Approximating Maximum Integral Multiflows on Bounded Genus Graphs

Chien-Chung Huang ✉
CNRS, ENS, PSL, Paris, France

Mathieu Mari ✉
University of Warsaw, Poland

Claire Mathieu ✉
CNRS, IRIF, Université de Paris, France

Jens Vygen ✉
Research Institute for Discrete Mathematics & Hausdorff Center for Mathematics,
University of Bonn, Germany

─── **Abstract** ───

We devise the first constant-factor approximation algorithm for finding an integral multi-commodity flow of maximum total value for instances where the supply graph together with the demand edges can be embedded on an orientable surface of bounded genus. This extends recent results for planar instances. Our techniques include an uncrossing algorithm, which is significantly more difficult than in the planar case, a partition of the cycles in the support of an LP solution into free homotopy classes, and a new rounding procedure for freely homotopic non-separating cycles.

1 Introduction

Multi-commodity flows, or *multiflows* for short, are well-studied objects in combinatorial optimization; see, e.g., Part VII of [34]. A multiflow of maximum total value can be found in polynomial time by linear programming. Often, a multiflow must be integral, and then the problem is much harder; the well-known edge-disjoint paths problem is a special case. Recently, constant-factor approximation algorithms have been found for maximum edge-disjoint paths and integral multiflows in *fully planar instances*, i.e., when $G + H$, the supply graph together with the demand edges, can be embedded in the plane [21, 16]. We generalize these results to surfaces of bounded genus and devise the first constant-factor approximation algorithm for that case.

Beyond using some ideas of [16, 21], we need several new ingredients. Like [16], we start by computing an optimal (fractional) multiflow and "uncross" the cycles in its support as much as possible, but uncrossing is significantly more complicated on general surfaces than in the plane. Next, we need to deal with two cases separately: depending on whether most of the fractional multiflow is on separating cycles (that case is similar to the planar case)

or on non-separating cycles. In the latter case we partition the cycles into free homotopy classes and define a cyclic order in each free homotopy class, which is possible due to the uncrossing and allows for a simple greedy algorithm.

1.1   Our results

The (fractional) *maximum multiflow problem* can be described as follows. An instance consists of an undirected graph $(V, D \dot\cup E)$ whose edge set is partitioned into *demand edges*, in D, and *supply edges*, in E. We write $G = (V, E)$, $H = (V, D)$, and $G + H = (V, D \dot\cup E)$. Moreover we have a function $u : D \dot\cup E \to \mathbb{Z}_{>0}$ which defines a *capacity* $u(e)$ for each supply edge $e \in E$ and a *demand* $u(d)$ for each demand edge $d \in D$. The goal is to satisfy as much of the demand as possible by routing flow on supply edges. More precisely, we ask for an *s-t-flow* f^d of value at most $u(d)$ for every demand edge $d = \{t, s\}$ such that the total flow on each supply edge is at most its capacity and the total value of all those flows is maximum.

It is well known that every *s-t*-flow can be decomposed into flow on *s-t*-paths and on cycles, and for integral flows there is an integral decomposition. The cycles in such a decomposition do not contribute to the value of the *s-t*-flow and can be ignored. An *s-t*-path in (V, E) together with the demand edge $d = \{t, s\}$ forms a *D-cycle*: a cycle in $G + H$ that contains exactly one demand edge. If we let $\mathcal{C}$ denote the set of all *D*-cycles in $G + H$, we can write the maximum multiflow problem equivalently as

$$\max \sum_{C \in \mathcal{C}} f_C \quad \text{s.t.} \quad \begin{cases} \sum_{C \in \mathcal{C}: C \ni e} f_C \leqslant u(e) & \text{for all } e \in D \dot\cup E \\ f_C \geqslant 0 & \text{for all } C \in \mathcal{C} \end{cases} \tag{1}$$

In some previous works, the problem has been defined with $u(d) = \infty$ for $d \in D$, and this variant is easily seen to be equivalent. We call the linear program (1) the *maximum multiflow LP*. The *maximum integral multiflow* problem is identical, except that the flow must be integral:

$$\max \sum_{C \in \mathcal{C}} f_C \quad \text{s.t.} \quad \begin{cases} \sum_{C \in \mathcal{C}: C \ni e} f_C \leqslant u(e) & \text{for all } e \in D \dot\cup E \\ f_C \in \mathbb{Z}_{\geqslant 0} & \text{for all } C \in \mathcal{C} \end{cases} \tag{2}$$

The special case where $u(e) = 1$ for every edge $e \in D \dot\cup E$ is known as the *maximum edge-disjoint paths* problem. Even that special case is unlikely to have a constant-factor approximation algorithm for general graphs (see Section 1.2). Our main result is a constant-factor approximation algorithm in the case when $G + H$ can be embedded on an orientable surface of bounded genus.

▶ **Theorem 1.** *There is a polynomial-time algorithm which takes as input an instance (G, H, u) of the maximum integral multiflow problem such that $G + H$ is embedded on an orientable surface of genus g, and which outputs an integral multiflow whose value is at most a factor $O(g^2 \log g)$ smaller than the value of any fractional multiflow.*

See Section 3 for an outline of the algorithm and the proof. In the full version of the paper, we explain how to improve the approximation ratio of the algorithm to $O(g^2)$. It is worth pointing out that almost all known hardness results for the maximum edge-disjoint paths problem hold even when G is planar (see Section 1.2). Theorem 1, along with the two recent papers [16, 21], highlight that for tractability one needs more than the planarity of G alone. The topology of $G + H$ together plays an important role.

