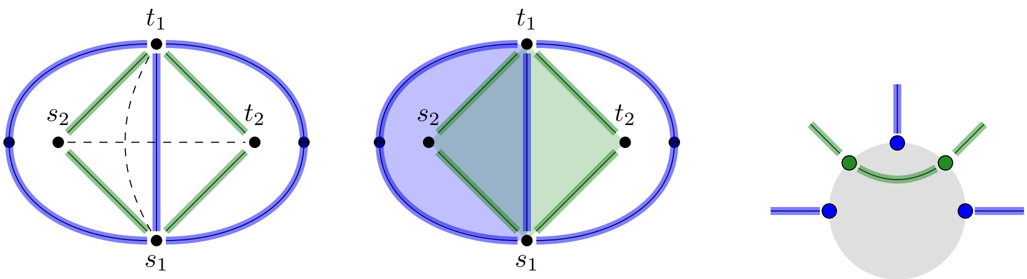


Figure 5 (Left) An uncrossed multiflow. The blue paths are for (s_1, t_1) and the green paths are for (s_2, t_2). No pair of support paths cross. (Center) A sector from the s_1t_1-flow (blue shaded region) that crosses a sector from the s_2t_2-flow (green shaded region). (Right) The disk-expansion of s_1. The green path separates the blue nodes in the disk. There is a quasicrossing at s_1.

from Definition 10, however Figure 5 shows a counterexample[9]. In the figure, no pair of flow paths cross, but the green s_2t_2-paths "slip between" the blue s_1t_1-paths to move between two distinct sectors. We define *strongly uncrossed* multiflows by imposing stricter restrictions on the support paths of different commodities at their terminals. Many layout classes admit strongly uncrossed flows, including fully-planar, single-source, and rings, as well as pairwise-planar.

▶ **Definition 11** (Quasicrossing, Strongly Uncrossed). *Let f be a multiflow with a fixed parallelization (so we assume edge-disjoint support paths). Consider a node z and the clockwise ordering of edges $e_1, \ldots, e_k$ about z. For a commodity h, a* type-1 (h, z)-interval *is $[i, j]$ (where $i < j$) such that e_i, e_j are consecutive edges of a single support path for h that transits through z. A* type-2 (h, z)-interval *is $[i, j]$ such that e_i, e_j both terminate (different) support paths for h at z. Note that two paths cross at z if and only if their corresponding type-1 intervals cross*[10]. *We say a* quasicrossing *occurs at z if there exists an (h, z)-interval I and an (h', z)-interval I' that cross where $h \neq h'$ and at least one of the intervals is type-2. If there are no crossings and no quasicrossings at any node, we say f is* strongly uncrossed.

2.5 Walk-Multiflows

It is standard for flows to only route on (simple) paths, since a flow walk could just be short-circuited into a flow path. Disturbingly, this operation is not benign in the uncrossed flow setting. The *Keyhole Instance* of Figure 3 is an example that requires the use of a non-simple flow trail to obtain an uncrossed solution. Since we would like to understand which instances have uncrossed solutions, we must consider *trail-multiflows* and *walk-multiflows* in addition to the usual *path-multiflows*. A *trail* can repeat nodes but not edges, while a *walk* can repeat both. Ultimately we can show our rounding results (Theorem 2, Theorem 3) and hardness results (Theorem 6, Theorem 7) presented earlier hold for all three models. The necessary modifications are minor and we defer the details to the full version.

---

[9] The desired non-crossing sector property does hold for uncrossed flows in *leaf instances*, where terminals all have degree 1. Moving terminals to leaves is a standard pre-processing operation, but is not safe for uncrossed flows - it may turn an uncrossable layout into one that is not. For example, ring instances admit uncrossed flows but Figure 3 (left) does not, even though it is obtained by adding leaves to a ring.
[10] Two intervals $[i, j], [i', j']$ cross if $i < i' < j < j'$ or $i' < i < j' < j$.

3 Uncrossed Flows in Pairwise-Planar Instances

Here we introduce a new class of multiflow layouts and prove they are uncrossable. We assume a fixed embedding of G. We say $(s,t) \in E(H)$ *fits* face F of G if both s,t lie in F (i.e. the demand could be drawn within the face). A layout (G,H) is *facial* if each (s,t) fits at least one face of G (but demands fitting a common face may still cross each other).

▶ **Definition 12.** *We say (G,H) is* pairwise-planar *if (G,H) is facial, and moreover for any pair $h_1, h_2 \in E(H)$, h_1 and h_2 can be embedded inside (possibly different) faces of G so they do not cross each other.*

Unlike fully-planar, we do not require simultaneous uncrossed embedding of all demands. This matters because a demand edge may fit multiple faces (a *multi-faced* demand). Figure 6 shows an example. However we emphasize that the embedding of G is fixed and does not change as we embed different pairs of $E(H)$.

Clearly pairwise-planar generalizes fully-planar, but how substantial is this generalization? We briefly argue it is a much richer class. First, it is easy to construct examples where H contains arbitrarily large cliques, meaning H and $G+H$ have arbitrarily large genus (though G remains planar). It also generalizes rings (G is a cycle with parallel edges, H is arbitrary). This is worth remarking because fully-planar does not capture ring layouts. More broadly, pairwise-planar captures what are known as *fully-compliant* layouts (under mild connectivity assumptions). Introduced in [7], in such layouts G is series-parallel and every $h \in E(H)$ satisfies that $G+h$ is series-parallel. In the full version, we argue the following.

▶ **Proposition 13.** *If (G,H) is fully-compliant and G is 2-connected, then (G,H) is pairwise-planar for any planar embedding of G.*

Pairwise-planar extends further than fully-planar and fully-compliant in another fundamental way through a property known as *cut-sufficiency*. For a multiflow instance (G,H,u,d), we say the *cut condition* holds if for every $S \subseteq V$, $u(\delta_G(S)) \geq d(\delta_H(S))$, where $\delta(S)$ denotes the edges with one end in S and one end in $V \setminus S$. The cut condition is necessary, but not in general sufficient, for the existence of a feasible (fractional) multiflow. A layout (G,H) is called cut-sufficient if, for all capacities and demands (u,d), the cut condition is sufficient for the existence of a feasible multiflow. The Max-Flow Min-Cut Theorem can be equivalently stated as: single-source (G,H) are cut-sufficient. Understanding the relationship between the cut condition and feasibility is a fundamental question in network flows, and cut-sufficient instances tend to admit stronger integrality properties and approximation algorithms. A result of Seymour [29] shows that fully-planar (G,H) are cut-sufficient, and Chekuri et al. [7] show that fully-compliant instances are cut-sufficient. However, pairwise-planar layouts need not be cut-sufficient. The quintessential examples of series-parallel layouts that are not cut-sufficient are the *odd spindles*[11] [6, 2], but these are pairwise-planar.

3.1 Uncrossing Procedure

In this section we show why feasible pairwise-planar instances have uncrossed flows. We focus on congestion instances (G,H,u,d) (where a feasible flow must satisfy all demands), since given a feasible flow x for a maximization instance (G,H,u), one can define demand

[11] In a *spindle*, $G = K_{2,p}$ where $p \geq 3$ and H consists of a simple p-cycle between the nodes in the larger side of the partition and an edge between the two nodes in the other side. A spindle is *odd* if p is odd. In some sense, the odd spindles characterize non-cut-sufficiency for series-parallel layouts, see [2].

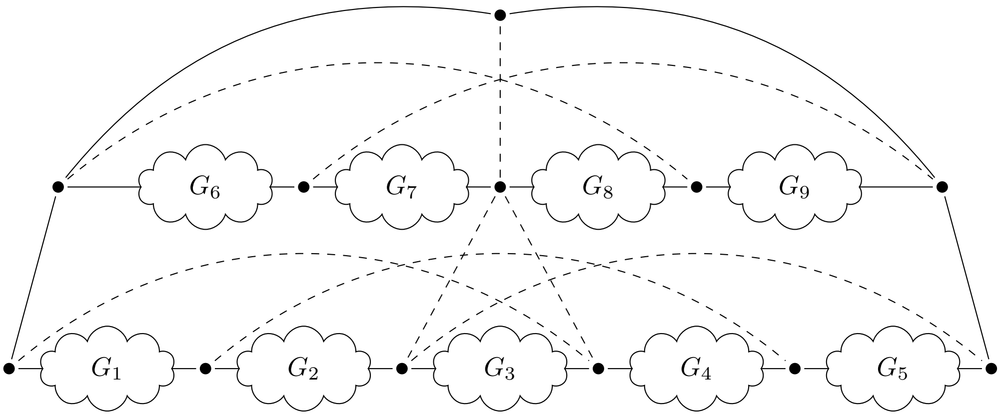


Figure 6 A pairwise-planar layout. Each G_i is a connected planar graph. The demand edges cannot be simultaneously embedded so that $G + H$ is planar ($G + H$ contains a $K_{3,3}$-minor and a K_5-minor). However, taking advantage of multi-faced demands, any *single pair* of demands can be embedded without crossing. One could make the layout more complicated by putting demands inside the G_i's following similar patterns.

values $d(s,t) = |x^{st}|$ and apply the congestion result. We present an inductive proof that a feasible uncrossed flow exists. It is straightforward to convert this into an algorithm, with one catch for polynomial time implementation that we explain at the end.

For fully-planar instances, [15, 12] use an algorithm by Matsumoto et al. [21] that builds an uncrossed flow from scratch starting from the zero flow. Our procedure has more in common with [18] (which concerns finding cycles in higher-genus surfaces that cross at most once) in that we begin with a (potentially highly crossing) feasible flow x and "uncross" it bit-by-bit. The following is an important lemma.

▶ **Lemma 14.** *Let (G, H, u, d) be a planar congestion instance that is feasible. There exists a feasible flow f^* such that every pair of support paths crosses at most once, and any pair of support paths that share a terminal do not cross.*

We say a flow satisfying this property is *almost-uncrossed*. Proving this lemma formally is somewhat tedious, but such arguments have appeared in the literature before - the essence of the argument is in Lemma 1 of [22] and a similar method is used in [18]. Loosely, the argument goes as follows. Begin with a feasible flow x. One can show that if a pair of support paths cross twice (or cross once but share a terminal), then one can re-route flow on their edges to decrease the "amount of crossing" between the two paths ("uncrossing" the two paths) and argue that this re-routing does not increase crossing with other support paths. Thus the total amount of crossing decreases, and we iterate until x is uncrossed. For completeness we give a careful proof of this lemma (and precisely define "amount of crossing") in the full version.

Lemma 14 immediately implies single-source layouts are uncrossable. With a small trick it also proves fully-planar layouts are uncrossable: augment the instance by adding leaves. That is, for each $h = (s,t) \in E(H)$, add nodes s_h, t_h into the interior of the face where h is embedded, add supply edges $(s_h, s), (t_h, t)$, move the endpoints of the demand h to (s_h, t_h), and extend the flow paths appropriately to route this instance. At first, adding leaves may actually increase the amount of crossing. However, since the demands do not cross, it can be shown that every pair of support paths that cross must cross at least twice. Then Lemma 14 allows one to conclude there exists a feasible uncrossed flow in the augmented instance. The leaves can then be contracted away without introducing any new crossing.

Extending this trick to pairwise-planar instances is not immediate. After adding leaves, it may not be the case that support paths that cross must cross at least twice. We cannot "commit" a demand to a particular face up front. Consider if a demand h fits into two different faces F_1 and F_2, but there are demands h_1 and h_2 that fit those respective faces and cross h in those faces. If we add leaves into F_1, then we can "uncross" support paths for h and h_2, however we cannot uncross support paths between h and h_1, and vice versa.

We take the following approach. Begin with a feasible flow z that we want to uncross. We temporarily "commit" each demand to one of the faces it fits (if multi-faced, choose arbitrarily). We accept for now that flow paths of demands committed to the same face may cross, but compute a flow x so that demands committed to different faces do not cross. In fact we aim for a more strict condition on x: we show how to find x so that, if we define x^F as the multiflow corresponding to demands committed to face F, the x^F's are "uncoupled" from each other. Here, "uncoupled" is a structural property (defined precisely below) which guarantees that if we can uncross each x^F separately, regardless of how we do so, we can still safely glue the resulting flows back together without introducing crossing. Then we can just focus on uncrossing each x^F inductively/recursively. We now make this "committing" process and "uncoupled" property precise.

▶ **Definition 15** (Facial Decomposition). *Let (G, H, u, d) be a facial multiflow instance. Let $\mathcal{F}$ be the set of faces of G. A* facial decomposition *is a collection $(H_F : F \in \mathcal{F})$ of subgraphs of H indexed by F such that:*

- *$(E(H_F) : F \in \mathcal{F})$ forms a partition of $E(H)$, and*
- *for every face $F \in \mathcal{F}$, every demand $h \in E(H_F)$ fits F (but cannot necessarily all be embedded simultaneously in F without crossing).*

For a facial decomposition $(H_F : F \in \mathcal{F})$ and a feasible flow x, we let x^F be the portion of x used to route H_F. Note that if $h \in E(H)$ is multi-faced, the facial decomposition still only puts h into one of the H_F's. Now we define the "uncoupled" condition and show that we can find a flow satisfying this property. We let $\text{supp}(x^F)$ denote the set of support paths of x^F, and let $G[\text{supp}(x^F)]$ denote the graph induced by the edges in those paths.