The dual LP of (1) is:

$$\min \sum_{e \in D \dot\cup E} u(e) y_e \quad \text{s.t.} \quad \begin{cases} \sum_{e \in C} y_e \geqslant 1 & \text{for all } C \in \mathcal{C} \\ y_e \geqslant 0 & \text{for all } e \in D \dot\cup E \end{cases} \tag{3}$$

and this may be called the *minimum fractional multicut problem*. The *minimum multicut problem* results from replacing the inequality $y_e \geqslant 0$ in (3) by $y_e \in \{0, 1\}$ for all edges $e \in D \,\dot{\cup}\, E$. Again, many previous works considered the equivalent special case where $u(d) = \infty$ for $d \in D$, in which case no dual variables for demand edges are needed. By weak duality, the value of any multiflow is at most the capacity of any multicut. Using Theorem 1 and a previous result of [36], we obtain (in Section 9):

▶ **Corollary 2.** *For any instance (G, H, u) of the maximum integral multiflow problem such that $G + H$ is embedded on an orientable surface of genus g, the minimum capacity of a multicut is at most $O(g^{3.5} \log g)$ times the maximum value of an integral multiflow.*

In general the integral multiflow-multicut gap[1], and even the integrality gap of (1), can be as large as $\Theta(|D|)$, even when G is planar and $G + H$ is embedded in the projective plane [18]. In this paper we consider orientable surfaces only. Corollary 2 states that the gap becomes constant when $G + H$ has bounded genus. So far very few such constant integral multiflow-multicut gaps are known, for example when G is a tree [18], or when $G + H$ is planar, as recently shown in [16, 21].

1.2   Related Work

Approximation algorithms and hardness for integral multiflows.   Most of the hardness results for the maximum integral multiflow problem follow from the special case of the maximum edge-disjoint paths problem (EDP). The decision version of EDP is one of Karp's original NP-complete problems [22], and remains NP-complete even in many special cases [30], including the case of interest in this paper, namely even when $G + H$ is planar [28]. In terms of approximation, EDP is APX-hard [2]. Assuming that $\text{NP} \not\subseteq \text{DetTIME}(n^{O(\log n)})$, where $n = |V|$, there is no $n^{o(1/\sqrt{\log n})}$ approximation for EDP, even when G is planar and sub-cubic [6]. Assuming that for some positive δ, $\text{NP} \not\subseteq \text{RandTIME}(2^{n^{\delta}})$, there is no $n^{O(1/(\log \log n)^2)}$ approximation for EDP, even when G is planar and sub-cubic [7]. As far as we know, no stronger hardness result is known for integral mutliflows.

On the positive side, EDP can be solved in polynomial time when the number of demand edges is bounded by a constant [33]. The same holds for integral multiflows when $G + H$ is planar [35]. For exact algorithms in various special cases, see the survey [30]. In general, the best known approximation guarantee for EDP and maximum integral multiflows is $O(\sqrt{n})$ [4]. Approximation algorithms with better approximation ratios for various special cases have been designed. We refer the readers to the survey [10] and to [18, 23, 30] and the references therein.

Recent work on the planar case.   Recently, [16] and [21] gave constant-factor approximation algorithms for maximum integer multiflows when $G + H$ is planar. Both papers proceed by first obtaining a half-integral multiflow and then using the four color theorem to round it to an integral solution (similar to Section 6). The main difference of the two works is the way such half-integral multiflows are obtained. In [16], it is constructed by uncrossing a fractional multiflow (see Section 5 for a definition) to construct a certain network matrix, which is known to be totally unimodular; in [21], such a half-integral multiflow is obtained

---

[1]   There is a closely related, but different, notion of integral flow-cut gap introduced in [5]: they study the smallest constant c such that whenever $u(C \cap E) \geqslant u(C \cap D)$ for every cut C (the cut condition), there is an integral multiflow satisfying all demands and violating capacities by at most a factor c.

by rounding a feasible solution of a related problem in the planar dual graph of $G + H$. Both approaches do not extend to higher genus graphs in a straightforward way, because the dual of a cycle is no longer a cut in general and cycles cannot always be uncrossed.

Minimum multicut problem.   The minimum multicut problem is NP-hard even when there are only three demand edges [11]. In general, assuming that the Unique Games conjecture holds, there is no $O(1)$-approximation [3], but a $O(\log |D|)$-approximation algorithm [17]. Better approximations also have been shown for special cases; see [18, 36] and the references therein. In particular, when $G + H$ is planar, [25] gave an approximation scheme. When G has genus g, an FPT-approximation scheme with parameters of g and $|D|$ has been proposed [8].

Tools from topology.   The design of multiflows on surfaces is closely related to the properties of sets of curves on a surface. In a recent breakthrough, Przytycki [31] proved that the maximum number of essential curves on a closed surface of genus g such that no two of them are freely homotopic or intersect more than once is $O(g^3)$, improving on the previous exponential upper bound by [27]. Very recently, this number was shown to be $O(g^2 \log g)$ by [19], which almost matches the lower bound $\Omega(g^2)$ on the size of such sets [27]. We will use this result in Section 7.

2   Preliminaries

Consider an instance (G, H, u) of the maximum integral multiflow problem, and let $G + H = (V, E \,\dot\cup\, D)$ be the graph whose edge set is the disjoint union of the edge sets of the supply graph $G = (V, E)$ and the demand graph $H = (V, D)$. Throughout the paper, we assume that the graph $G + H$ is connected, otherwise we can run the algorithm on each of its connected components.

Graphs on surfaces.   Surfaces are either orientable or non-orientable; in this paper we only consider closed orientable surfaces. A closed orientable surface of genus g can be seen as a connected sum of g tori, or equivalently a sphere with g handles attached on it, where g is called the *genus* of the surface. Given an integer $g \geqslant 0$, all closed surfaces with genus g are mutually homeomorphic, and we refer to any one of them as $\mathbb{S}_g$. For instance, $\mathbb{S}_0$ is the sphere and $\mathbb{S}_1$ is the torus.

A (multi)graph has *genus g* or is a *genus-g graph*, if it can be drawn on $\mathbb{S}_g$ without edge crossings, but not on $\mathbb{S}_{g-1}$. A genus-g graph may have several non-equivalent embeddings on $\mathbb{S}_g$, but all of them satisfy the same invariant, called the *Euler characteristic*: #Faces − #Edges + #Vertices = $2 - 2g$.

A simple application of Euler's formula gives the following upper bound on the coloring number of genus-g graphs, when $g \geqslant 1$.

▶ **Theorem 3** (Map color theorem). *A genus-g graph can be colored in polynomial time with at most $\chi_g \leqslant \lfloor \frac{7+\sqrt{1+48g}}{2} \rfloor$ colors.*

For $g = 0$, this is an algorithmic version of the 4-color theorem [32]. For $g \geqslant 1$, the coloring is obtained in polynomial time by a simple recursive algorithm that removes a vertex of minimum degree and colors the remaining graph [20]. For additional details and results about graphs on surfaces see e.g. [29, 9].

Combinatorial embeddings.   Given a graph, let $\delta(v)$ denote the set of edges incident to a vertex v, and $\delta(U)$ the set of edges with exactly one endpoint in vertex set U. Given an embedding of a graph on an orientable surface, and an arbitrary orientation of this surface, for each vertex v, a clockwise cyclic order can be defined on the edges of $\delta(v)$. Note that contracting an edge $e = \{u, v\}$ results in removing e from $\delta(u)$ and from $\delta(v)$ and concatenating the orders to obtain the clockwise cyclic order of the edges around the vertex created by the contraction. Using these orders together with the incidence relation between edges and faces, embeddings become purely combinatorial objects. For additional details see, e.g., [29], Chapter 4.

Graph duality.   Given an embedding of a genus-g graph G on $\mathbb{S}_g$, there exists a uniquely defined dual graph, denoted as G^*. This graph can be embedded on the same surface as G. There exists a bijection between the faces of G and the vertices of G^*, a bijection between the vertices of G and the faces of G^*, and a bijection between the edge sets of G and of G^*. Moreover, the embeddings of G and G^* are consistent: with this bijection, every edge only crosses its dual edge, every face only contains its corresponding dual vertex and reciprocally. For notational simplicity, the latter bijection is implicit.

Cycles and cuts.   A *path* in a graph G is a sequence $(v_0, e_1, v_1, \ldots, e_k, v_k)$ for some $k \geqslant 0$, where $v_0, \ldots, v_k$ are distinct vertices and $e_i = \{v_{i-1}, v_i\}$ is an edge for all $i = 1, \ldots, k$. A *cycle* in a graph G is a sequence $(v_0, e_1, v_1, \ldots, e_k, v_k)$ such that $v_1, \ldots, v_k$ are distinct vertices, $\{v_{i-1}, v_i\}$ is an edge for all $i = 1, \ldots, k$, and $v_0 = v_k$. Sometimes we view cycles as edge sets or as graphs. A *cut* is an edge set $\delta(U)$ for some proper subset $\emptyset \neq U \subset V$. A cut $\delta(U)$ is *simple* if both U and $V \setminus U$ are connected.

   We say that an edge set F in a graph is a (simple) *dual cut* if the corresponding set of edges F^* in the dual is a (simple) cut. A cycle C in G is called *separating* if it is a dual cut, and *non-separating* otherwise. Note that every separating cycle is a simple dual cut.

Homotopy.   Given a surface $\mathbb{S}$, a (simple) *topological cycle* is a continuous injective map γ from the unit cycle $S^1 := \{z \in \mathbb{C}, \; ||z|| = 1\}$ to $\mathbb{S}$. Two topological cycles γ_1 and γ_2 are *freely homotopic* if there exists a continuous function $\varphi : [0, 1] \times S^1 \to \mathbb{S}$ such that $\varphi(0, \cdot) = \gamma_1$ and $\varphi(1, \cdot) = \gamma_2$. Intuitively, cycle γ_1 is transformed into cycle γ_2 by continuously moving it on the surface. Free homotopy is an equivalence relation.

   Given an embedding of the graph $G + H$ on $\mathbb{S}$, we say that a cycle C in $G + H$ *is represented[2] by* a topological cycle γ of $\mathbb{S}$ if the image of γ is the embedding of C on $\mathbb{S}$. Two cycles in $G + H$ are freely homotopic if and only if they can be represented by two freely homotopic topological cycles. In the sequel, we use the following well-known fact.

▶ **Fact 4.** *If two cycles C and C' are freely homotopic, then their symmetric difference is a dual cut. If C and C' are additionally disjoint and non-separating, then their union is a simple dual cut.*

   Intuitively, the image of the continuous homotopy function from C to C' on the surface forms an annulus [12]. See Figure 1 for an illustration.

---

[2] Topological cycles are considered up to orientation-preserving reparameterization. Therefore, a cycle in $G + H$ may be represented by a topological cycle from two classes, one for each orientation: the class of γ and the class of γ' where $\gamma'(e^{i\theta}) = \gamma(e^{-i\theta})$.
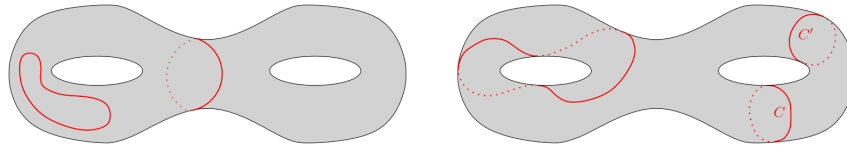
Figure 1 Some cycles on an orientable surface of genus 2. On the left, two *separating* cycles. On the right, three *non-separating* cycles. C and C' are *freely homotopic* and their union disconnects the surface.

3   Overview

In this section, we give an overview of our constant-factor approximation algorithm for the maximum integral multiflow problem when $G + H$ is embedded on an orientable surface $\mathbb{S}_g$ of genus g, where g is bounded by a constant (Theorem 1). Again, without loss of generality, we assume that $G + H$ is connected. Here is the main algorithm. Steps 1,2,3,4 will be described in detail in Sections 4,5,6,7, respectively.