▶ **Definition 16** (Nested Supports Property). *Let (G, H, u, d) be a facial multiflow instance, and let $(H_F : F \in \mathcal{F})$ be a facial decomposition. For a flow x and a face F, we define the* stellated support graph *$G[x, F]$ as follows: begin with the support graph $G[\text{supp}(x^F)]$, add a node v_F into the interior of F, and then for each demand $(s, t) \in E(H_F)$ add edges (s, v_F) and (t, v_F), embedded (without crossing) in the face F. We say x satisfies the* nested supports property *if for every pair of distinct faces $F_1, F_2 \in \mathcal{F}$, the graph $G[x, F_1]$ is entirely contained within a single face of the graph $G[x, F_2]$, and vice versa.*

▶ **Lemma 17.** *Let (G, H, u, d) be a facial multiflow instance that is feasible. Let $(H_F : F \in \mathcal{F})$ be a facial decomposition. There exists a feasible flow x satisfying the nested supports property.*

The proof of this lemma is slightly tricky to make rigorous, so we include only a sketch here and defer details to the full version.

Proof Sketch of Lemma 17. We augment the instance by adding leaves into the faces: for every face F and demand $h = (s, t) \in E(H_F)$, add nodes s_h, t_h into the face F and supply edges $(s_h, s), (t_h, t)$ of capacity equal to $d(h)$, and move the demand h from (s, t) to (s_h, t_h). We keep the same facial decomposition. We now invoke Lemma 14 to obtain a feasible flow x that is almost-uncrossed for this new instance. It is not hard to show that for support paths P_1, P_2 associated with distinct faces, if we consider the embedded cycles $C(P_1), C(P_2)$ obtained by joining the endpoints of the paths with their respective demand edges, that these cycles do not cross (though cycles associated with paths of the same face may cross).

Now, for each face F, we let G_F be the union of $G[\text{supp}(x^F)]$ and all supply edges incident with the face F itself (regardless of whether they are in the support). It is possible to argue that for distinct faces F, F', G_F must be contained in a single face of $G_{F'}$, and vice versa. Once this property is established, we argue the nested supports property holds for flow x. ◀

Given such a flow, we show that we can focus on uncrossing each face's flow independently.

▶ **Lemma 18.** *Let (G, H, u, d) be a facial multiflow instance. Let $(H_F : F \in \mathcal{F})$ be a facial decomposition, and let x be a feasible flow satisfying the nested supports property. For every $F \in \mathcal{F}$, let u_F be the capacity vector defined by $u_F(e) = x^F(e)$ and let d_F be the demand vector d restricted to H_F. If for every $F \in \mathcal{F}$, the instance (G, H_F, u_F, d_F) has a feasible uncrossed solution y^F, then $y = \sum_{F \in \mathcal{F}} y^F$ is a feasible uncrossed solution for (G, H, u, d).*

Proof. The feasibility of y is clear from the construction of the subproblems. Also observe that, by construction, y still satisfies the nested supports property (since $G[y, F]$ is a subgraph of $G[x, F]$ for each face F of G).

We prove that y is uncrossed assuming each y^F is uncrossed. It remains to check for distinct faces F_1, F_2 and $P_1 \in \text{supp}(y^{F_1}), P_2 \in \text{supp}(y^{F_2})$, that P_1 and P_2 do not cross.

Consider two distinct faces F_1, F_2. Let $P_1 \in \text{supp}(y^{F_1}), P_2 \in \text{supp}(y^{F_2})$. Let $h_1 = (s_1, t_1), h_2 = (s_2, t_2)$ be the demands routed by P_1, P_2 respectively. Suppose P_1 and P_2 cross, for the sake of contradiction. Consider the (simple) cycle C in $G[y, F]$ obtained as $C = P_1 \bullet (s_1, v_{F_1}) \bullet (v_{F_1}, t_1)$, where $\bullet$ denotes concatenation. Since P_1 and P_2 cross, P_2 has some segment in the open exterior of C and some segment in the open interior of C. However, since C is a simple cycle, these two segments of P_2 must lie in distinct faces of $G[y, F]$, contradicting that y satisfies the nested supports property. ◀

Using Lemma 18, we are able to reduce to finding a feasible uncrossed flow for each instance (G, H_F, u_F, d_F) in isolation. No matter how we do this, we can safely combine the flows using the lemma. Importantly, to tackle the instance (G, H_F, u_F, d_F), we are allowed to re-commit multi-faced demands that were originally committed to F to other faces.

▶ **Theorem 19.** *Let (G, H) be pairwise-planar and let (G, H, u, d) be a congestion instance. If the instance is feasible, there exists a feasible uncrossed flow.*

Proof. We prove the result by induction on $|E(H)|$. If $|E(H)| = 1$ then the instance is single-source and the result follows, so suppose $|E(H)| \geq 2$. Let $\mathcal{H} = (H_F : F \in \mathcal{F})$ be an arbitrary facial decomposition.

We first consider the case that at least two faces F have that H_F is non-empty. We obtain a feasible flow x for (G, H, u, d) that satisfies the nested supports property for $\mathcal{H}$ by way of Lemma 17. For each face F, let u_F, d_F be defined as in the statement of Lemma 18. Each instance (G, H_F, u_F, d_F) is feasible via the flow x^F. Additionally, since there are at least two faces F such that H_F is non-empty, we have $|E(H_F)| < |E(H)|$ for every face F. Hence by induction, there is a feasible uncrossed flow y^F for each instance (G, H_F, u_F, d_F). Then by Lemma 18, the sum of these flows is a feasible uncrossed flow for (G, H, u, d).

Now, we consider the case where H_F is empty for all faces F except one, call it F^*. That is, $E(H) = E(H_{F^*})$, and in particular every demand fits F^*. If none of the demands cross within F^*, then the layout is fully-planar and we are done. Otherwise, there exist $h, h' \in E(H)$ that cross when drawn in the face F^*. However, since (G, H) is pairwise-planar, there exists another face $F' \neq F^*$ such that at least one of these two demands, without loss of generality say h', also fits F'. Now we define a facial decomposition $\widehat{\mathcal{H}} = (\widehat{H}_F : F \in \mathcal{F})$ as follows. We define $E(\widehat{H}_{F^*}) = E(H) \setminus \{h'\}$, $E(\widehat{H}_{F'}) = \{h'\}$, and $E(\widehat{H}_F) = \emptyset$ for all other faces F. That is, we now view h' as being associated with F' rather than F^*.

We obtain a feasible flow x for the instance (G, H, u, d) satisfying the nested supports property for this new facial decomposition $\widehat{\mathcal{H}}$ by way of Lemma 17. We define capacities u_F, d_F for each face F as in the statement of Lemma 18 using the new decomposition $\widehat{\mathcal{H}}$. Each instance $(G, \widehat{H}_F, u_F, d_F)$ is feasible. Additionally, each satisfies that $|E(\widehat{H}_F)| < |E(H)|$, and thus $(G, \widehat{H}_F, u_F, d_F)$ has a feasible uncrossed flow y^F by induction. Then the sum of these flows is a feasible uncrossed flow for (G, H, u, d) by Lemma 18. ◀

With a little extra effort, we can modify Lemma 18 and Theorem 19 to show that pairwise-planar instances admit strongly uncrossed flows.