1. Solve the linear program (1) to obtain a (fractional) multiflow f^*.
2. Construct another multiflow $\overline{f}$ such that any two cycles in the support of $\overline{f}$ cross at most once (Lemma 7). See Definition 6 for the definition of "crossing."
3. If at least half of the total value of $\overline{f}$ is contributed by separating cycles, these cycles now form a laminar family. Construct a half-integral multiflow f^{half} (Theorem 10), and from there, using the map color theorem (Theorem 3), compute an integral multiflow f' (Lemma 11), which is the output.
4. Otherwise, partition the non-separating cycles in the support of $\overline{f}$ into free homotopy classes. Pick the class $\mathcal{H}$ with largest total flow value. Remove the flow on all other cycles and greedily construct an integral multiflow (Lemmas 20 and 17), which is the output.

It can be proved that we only lose a constant factor at every step of the algorithm: see Section 8 for the analysis of the above algorithm, proving Theorem 1.

4   Finding a fractional multiflow (Step 1)

A feasible solution f to the maximum multiflow LP (1) will be simply called a *multiflow*. Recall that $\mathcal{C}$ denotes the set of all D-cycles, i.e., all cycles in $G + H$ that contain precisely one demand edge. We denote by $|f| = \sum_{C \in \mathcal{C}} f_C$ the *value* of f, and by $\mathcal{C}(f) := \{C \in \mathcal{C} \mid f_C > 0\}$ the *support* of f. Although formulation (1) has an exponential number of variables, it is well known that it can be reformulated by polynomially many flow variables and constraints (see, e.g., [15, 1]) and thereby solved in polynomial time:

▶ **Proposition 5.** *There is an algorithm that finds an optimal solution f^* to the maximum multiflow LP (1) such that $|\mathcal{C}(f^*)| \leqslant |D||E|$. Its running time is polynomial in the size of the input graph.*

Proof. By introducing flow variables $x_e^d := \sum_{C \in \mathcal{C}:d,e \in C} f_C$ for all $d \in D$ and $e \in D \cup E$ we can maximize the total value $\sum_{d \in D} x_d^d$ subject to nonnegativity and flow conservation constraints (for each $d \in D$ and for each vertex). This is a linear program of polynomial size. By flow decomposition, one can then construct a feasible solution to (1) of the same value and with support at most $|D||E|$. ◀

Later we will restrict a multiflow to subsets of D-cycles. For $\mathcal{C}' \subseteq \mathcal{C}$ we define a multiflow f' by $f'_C := f_C$ for $C \in \mathcal{C}'$ and $f'_C := 0$ for $C \in \mathcal{C} \setminus \mathcal{C}'$, and write $f(\mathcal{C}') := f'$.

5   Making a fractional flow minimally crossing (Step 2)

In this section we show that for a given embedding, we can "uncross" a multiflow in such a way that any two D-cycles in the support cross at most once. While doing this we will lose only an arbitrarily small fraction of the multiflow value.

Uncrossing is a well-known technique in combinatorial optimization, but in most cases it is applied to families of subsets of a ground set U. Such a family is said to be cross-free if, for any two of its sets, A and B, at least one of the four sets $A \setminus B$, $B \setminus A$, $A \cap B$, and $U \setminus (A \cup B)$ is empty. Here we want to uncross D-cycles in the topological sense, and this can be reduced to the above (with some extra care) only if all these cycles are separating (which, for example, is always the case if $G + H$ is planar; cf. [16]).

▶ **Definition 6.** *We say that two D-cycles C_1 and C_2 cross if there exists a path P (possibly a single vertex), which is a subpath of both C_1 and C_2, and such that in the embedding, after contracting the edges of P, the vertex v thus obtained is incident to two edges of C_1 and to two edges of C_2, all distinct, and in the embedding the restriction of the cyclic order of $\delta(v)$ to those four edges alternates between an edge of C_1 and an edge of C_2.*
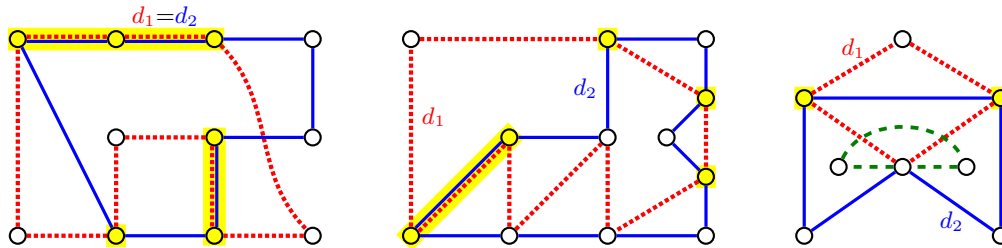
■ **Figure 2** Each of the two figures on the left show two D-cycles, C_1 (red, dotted) and C_2 (blue, solid). The edges belonging to D are marked as d_1 and d_2. Edges are arranged at every vertex in the order of their embedding. Crossings are marked by yellow shade. The two D-cycles on the left cross three times. The two D-cycles in the middle cross four times. The figure on the right shows two D-cycles C_1 and C_2 that cross twice, and a third D-cycle C_3 (green, dashed) that crosses neither C_1 nor C_2. Uncrossing C_1 and C_2 here generates a crossing of C_3 with a new D-cycle (namely with the triangle containing d_2).

Two cycles may cross multiple times. We denote by $cr(C, C')$ the number of times that C and C' cross. See Figure 2 for three examples. In contrast to the planar case, it is possible that two cycles cross exactly once and cannot be uncrossed. The third example in Figure 2 shows another difficulty: when uncrossing two D-cycles it might be necessary to generate new crossings with other cycles.