It is straightforward to convert this inductive argument into a polynomial time algorithm, provided we have a polynomial time algorithm to compute a feasible almost-uncrossed flow, as in Lemma 14. However, this is somewhat tricky. To prove Lemma 14, we re-route flow on twice-crossing paths to reduce the total amount of crossing, but the amount of progress depends on the flow values of the crossing paths and could be very small. We defer the details to the full version, but one can efficiently compute an almost-uncrossed flow that is *approximately* feasible - for any $\epsilon > 0$, in polynomial time we can find an almost-uncrossed flow that is feasible for $(G, H, (1+\epsilon)u, d)$. The details are similar to the uncrossing algorithm in [18]. Essentially, by judiciously rounding each path's flow value, we ensure the values are common multiples of a small but sufficiently large fraction to ensure each uncrossing step makes significant progress. We summarize the final result below.

▶ **Theorem 20.** *Let (G, H) be pairwise-planar and let (G, H, u, d) be a congestion instance. Let $\epsilon > 0$ be any constant. If the instance is feasible, then there exists a strongly uncrossed flow f' that is feasible for $(G, H, (1+\epsilon)u, d)$. The flow can be computed in polynomial time.*

Note that $f'/(1+\epsilon)$ is feasible for $(G, H, u, d/(1+\epsilon))$, so this approximate guarantee is useful for maximization as well as congestion. Combining this theorem with our rounding results Theorem 2 and Theorem 3, we obtain Corollary 4.

3.2 Integral Flow-Multicut Gap

In this subsection we prove Corollary 5. Here we are concerned with maximization instances (G, H, u). A *multicut* is a set of supply edges $R \subseteq E(G)$ such that $G - R$ does not contain an st-path for any $(s, t) \in E(H)$. The minimum capacity of a multicut is at least the maximum value of a (fractional) multiflow. The ratio of these quantities is the *(fractional) flow-multicut gap*[12]. The *integral flow-multicut gap* is the ratio of the minimum capacity of a multicut to the maximum value of an integral multiflow.

An immediate corollary of Theorems 19 and 2 is that for pairwise-planar (G, H), the integral flow-multicut gap is bounded by $O(1)$ times the (fractional) flow-multicut gap. We combine this with the following result by Tardos and Vazirani [30] to conclude Corollary 5.

▶ **Theorem 21** (Tardos and Vazirani [30]). *For any maximization instance (G, H, u) where G is planar, the (fractional) flow-multicut gap is bounded by $O(1)$.*

[12] This is different from the similarly named *flow-cut gap* (cf. [7]), which is defined for congestion instances and relates the minimum congestion value to sparsity of cuts (not necessarily multicuts).

4 Maximization Setting: Rounding and Hardness

In this section, we prove Theorem 2 and Theorem 6, which cover our rounding and inapproximability results for maximum uncrossed multiflow.

4.1 Uncrossed String Graphs and Approximate Integer Decomposition

First we prove Theorem 2. Our approach is to establish a so-called approximate integer decomposition property [3, 5] for uncrossed multiflows. The outline of the strategy is as follows. Suppose we are given an uncrossed multiflow f, which achieves a value of β. Imagine scaling f by a positive integer k such that kf is integral. Of course, kf may not be feasible. However suppose we are able to show that kf can be decomposed as the sum of a small number b of feasible integral flows. That is $kf = \sum_{i=1}^{b} f^i$ where each f^i is integral and feasible. At least one of the f^i flows achieves a $1/b$ fraction of the value of kf, and hence a k/b fraction of the value of f.

How can we achieve such a decomposition? Suppose (for simplicity) that kf sends 1 unit of flow along paths $\mathcal{P} = \{P_1, \ldots, P_\ell\}$. Furthermore suppose that each edge of G has unit capacity and hence each appears on at most k paths. Our goal then is to partition the paths into classes such that every edge appears at most once in each class; the classes then define the flows f^i. Equivalently, we seek to partition the paths so that paths in the same class do not share an edge. Phrased this way, we can recognize this as a graph colouring problem on the edge-intersection graph of $\mathcal{P}$.

The edge-intersection graph U of $(G, \mathcal{P})$ has node set $\mathcal{P}$, and an edge between distinct $P_1, P_2 \in \mathcal{P}$ if there exists $e \in E(P_1) \cap E(P_2)$. The *load* of $e \in E(G)$ is the number of paths in which e occurs, and the *load* of $(G, \mathcal{P})$ is the maximum load of an edge $e \in E(G)$. Graphs U constructed in this way are a special type of the well-studied family of *string graphs*, which are the intersection graphs of curves in the plane.

▶ **Definition 22** (Uncrossed String Graph, Realization). *An* uncrossed string graph *U is the edge-intersection graph of a family of uncrossed paths $\mathcal{P}$ in a planar graph G. We say $(G, \mathcal{P})$ is a* realization *of U.*

Uncrossed string graphs have been studied before for their own sake [11, 32]. Given that U has a load k realization, we want that its chromatic number $\chi(U)$ is some small function of k, ideally $O(k)$. This question is strongly related to the concept of a χ-*bounded* graph family. A family of graphs $\mathcal{G}$ is χ-bounded if there exists a function f such that for all $U \in \mathcal{G}$, $\chi(U) \leq f(\omega(U))$, where $\omega(U)$ is the clique number of U. It is worth noting that general string graphs are not χ-bounded, and it is not obvious that uncrossed string graphs should be. An $O(k)$ bound was conjectured originally in [11], and proven by Fox and Pach [13].

▶ **Theorem 23** (Fox and Pach). *Let U be an uncrossed string graph with realization $(G, \mathcal{P})$ such that each edge $e \in E(G)$ has load at most k. Then $\chi(U) \leq 6ek + 1$.*

The result of Fox and Pach has not been published, however the proof is elegant and short. It establishes via a probabilistic proof (similar to a standard proof giving a bound on the crossing number of a graph) that U has at most $3ek|\mathcal{P}|$ edges. Hence a greedy colouring procedure that removes the minimum degree vertex (which has degree at most $6ek$) and colours the rest of the graph inductively obtains a $6ek + 1$ colouring. We were not initially aware of Fox and Pach's result and independently proved a weaker bound that $\chi(U) \leq O(k^2 \log |\mathcal{P}|)$. For the sake of completeness we present this proof in the full version.

We now use the colouring result to establish an approximate integer decomposition property and prove a generalization of Theorem 2 with weighted profits on $E(H)$.