▶ **Lemma 7.** *Let $\epsilon > 0$ be fixed. Given a multiflow f whose support has size at most $|E||D|$, there is a polynomial-time algorithm to construct another multiflow $\overline{f}$, of value at least $|\overline{f}| \geqslant (1 - \epsilon)|f|$, and such that any two cycles in the support of $\overline{f}$ cross at most once.*

Proof. First we discretize the multiflow, losing an ϵ fraction in value; then we iteratively modify it, without changing its value, to reduce the number of crossings or the total amount of flow on all edges; finally, we analyze the process and argue that the number of iterations is polynomially bounded.

Discretization.   The statement is trivial if $|f| = 0$. Otherwise, before uncrossing, we round down the flow on every D-cycle to integer multiples of $\frac{\epsilon|f|}{|E||D|}$. That is, we define $f'_C :=$ $\frac{\epsilon|f|}{|E||D|} \left\lfloor \frac{|E||D|f_C}{\epsilon|f|} \right\rfloor$ for all $C \in \mathcal{C}$. Note that f' is a multiflow. We claim that $|f'| \geqslant (1 - \epsilon)|f|$. Indeed,

$$|f'| = \sum_{C \in \mathcal{C}} f'_C \geqslant \sum_{C \in \mathcal{C}(f)} \left(f_C - \frac{\epsilon|f|}{|E||D|} \right) = |f| - |\mathcal{C}(f)| \frac{\epsilon|f|}{|E||D|} \geqslant |f| - \epsilon|f|.$$

The discretized multiflow f' can be represented by a multi-set $\mathcal{S}$ of unweighted D-cycles: if $f'_C = k \frac{\epsilon|f|}{|E||D|}$, then k identical copies of cycle C are added to $\mathcal{S}$. The number of cycles in $\mathcal{S}$ (counting multiplicities) is at most $\frac{|E||D|}{\epsilon}$ because $|\mathcal{S}| = \sum_{C \in \mathcal{C}} f'_C \frac{|E||D|}{\epsilon|f|} \leqslant \sum_{C \in \mathcal{C}} f_C \frac{|E||D|}{\epsilon|f|} = \frac{|E||D|}{\epsilon}$.

Uncrossing.   To construct $\overline{f}$, we perform a sequence of transformations of the multiflow. We will modify $\mathcal{S}$ while maintaining the following invariants:

(a) The number of elements of $\mathcal{S}$ (counting multiplicities) remains constant.

(b) For every $e \in D \,\dot\cup\, E$, the number of elements of $\mathcal{S}$ (counting multiplicities) that contain e never increases.

Thanks to (b), at any stage, $\overline{f}$ is a multiflow, where $\overline{f}$ is defined by $\overline{f}_C = k \frac{\epsilon|f|}{|E||D|}$ for $C \in \mathcal{C}$, where k is the multiplicity of C in $\mathcal{S}$. Initially $\overline{f} = f'$. Thanks to (a), the value of the multiflow is preserved. In the following we work only with $\mathcal{S}$.

While there exist two cycles C_1 and C_2 in $\mathcal{S}$ that cross at least twice, do the following *uncrossing* operation (on one copy of C_1 and one copy of C_2). Let d_1 be the edge in $C_1 \cap D$, and let d_2 be the edge in $C_2 \cap D$. Let P and Q be two of the paths where C_1 and C_2 cross (cf. Definition 6), such that Q contains only edges of E. Orient C_1 so that in that orientation, when traversing the entirety of P and then walking towards Q, edge d_1 is traversed before reaching Q. Let $\vec{C}_1$ denote the resulting directed cycle. Let a be the first vertex on P in the orientation of $\vec{C}_1$, and let b be an arbitrary vertex on Q. Vertices a and b partition $\vec{C}_1$ into a path C_1^+ from a to b that contains d_1 and a path C_1^- from b to a that does not contain d_1.

Case 1: P contains an edge of D. Then this edge is $d_1 = d_2$. We orient C_2 so that the orientation on P agrees with the orientation of $\vec{C}_1$ on P. Let $\vec{C}_2$ denote the resulting directed cycle. Then the vertices a and b also partition $\vec{C}_2$ into a path C_2^+ from a to b that contains d_2 and a path C_2^- from b to a that does not contain d_2.

Case 2: P contains edges of E only. Then we orient C_2 so that in that orientation, when traversing the entirety of P and then walking towards Q, edge d_2 is traversed before reaching Q. Let $\vec{C}_2$ denote the directed cycle. With that orientation, vertices a and b also partition $\vec{C}_2$ into a path C_2^+ from a to b that contains d_2 and a path C_2^- from b to a that does not contain d_2.

To obtain C'_1, we concatenate C_1^+ and C_2^-, remove any cycle that does not contain d_1, and remove the orientation. To obtain C'_2, we concatenate C_2^+ and C_1^-, remove any cycle that does not contain d_2, and remove the orientation. Note that C'_1 and C'_2 are D-cycles because each of C_1^+ and C_2^+ contains exactly one demand edge, and C_1^- and C_2^- contain no demand edge.