▶ **Theorem 24.** *There exists a universal constant $\gamma \geq 1$ such that the following holds. Let (G, H, u) be a planar maximization instance where u is integral, equipped with per-unit profits $w(h) \geq 0$ for $h \in E(H)$. Let f be a feasible uncrossed multiflow. There exists a feasible integral multiflow f^* that is uncrossed and achieves value $|f^*|_w \geq \frac{1}{\gamma}|f|_w$. This solution can be computed in polynomial time.*

Proof. We first present the case where the capacities are unit-valued $u = \vec{1}$. We show how to handle general capacities in the full version - while not hard, it requires some care to handle polynomial time aspects.

Let $\beta = |f|_w$. Let $\mathcal{P}$ be the family of support paths for f. Since $u = \vec{1}$, we have that $f(P) < 1$ for each $P \in \mathcal{P}$. Now, let k be a positive integer such that kf is integral. Since kf is integral, for each P we have that $f(P) = \alpha_P/k$ where $\alpha_P \leq k$ is a positive integer. By treating each P as α_P distinct paths, we may assume that each path routes exactly $\frac{1}{k}$ units of flow. Then, since $\sum_{P:e\in P} f(P) \leq 1$ by feasibility, at most k paths use each edge of G.

Let U be the edge-intersection graph of $(G, \mathcal{P})$, which has load at most k. By Theorem 23, U is b-colourable where $b = O(k)$. Now for each colour class i, we let f_i denote the flow obtained by sending one unit of flow on each path in that colour class, and zero flow on all other paths. Then f_i is integral and satisfies that $\sum_{P:e\in P} f_i(P) \leq 1$ for each $e \in E(G)$, and $kf = \sum_{i=1}^{b} f_i$ since each path's flow was blown up by a factor of k. Since the value of kf is $k\beta$, at least one of the f_i flows has a value of $k\beta/b$. Since $b = O(k)$, then this flow has value $\Omega(1)$ times that of f, as desired.

However, this proof is existential thus far, and does not immediately give a polynomial time algorithm because k could be very large. To fix this, we first have a pre-processing step where we convert f into a feasible flow $\hat{f}$ that uses a subset of f's support paths (so that it remains uncrossed) such that each value $\hat{f}(P)$ is a multiple of $\frac{1}{\mathrm{poly}(n)}$, and so that $|\hat{f}|_w$ is a large fraction of $|f|_w$. We then run the above procedure on $\hat{f}$ instead of f, giving a polynomial time algorithm. We prove the following.

▶ **Lemma 25.** *Let $\epsilon \in (0, 1)$ be any constant. There exists a positive constant N_ϵ such that the following holds. Let $(G, H, u = \vec{1})$ be a unit-capacity planar maximization instance on $n \geq N_\epsilon$ nodes, equipped with per-unit profits $w(h) \geq 0$ for $h \in E(H)$. Let f be a feasible multiflow such that $|f|_w \geq w_{\max} = \max_{h\in E(H)} w(h)$. There exists a multiflow $\hat{f}$ satisfying:*

1. *$\hat{f}$ is feasible,*
2. *the support paths of $\hat{f}$ are a subset of the support paths of f,*
3. *$|\hat{f}|_w \geq (1 - \epsilon)|f|_w$, and*
4. *for every path P, $\hat{f}(P)$ is a multiple of $\frac{1}{\log^2 n}$.*

There is a polynomial time algorithm that computes such an $\hat{f}$.

We prove this in the full version using a randomized rounding and re-scaling argument (which can be de-randomized to give a deterministic algorithm). The two conditions that $|f|_w \geq w_{\max}$ and $n \geq N_\epsilon$ are only minor technicalities, as we argue in that version. ◀

It is straightforward to generalize this to the node-capacitated setting - the colouring and integer decomposition property proceed in exactly the same way.

4.2 Inapproximability of Maximum Uncrossed Fractional Multiflow

Next we prove Theorem 6 on the inapproximability of maximum uncrossed (fractional) multiflow. We make use of a strong inapproximability result given by Chuzhoy, Kim, and Nimavat for maximum edge-disjoint paths (MEDP) and maximum node-disjoint paths (MNDP) in planar graphs; i.e. maximum integer multiflow for unit capacities. One minor technicality is that in their formulation, a feasible solution is only allowed to route one unit of flow for each demand. We call these problems *capped MEDP* and *capped MNDP* respectively.

▶ **Theorem 26** (Theorems 1.1 and 1.2, [9])**.** *For every constant $\epsilon > 0$, there is no $2^{\Omega(\log^{1-\epsilon} n)}$-approximation polynomial-time algorithm for capped MEDP or capped MNDP, assuming that $NP \not\subseteq DTIME(n^{polylog\ n})$. Moreover, there is no $n^{O\left(\frac{1}{(\log\log n)^2}\right)}$-approximation polynomial-time algorithm for capped MEDP or capped MNDP, assuming that for some constant $\delta > 0$, $NP \not\subseteq DTIME(2^{n^{\delta}})$. These results hold even when the input graph is a maximum degree 3 planar graph, or even if G is a wall graph.*

We now prove Theorem 6 which relies on this result. We focus on the edge-capacitated version (the node-capacitated version is similar).

Proof of Theorem 6. We give a polytime reduction from capped MEDP in wall graphs to maximum uncrossed multiflow in wall graphs, that preserves the approximation ratio up to a constant factor. For an instance (G, H) of the capped MEDP problem where G is a wall graph (which is planar and has maximum degree 3), define the following quantities.

- Let $D_E(G, H)$ denote the maximum number of demands in H which can be routed on edge-disjoint paths in G (i.e. the optimal value for capped MEDP).
- Let $U_E(G, H)$ denote the maximum value of a (fractional) uncrossed multiflow for $(G, H, u = \vec{1})$.

Observe that any capped MEDP solution for (G, H) is a feasible (integral) uncrossed multiflow for $(G, H, u = \vec{1})$. To see this, consider a collection of edge-disjoint paths that form a capped MEDP solution. Recall from Section 2.2 that two edge-disjoint paths cross only if they cross at a node (rather than a longer common subpath). Since G has maximum degree 3, a node crossing is not possible. Thus we have $D_E(G, H) \leq U_E(G, H)$.