See Figure 3 for two examples, one for each case.
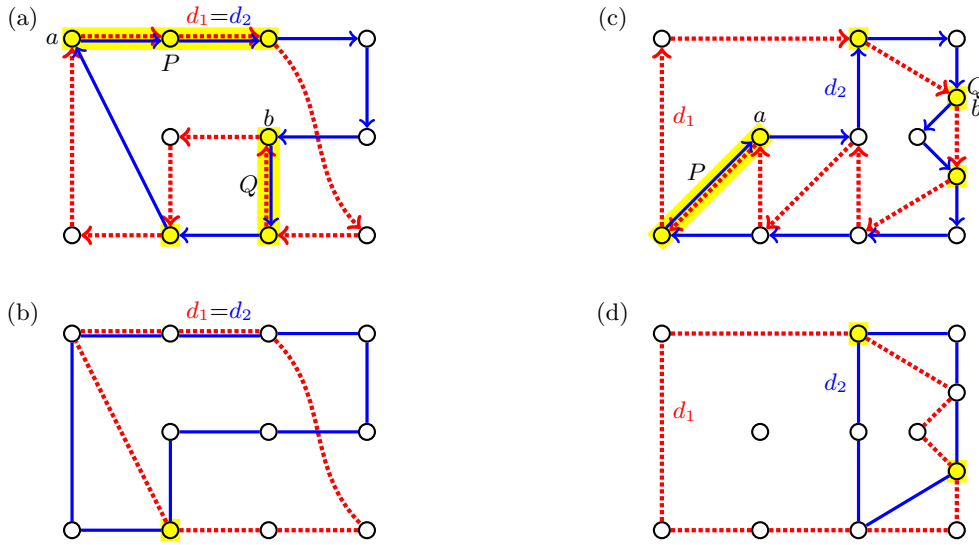
Figure 3 Uncrossing the pairs of D-cycles from Figure 2. (a) and (b) show an example for Case 1, (c) and (d) an example for Case 2. The initial situation (C_1 red, dotted, and C_2 blue, solid) and a possible choice of P, Q, a, b and the resulting orientation is shown in (a) and (c). As the result of the uncrossing operation, shown in (b) and (d), we have the new D-cycles C_1' (red, dotted) and C_2' (blue, solid) with fewer crossings among each other.

Analysis. From the construction it follows that C_1' and C_2' are D-cycles and $C_1' \dot\cup C_2' \subseteq C_1 \dot\cup C_2$. Hence removing one copy of C_1 and C_2 from $\mathcal{S}$ and adding one copy of C_1' and C_2' to $\mathcal{S}$ maintains the invariants (a) and (b).

To show that the after a polynomial number of uncrossing operations any pair of cycles in $\mathcal{S}$ crosses at most once, we consider the total number of edges $\Phi_1 = \sum_{C \in \mathcal{S}} |C|$ (counting multiplicities) and the total number of crossings $\Phi_2 = \sum_{C,C' \in \mathcal{S}} cr(C, C')$ (where we again count multiplicities). Note that $|\mathcal{S}|$ remains constant by invariant (a), and Φ_1 never increases by invariant (b). Moreover $0 \leqslant \Phi_1 \leqslant |V||\mathcal{S}|$ and $0 \leqslant |\Phi_2| \leqslant |V||\mathcal{S}|^2$.

$\triangleright$ **Claim 8.** Each uncrossing operation either decreases Φ_1 or leaves Φ_1 unchanged and decreases Φ_2.

To prove Claim 8, consider an uncrossing operation that replaces C_1 and C_2 by C_1' and C_2', and suppose that Φ_1 remains the same, so C_1' consists of C_1^+ plus C_2^-, and C_2' consists of C_2^+ plus C_1^-. We first observe that $cr(C_1', C_2') < cr(C_1, C_2)$. Indeed, the crossings at P and at Q go away, and no new crossing arises.

Finally we need to show that for any cycle $C \in \mathcal{C}$,

$$cr(C, C_1') + cr(C, C_2') \leqslant cr(C, C_1) + cr(C, C_2). \tag{4}$$

To show (4), consider a crossing of C and $C' \in \{C_1', C_2'\}$ at a path R. Let $e_1' = \{v_0, v_1\}, \ldots, e_k' = \{v_{k-1}, v_k\}$ be the edges of R ($k \geqslant 0$), and let $e_0, e_{k+1}, e_0', e_{k+1}'$ be edges such that $e_0, e_1', \ldots, e_k', e_{k+1}$ are subsequent on C and $e_0', e_1', \ldots, e_k', e_{k+1}'$ are subsequent on C'. After contracting R, the incident edges $e_0, e_0', e_{k+1}, e_{k+1}'$ are embedded in this cyclic order. (Note that $e_0 = e_{k+1}$ or $e_0' = e_{k+1}'$ is possible if $k \geqslant 1$, then contracting R yields a loop.) See Figure 4 (a).

Now e_0' belongs to C_1 or C_2, say C_1. If R contains neither a nor b, then $e_0', \ldots, e_{k+1}'$ all belong to C_1, and C_1 crosses C at R. If R contains either a or b, say at v_i, then $e_0', \ldots, e_i'$ belong to C_1 and $e_{i+1}', \ldots, e_{k+1}'$ belong to C_2. Moreover C_1 and C_2 cross at a path containing
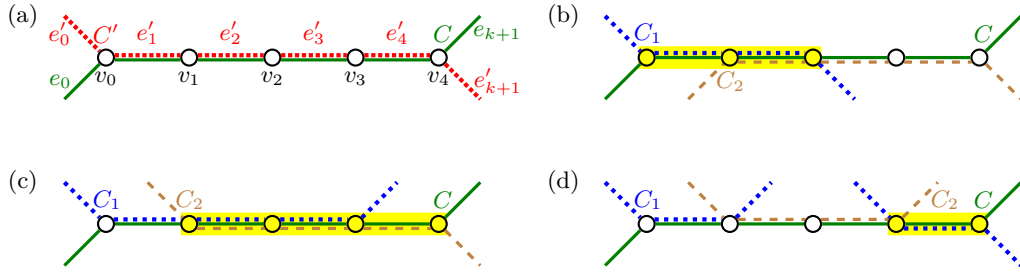
Figure 4 For each crossing of C with a new cycle $C' \in \{C'_1, C'_2\}$ at a path R there is a crossing of C with one of the old cycles C_1 and C_2 at a subpath of R. This crossing is marked with yellow shade in the three examples.

v_i, so either C_1 crosses C at a subpath of R (Figure 4(b)) or C_2 crosses C at a subpath of R (Figure 4(c)). Finally, if R contains a and b, say at v_i and v_j for $0 \leqslant i < j \leqslant k$, then $e'_0, \ldots, e'_i$ and $e'_{j+1}, \ldots, e'_{k+1}$ belong to C_1 and $e'_{i+1}, \ldots, e'_j$ belong to C_2 (Figure 4(d)). Again, C_1 or C_2 crosses C at a subpath of R. This concludes the proof of Claim 8.

We can now conclude the proof of Lemma 7 because Φ_1 decreases at most $|V||\mathcal{S}|$ times, and while Φ_1 is constant, Φ_2 decreases at most $|V||\mathcal{S}|^2$ times, so the total number of uncrossing operations is at most $|V|^2|\mathcal{S}|^3 \leqslant \frac{|V|^2|E|^3|D|^3}{\epsilon^3}$.   ◀

6   Separating cycles: routing an integral flow (Step 3)

Let $\overline{f}$ result from Lemma 7, and let $\mathcal{C}_{\mathrm{sep}}$ denote the set of separating cycles in the support of $\overline{f}$. We now consider the case when the separating cycles contribute at least half to the total flow value, i.e., $|\overline{f}(\mathcal{C}_{\mathrm{sep}})| \geqslant \frac{1}{2}|\overline{f}|$.

This branch of our algorithm consists of two steps:

1. Given $\overline{f}(\mathcal{C}_{\mathrm{sep}})$, construct a half-integral multiflow f^{half} of value at least $|\overline{f}|/2$;
2. Given f^{half}, construct an integral multiflow of value at least $|f^{\mathrm{half}}|/\Theta(\sqrt{g})$.

6.1   Obtaining a half-integral multiflow

To obtain a half-integral multiflow, we follow the technique used by [16] for the case where $G + H$ is planar. By the Jordan curve theorem, any cycle in a planar graph is separating. As for the plane, the following property is easy to check for higher genus surfaces.

▶ **Proposition 9.** *If C and C' are two cycles embedded on a surface, and C' is a separating cycle, then C and C' must cross an even number of times.*

Proof. C' is separating the surface into two sides. While walking along C from a vertex v, we go from one side to the other each time we cross C'. When we return at v, we are on the same side where we started so the number of crossing is even.   ◀

Since any pair of cycles in the support of $\overline{f}$ crosses at most once, $\mathcal{C}_{\mathrm{sep}}$ must be a non-crossing family by Proposition 9. In particular, we can show that $\mathcal{C}_{\mathrm{sep}}$ have a laminar structure.

We say that a family of subsets of the dual vertex set V^* is laminar if any two members either are disjoint or one contains the other. Let us take any face of $G + H$ that we call ∞. For any cycle $C \in \mathcal{C}_{\mathrm{sep}}$ we define $\mathrm{in}(C)$ and $\mathrm{out}(C)$ to be the two connected components of $(G + H)^* \setminus C^*$, such that $\infty \in \mathrm{out}(C)$. We claim that the family $\mathcal{L} := \{\mathrm{in}(C) : C \in \mathcal{C}_{\mathrm{sep}}\}$ is laminar.

Indeed, take any two cycles C and C' in $\mathcal{C}_{\text{sep}}$. Since they do not cross, either (i) $(C' \setminus C)^* \subseteq \text{in}(C)$ or, (ii) $(C' \setminus C)^* \subseteq \text{out}(C)$. In case (i) we must have $\text{in}(C') \subseteq \text{in}(C)$. In case (ii), we have either (ii.a) $\text{in}(C) \subseteq \text{in}(C')$ or (ii.b) $\text{in}(C) \cap \text{in}(C') = \emptyset$, hence laminarity.

Using the terminology in [16], we say that a multiflow f is *laminar* if $\{C^* : C \in \mathcal{C}, f_C > 0\} = \{\delta(U) : U \in \mathcal{L}\}$ where $\mathcal{L}$ is a laminar family (of subsets of V^*). Thus, $\overline{f}(\mathcal{C}_{\text{sep}})$ is laminar and we can apply the following result to get f^{half}.

▶ **Theorem 10** ([16]). *If f is a laminar multiflow, then there exists a laminar half-integral multiflow f' such that $\mathcal{C}(f') \subseteq \mathcal{C}(f)$ of value $|f'| \geqslant \frac{1}{2}|f|$. Such a multiflow can be computed in polynomial time.*

6.2 Obtaining an integral multiflow

In this section we show the following result, which is an extension of a result from [21, 16], who proved it for planar graphs.

▶ **Lemma 11.** *Let (G, H, u) be an instance of the maximum multiflow problem such that $G+H$ has genus g, and let f^{half} be a laminar half-integral multiflow whose support $\mathcal{C}(f^{\text{half}})$ contains only separating cycles. Then there exists an integral multiflow f' of value $|f'| \geqslant 2|f^{\text{half}}|/\chi_g$ (such that $\mathcal{C}(f') \subseteq \mathcal{C}(f^{\text{half}})$). Such a multiflow can be found in polynomial time.*

Our proof follows the same outline as the proof of Theorem 1 of Fiorini et al. [14]. Let $\mathcal{C}^{\text{half}} := \mathcal{C}(f^{\text{half}})$ be the set of D-cycles C such that $f_C^{\text{half}} > 0$. We first reduce the problem to the case where all cycles in $\mathcal{C}^{\text{half}}$ have flow value $\frac{1}{2}$ and every edge has capacity 1. To do that, we reduce the flow f_C^{half} by $\lfloor f_C^{\text{half}} \rfloor$ for each cycle $C \in \mathcal{C}^{\text{half}}$, and reduce edge capacities accordingly. Then, since now f^{half} is small, we can further reduce demands and capacities to $u'(e) = \min\{u(e), |\mathcal{C}(f^{\text{half}})|\}$ for each $e \in E \cup D$, so that $\sum_{e \in D \dot\cup E} u(e)$ is polynomially bounded. We can then replace each edge e by $u(e)$ parallel edges of unit capacity. Given a cycle C such that $f_C^{\text{half}} = \frac{1}{2}$, we replace each edge $e \in C$ by one of its parallel edges. This can be done while ensuring that the resulting flow is still feasible and laminar. To facilitate the proof, we still denote this graph by $G + H$ and keep all other notations.