Suppose we have a β-approximation algorithm for maximum uncrossed (fractional) multiflow in wall graphs. We obtain an approximation algorithm for capped MEDP as follows. We run our β-approximation algorithm for maximum uncrossed multiflow on $(G, H, u = \vec{1})$ to obtain flow f whose value is at least $U_E(G, H)/\beta$. We apply the polytime algorithm from Theorem 24 to compute a feasible uncrossed integral multiflow f' whose value is $1/\gamma$ that of f. It may still not be a valid capped MEDP solution, since it may route more than one flow path for the same commodity. However, since G has maximum degree 3, it routes on at most 3 paths for each commodity, so by selecting exactly one path for each commodity we obtain a capped MEDP solution whose value is at least $\frac{1}{3}$ that of f'. Thus this algorithm outputs a capped MEDP solution whose value is at least $\frac{U_E(G,H)}{3\gamma\beta} \geq \frac{D_E(G,H)}{3\gamma\beta}$, and is thus a $3\gamma\beta$-approximation algorithm. ◀

5 Congestion Setting: Rounding

In this section, we focus on planar congestion instances (G, H, u, d) where the demands must be fully routed. We show Theorem 3, which states that we can convert (fractional) strongly uncrossed multiflows into integral flows that violate the capacities by a factor of at most 2, and if we accept an additional additive error of less than $d_{\max} = \max_{h \in E(H)} d(h)$ the flow can be made unsplittable. Section 5.1 presents the relevant properties of strongly uncrossed flows, and Section 5.2 gives the proof of Theorem 3.

Figure 7 (Left) The disk expansion of a node z. Paths of the same colour are for the same commodity. The blue and red paths terminate at z, while the green path transits through z. There is no crossing or quasicrossing at z. (Right) We add a node into the disk for each commodity terminating there (blue, red) and connect the new node to the endpoints of that commodity's support paths. Since there is no crossing or quasicrossing at z, this operation can be done without causing edge crossings.

5.1 Clockwise Orderings and Sectors

Let f be a strongly uncrossed multiflow for planar congestion instance (G, H, u, d). Let $\mathcal{P}$ be the set of support paths, and let $\mathcal{P}^h$ be the support paths routing demand $h \in E(H)$. Fix a strongly uncrossed parallelization ϕ. We parallelize the edges and perform disk-expansion at each node (see Section 2.2) so the paths become node-disjoint. Then, for each commodity $h = (s, t) \in E(H)$, we add a node and edges into the disk corresponding to s to join the endpoints of the paths for h, and similarly for the disk of t. This can be done in a planar fashion for all demands simultaneously since the flow is strongly uncrossed (see Section 2.4 and Figure 7). We denote the new graph G_ϕ. For $P \in \mathcal{P}$, we let $\widehat{P}$ be its corresponding path in G_ϕ. For $P, Q \in \mathcal{P}^h$, $\widehat{P}$ and $\widehat{Q}$ are internally node-disjoint paths that share their endpoints. For $P, Q \in \mathcal{P}$ routing different demands, $\widehat{P}$ and $\widehat{Q}$ are (completely) node-disjoint.

Consider a commodity $h = (s, t) \in E(H)$. We use the notation $G_\phi[h]$ to mean the subgraph of G_ϕ induced by the paths $\widehat{\mathcal{P}}^h = \left\{\widehat{P} : P \in \mathcal{P}^h\right\}$. In G_ϕ, the paths of $\widehat{\mathcal{P}}^h$ have a clockwise cyclic ordering about s, say $\widehat{P}_1, \ldots, \widehat{P}_k$. Informally, a *sector* is the region between consecutive paths in the clockwise ordering. We make this more precise as follows.

Consider two paths $\widehat{P}_i$ and $\widehat{P}_j$ in G_ϕ. Orient $\widehat{P}_i$ from s to t, and orient $\widehat{P}_j$ from t to s. The directed curve $\widehat{P}_i \bullet \widehat{P}_j$, obtained by concatenating these oriented curves, is a simple closed (Jordan) curve, and partitions the plane into two regions. We define $\mathrm{CW}^\circ(\widehat{P}_i, \widehat{P}_j)$ to be the set of points q in $\mathbb{R}^2 \setminus (\widehat{P}_i \cup \widehat{P}_j)$ such that q is "to the right" of $\widehat{P}_i \bullet \widehat{P}_j$; that is, if we draw a directed curve from q to $\widehat{P}_i \bullet \widehat{P}_j$ (avoiding $\widehat{P}_i \bullet \widehat{P}_j$ except at the extremity), it hits $\widehat{P}_i \bullet \widehat{P}_j$ from the right. We let $\mathrm{CW}(\widehat{P}_i, \widehat{P}_j)$ denote the closed region $\mathrm{CW}^\circ(\widehat{P}_i, \widehat{P}_j) \cup \widehat{P}_i \cup \widehat{P}_j$. We analogously define $\mathrm{ACW}^\circ(\widehat{P}_i, \widehat{P}_j)$ and $\mathrm{ACW}(\widehat{P}_i, \widehat{P}_j)$ by taking the opposite orientation. Any curve that does not cross and is internally disjoint from $\widehat{P}_i$ and $\widehat{P}_j$ either stays entirely within $\mathrm{CW}(\widehat{P}_i, \widehat{P}_j)$ or stays entirely within $\mathrm{ACW}(\widehat{P}_i, \widehat{P}_j)$.

Now, for our clockwise-ordered family of paths $\widehat{P}_1, \ldots, \widehat{P}_k$, we refer to $\mathrm{CW}^\circ(\widehat{P}_i, \widehat{P}_{i+1})$ for $i = 1, \ldots, k$ (addition modulo k) as the *open sectors* of f^h, and define the *closed sectors* similarly using $\mathrm{CW}(\widehat{P}_i, \widehat{P}_{i+1})$. The open sectors partition $\mathbb{R}^2 \setminus (\widehat{P}_1 \cup \cdots \cup \widehat{P}_k)$. One sector contains the infinite face of $G_\phi[h]$; we call this the *outer sector* of h, and the others the *inner sectors* of h. The *outer paths* of h are the two paths $P, Q \in \mathcal{P}^h$ whose corresponding paths $\widehat{P}, \widehat{Q} \in \widehat{\mathcal{P}}^h$ define the outer sector of h.

Now consider an edge e of the original graph G that appears in $\mathcal{P}^h$. If e lies in any outer path of $\mathcal{P}^h$, it is an *outer edge* of h, otherwise it is an *inner edge* of h. For our congestion result, we exploit the following key properties.

▶ **Lemma 27.** *Let f be a strongly uncrossed multiflow, with a fixed strongly uncrossed parallelization ϕ. For distinct $h, h' \in E(H)$, the sets of inner edges of h and h' are disjoint.*

Proof. To prove the result, we show that if an edge e is shared between the two single-commodity flows in G, then it is an outer edge for one of the commodities.

▷ Claim 28. $G_\phi[h']$ is entirely contained within a single sector of h, and $G_\phi[h]$ is entirely contained within a single sector of h'.

Proof. By symmetry, it suffices to prove the first part. Recall that in G_ϕ, two paths for distinct commodities are completely disjoint, so each individual $\widehat{P} \in \widehat{\mathcal{P}}^{h'}$ lies in at most one sector of h. However, all paths of $\widehat{\mathcal{P}}^{h'}$ share common terminals in G_ϕ, so it must be the same sector for all of $\widehat{\mathcal{P}}^{h'}$. ◁

▷ Claim 29. Either $G_\phi[h']$ is contained in the outer sector of h, or $G_\phi[h]$ is contained in the outer sector of h'.

Proof. By the previous claim and by symmetry, it suffices to show that if $G_\phi[h']$ is contained within an inner sector of h, then $G_\phi[h]$ is contained within the outer sector of h'. Suppose $G_\phi[h']$ is in the inner sector of h defined by paths $\widehat{P}_i$ and $\widehat{P}_{i+1}$. Since $\mathrm{CW}(\widehat{P}_i, \widehat{P}_{i+1})$ is an inner sector, it is bounded, which means $G_\phi[h']$ is contained in the interior of the curve $\widehat{P}_i \bullet \widehat{P}_{i+1}$ rather than the exterior. This implies that $\widehat{P}_i$ and $\widehat{P}_{i+1}$ are contained within the infinite face of $G_\phi[h']$, and so therefore are in the outer sector of h'. Since the entirety of $G_\phi[h]$ is within the same sector of h', it is within the outer sector of h'. ◁

Now, without loss of generality, suppose that $G_\phi[h]$ is contained within the outer sector of h'. Then $G_\phi[h]$ and $G_\phi[h']$ both contain one of the copies of the shared edge e. The copy belonging to h is contained in the infinite face of $G_\phi[h']$, and (by how G_ϕ is constructed) so too is the copy belonging to h'. So then e is an outer edge of h', not an inner edge. ◀

Now, associating the circular ordering $P_1, \ldots, P_k$ to $\mathcal{P}^h$ corresponding to the clockwise ordering $\widehat{P}_1, \ldots, \widehat{P}_k$ of $\widehat{\mathcal{P}}^h$, we prove the second key property.

▶ **Lemma 30.** *Suppose that in G, $P_i, P_j \in \mathcal{P}^h$ share an edge e. One of the following holds:*
- *every path between P_i and P_j in the circular ordering also uses e, or*
- *every path between P_j and P_i in the circular ordering also uses e.*

Proof. In G_ϕ, consider subdividing the copies of e belonging to $\widehat{P}_i$ and $\widehat{P}_j$ by adding one node into each edge. Then add an edge e' between the two new nodes, adding e' such that it only crosses copies of e. The new edge either separates $\mathrm{CW}(\widehat{P}_i, \widehat{P}_j)$ or it separates $\mathrm{ACW}(\widehat{P}_i, \widehat{P}_j)$, where by separates we mean that any path of $\widehat{\mathcal{P}}^h$ that is within the sector must cross e'. The paths between $\widehat{P}_i$ and $\widehat{P}_j$ in the clockwise ordering all lie in $\mathrm{CW}(\widehat{P}_i, \widehat{P}_j)$, and the paths between $\widehat{P}_j$ and $\widehat{P}_i$ in the clockwise ordering all lie in $\mathrm{ACW}(\widehat{P}_i, \widehat{P}_j)$. If a path $\widehat{P}_r$ crosses e', then it means that it uses a copy of e, and hence its corresponding path P_r in G also uses e. The result follows. ◀

5.2 Main Congestion Argument

We are now ready to prove Theorem 3.

Proof of Theorem 3. First, we show how to prove Item 2 (unsplittable flow), since Item 1 ultimately follows from it. Let f be a feasible strongly uncrossed flow with a fixed strongly uncrossed parallelization ϕ. Recall that we let $f(e) = \sum_{P:e\in P} f(P)$ and let f^h be the single-commodity flow for $h \in E(H)$.

Consider a commodity h. Let $P_1, \ldots, P_k$ be the paths of $\mathcal{P}^h$, ordered according to the cyclic clockwise ordering of their corresponding paths $\widehat{P}_1, \ldots, \widehat{P}_k \in \widehat{\mathcal{P}}^h$ in G_ϕ. Re-index so that P_1 and P_k are the outer paths. Let i^* be the smallest index i such that $\sum_{j \le i} f^h(P_j) \ge \frac{d(h)}{2}$. We define $Q^h = P_{i^*}$, and call it the *central path* for h.

▷ Claim 31. If $e \in Q^h$ and e is an outer edge for h (i.e. it lies in P_1 or P_k), then $f^h(e) \ge \frac{d(h)}{2}$.

Proof. Since e is an outer edge, it is in either P_1 or P_k, in addition to $Q^h = P_{i^*}$. From Lemma 30 we have that $e \in P_r$ either for all $1 \le r \le i^*$ or for all $i^* \le r \le k$. In the former case, $f^h(e) \ge \sum_{1 \le r \le i^*} f^h(P_r) \ge \frac{d(h)}{2}$. For the latter case, $f^h(e) \ge \sum_{i^* \le r \le k} f^h(P_r) = d(h) - \sum_{1 \le r < i^*} f^h(P_r) > d(h) - \frac{d(h)}{2} = \frac{d(h)}{2}$, where the last inequality follows since $\sum_{1 \le r < i^*} f^h(P_r) < \frac{d(h)}{2}$ by definition of i^*. ◁

To obtain an unsplittable flow f', for each $h \in E(H)$ we route $d(h)$ units of flow along the path Q^h. For the congestion bound, consider an arbitrary edge $e \in E(G)$. We will bound the amount of flow through e. Let $H_e = \{h \in E(H) : e \in Q^h\}$, so $f'(e) = \sum_{h \in H_e} d(h)$. Define $O_e = \{h \in H_e : e \text{ is an outer edge of } h\}$ and also $I_e = \{h \in H_e : e \text{ is an inner edge of } h\}$. These sets partition H_e, so $f'(e) = \sum_{h \in O_e} d(h) + \sum_{h \in I_e} d(h)$. We know that $|I_e| \le 1$ from Lemma 27. Additionally, by Claim 31, for each $h \in O_e$ we have that $f^h(e) \ge \frac{d(h)}{2}$. We now split into two cases.

Case 1: $\sum_{h \in O_e} \frac{d(h)}{2} < u(e)$. Then,

$$f'(e) < 2u(e) + \sum_{h \in I_e} d(h) \le 2u(e) + d_{\max}.$$

Case 2: $\sum_{h \in O_e} \frac{d(h)}{2} \ge u(e)$. Then $\sum_{h \in O_e} f^h(e) \ge \sum_{h \in O_e} \frac{d(h)}{2} \ge u(e)$, so the commodities of O_e saturate e in f. It follows that $I_e = \emptyset$. Hence,

$$f'(e) = \sum_{h \in O_e} d(h) \le \sum_{h \in O_e} 2f^h(e) \le 2f(e) \le 2u(e) < 2u(e) + d_{\max}.$$

This concludes the proof of Item 2. The above is easily turned into a polynomial time algorithm. We show how to prove Item 1 from Item 2 in the full version. ◀

References

1. Cédric Bentz. Disjoint paths in sparse graphs. *Discrete Applied Mathematics*, 157(17):3558–3568, October 2009. doi:10.1016/j.dam.2009.03.009.
2. Amit Chakrabarti, Lisa Fleischer, and Christophe Weibel. When the cut condition is enough: A complete characterization for multiflow problems in series-parallel networks. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 19–26. ACM, 2012. doi:10.1145/2213977.2213980.
3. Chandra Chekuri, Marcelo Mydlarz, and F Bruce Shepherd. Multicommodity demand flow in a tree and packing integer programs. *ACM Transactions on Algorithms (TALG)*, 3(3):27, 2007. doi:10.1145/1273340.1273343.
4. Chandra Chekuri, Guyslain Naves, and F Bruce Shepherd. Maximum edge-disjoint paths in k-sums of graphs. In *International Colloquium on Automata, Languages, and Programming*, pages 328–339. Springer, 2013. doi:10.1007/978-3-642-39206-1_28.
5. Chandra Chekuri and F Bruce Shepherd. Approximate integer decompositions for undirected network design problems. *SIAM Journal on Discrete Mathematics*, 23(1):163–177, 2009. doi:10.1137/040617339.

6 Chandra Chekuri, F Bruce Shepherd, and Christophe Weibel. Flow-cut gaps for integer and fractional multiflows. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1198–1208. Society for Industrial and Applied Mathematics, 2010. doi:10.1137/1.9781611973075.96.

7 Chandra Chekuri, F Bruce Shepherd, and Christophe Weibel. Flow-cut gaps for integer and fractional multiflows. *Journal of Combinatorial Theory, Series B*, 103(2):248–273, 2013. doi:10.1016/j.jctb.2012.11.002.

8 Julia Chuzhoy, David HK Kim, and Rachit Nimavat. Almost polynomial hardness of node-disjoint paths in grids. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1220–1233, 2018.

9 Julia Chuzhoy, David Hong Kyun Kim, and Rachit Nimavat. Almost polynomial hardness of node-disjoint paths in grids. *Theory of Computing*, 17(6):1–57, 2021. doi:10.4086/toc.2021.v017a006.

10 Denis Cornaz. Max-multiflow/min-multicut for G + H series-parallel. *Discrete Mathematics*, 311(17):1957–1967, September 2011. doi:10.1016/j.disc.2011.05.025.

11 Louis Esperet, Daniel Gonçalves, and Arnaud Labourel. Coloring a set of touching strings. *Electronic Notes in Discrete Mathematics*, 34:213–217, August 2009. doi:10.1016/j.endm.2009.07.035.

12 David Alemán Espinosa, Nikhil Kumar, Joseph Poremba, and Bruce Shepherd. Unsplittable flow cut gap in undirected graphs. In *Proceedings of the 2026 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1570–1605. SIAM, 2026. doi:10.1137/1.9781611978971.57.

13 Jacob Fox and János Pach. Touching Strings. Unpublished manuscript, 2012.

14 N. Garg, V. V. Vazirani, and M. Yannakakis. Primal-dual approximation algorithms for integral flow and multicut in trees. *Algorithmica*, 18(1):3–20, May 1997. doi:10.1007/BF02523685.

15 Naveen Garg, Nikhil Kumar, and András Sebő. Integer plane multiflow maximisation: one-quarter-approximation and gaps. *Mathematical Programming*, 195(1):403–419, 2022. doi:10.1007/S10107-021-01700-8.

16 Michel X Goemans and David P Williamson. Primal-dual approximation algorithms for feedback problems in planar graphs. *Combinatorica*, 1(18):37–59, 1998. doi:10.1007/PL00009810.

17 Chien-Chung Huang, Mathieu Mari, Claire Mathieu, Kevin Schewior, and Jens Vygen. An approximation algorithm for fully planar edge-disjoint paths. *SIAM Journal on Discrete Mathematics*, 35(2):752–769, 2021. doi:10.1137/20M1319401.

18 Chien-Chung Huang, Mathieu Mari, Claire Mathieu, and Jens Vygen. Approximating maximum integral multiflows on bounded genus graphs. *Discrete & Computational Geometry*, 70(4):1266–1291, 2023. doi:10.1007/S00454-023-00552-7.

19 Philip Klein, Serge A. Plotkin, and Satish Rao. Excluded minors, network decomposition, and multicommodity flow. In *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*, STOC '93, pages 682–690, New York, NY, USA, 1993. ACM. doi:10.1145/167088.167261.

20 Bernhard Korte, Làszlo Lovàsz, Hans Jürgen Prömel, and Alexander Schrijver. *Paths, flows, and VLSI-layout.* Springer-Verlag, 1990.

21 Kazuhiko Matsumoto, Takao Nishizeki, and Nobuji Saito. Planar multicommodity flows, maximum matchings and negative cycles. *SIAM Journal on Computing*, 15(2):495–510, 1986. doi:10.1137/0215034.

22 Guyslain Naves. The hardness of routing two pairs on one face. *Mathematical Programming*, 131(1-2):49–69, February 2012. doi:10.1007/s10107-010-0343-0.

23 Janos Pách. personal communication.

24 Niklas Schlomberg. An improved integrality gap for disjoint cycles in planar graphs. *arXiv preprint*, 2024. arXiv:2404.17813.

25 Niklas Schlomberg, Hanjo Thiele, and Jens Vygen. Packing cycles in planar and bounded-genus graphs. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2069–2086. SIAM, 2023. `doi:10.1137/1.9781611977554.CH79`.

26 Alexander Schrijver. Homotopic routing methods. *Paths, Flows, and VLSI-layout*, pages 329–371, 1990.

27 Alexander Schrijver. Disjoint homotopic paths and trees in a planar graph. *Discrete & Computational Geometry*, 6:527–574, 1991. `doi:10.1007/BF02574704`.

28 Alexander Schrijver. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer, 2003.

29 Paul D Seymour. Matroids and multicommodity flows. *European Journal of Combinatorics*, 2(3):257–290, 1981. `doi:10.1016/S0195-6698(81)80033-9`.

30 Éva Tardos and Vijay V. Vazirani. Improved bounds for the max-flow min-multicut ratio for planar and K-free graphs. *Information Processing Letters*, 47(2):77–80, August 1993. `doi:10.1016/0020-0190(93)90228-2`.

31 Vera Traub, Laura Vargas Koch, and Rico Zenklusen. Single-source unsplittable flows in planar graphs. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 639–668. SIAM, 2024. `doi:10.1137/1.9781611977912.24`.

32 Wouter Cames Van Batenburg, Louis Esperet, and Tobias Müller. Coloring Jordan Regions and Curves. *SIAM Journal on Discrete Mathematics*, 31(3):1670–1684, January 2017. `doi:10.1137/16M1092726`.

33 Jens Vygen. Np-completeness of some edge-disjoint paths problems. *Discrete Applied Mathematics*, 61(1):83–90, 1995. `doi:10.1016/0166-218X(93)E0177-Z`.