Recall that cycles in $\mathcal{C}^{\text{half}} \subseteq \mathcal{C}_{\text{sep}}$ are separating and do not cross each other, so that the family $\{\text{in}(C), C \in \mathcal{C}^{\text{half}}\}$ is laminar. We partially order $\mathcal{C}^{\text{half}}$ with the following relation: $C \prec C'$ if $\text{in}(C) \subset \text{in}(C')$. We have the following simple property:

▶ **Lemma 12.** *If $C_1, C_2, C' \in \mathcal{C}^{\text{half}}$ are such that $C_1 \prec C'$ and $C_2 \nprec C'$, then C_1 and C_2 are edge-disjoint.*

For the proof, see the full version of the paper. Our goal is to get a large subset $\mathcal{C}' \subseteq \mathcal{C}^{\text{half}}$ such that any two cycles in $\mathcal{C}'$, are edge-disjoint. This is equivalent to finding a large independent set in a properly defined graph $\text{Int}(\mathcal{C}^{\text{half}})$ with vertex set $\mathcal{C}^{\text{half}}$ and such that two cycles are adjacent if they share at least one edge. Using Lemma 12 we can show:

▶ **Lemma 13.** *Given a graph embedded in $\mathbb{S}_g$, let $\mathcal{C}^{half}$ be defined as above. Let $Int(\mathcal{C}^{half})$ be the graph with vertex set $\mathcal{C}^{half}$ and such that two cycles are adjacent if they share at least one edge. Then $Int(\mathcal{C}^{half})$ is a genus-g graph.*

Proof. We prove the statement by induction on $g + |\mathcal{C}^{\text{half}}|$. When $g + |\mathcal{C}^{\text{half}}| \leqslant 2$, it is trivial. Otherwise let G be a connected genus-g graph, embedded on $\mathbb{S}_g$, and $\mathcal{C}^{\text{half}}$ a family as described above.

Suppose first that $\{\text{in}(C) \mid C \in \mathcal{C}^{\text{half}}\}$ are pairwise disjoint. Then, contract in G^* each set $\text{in}(C)$ into a single node. Two cycles C and C' share an edge if and only if in this contracted graph, the nodes corresponding to $\text{in}(C)$ and $\text{in}(C')$ are adjacent. This means that $\text{Int}(\mathcal{C}^{\text{half}})$ is a minor of G^*, and in particular has genus less than or equal to the genus of G^*.

The case where there is one cycle $\bar{C}$ such that $C \prec \bar{C}$ for all $C \in \mathcal{C}^{\mathrm{half}} \setminus \bar{C}$ and $\{\mathrm{in}(C) \mid C \in \mathcal{C}^{\mathrm{half}} \setminus \bar{C}\}$ are pairwise disjoint works similarly; here we contract $\mathrm{out}(\bar{C})$.

Otherwise there exists a triple $C_1, C_2, C \in \mathcal{C}^{\mathrm{half}}$ such that $C_1 \prec C$ and $C_2 \nprec C$. The separating cycle C divides $\mathbb{S}_g$ into two sides. Each side can be closed – by identifying the boundary of a disk with the boundary form by C – so that they are homeomorphic to $\mathbb{S}_{g_{\mathrm{in}}}$ and $\mathbb{S}_{g_{\mathrm{out}}}$, respectively. The connected sum of these two surfaces is homeomorphic to $\mathbb{S}_g$, and in particular we have $g_{\mathrm{in}} + g_{\mathrm{out}} = g$. This equality can easily be checked with Euler's formula.

Let G_{in} (*resp.* G_{out}) be the subgraph of G induced by the vertices embedded on the side corresponding to $\mathbb{S}_{g_{\mathrm{in}}}$ (*resp.* $\mathbb{S}_{g_{\mathrm{out}}}$), such that both contain C. The embedding of G in $\mathbb{S}_g$ induces an embedding of G_{in} in $\mathbb{S}_{g_{\mathrm{in}}}$ and an embedding of G_{out} in $\mathbb{S}_{g_{\mathrm{out}}}$. Thus, $\mathrm{genus}(G_{\mathrm{in}}) + \mathrm{genus}(G_{\mathrm{out}}) \leqslant g$.

Now we define $\mathcal{C}^{\mathrm{half}}_{\preceq C} := \{C' \in \mathcal{C}^{\mathrm{half}} | C' \prec C\} \cup \{C\}$ and $\mathcal{C}^{\mathrm{half}}_{\nprec C} := \{C' \in \mathcal{C}^{\mathrm{half}} | C' \nprec C\} \cup \{C\}$. The choice of C implies that these two families are proper subsets of $\mathcal{C}^{\mathrm{half}}$. Since the cycles in $\mathcal{C}^{\mathrm{half}}$ do not cross, we have $\{C \in \mathcal{C}^{\mathrm{half}} : C \subseteq G_{\mathrm{in}}\} = \mathcal{C}^{\mathrm{half}}_{\preceq C}$ and $\{C \in \mathcal{C}^{\mathrm{half}} : C \subseteq G_{\mathrm{out}}\} = \mathcal{C}^{\mathrm{half}}_{\nprec C}$.

By the induction hypothesis, $\mathrm{Int}(\mathcal{C}^{\mathrm{half}}_{\preceq C})$ and $\mathrm{Int}(\mathcal{C}^{\mathrm{half}}_{\nprec C})$ can be embedded on $\mathbb{S}_{g_{\mathrm{in}}}$ and $\mathbb{S}_{g_{\mathrm{out}}}$, respectively. By Lemma 12, the graph $\mathrm{Int}(\mathcal{C}^{\mathrm{half}})$ arises from $\mathrm{Int}(\mathcal{C}^{\mathrm{half}}_{\preceq C})$ and $\mathrm{Int}(\mathcal{C}^{\mathrm{half}}_{\nprec C})$ by identifying the two vertices that correspond to C.

Finally we prove that $\mathrm{Int}(\mathcal{C}^{\mathrm{half}})$ can be embedded on a surface genus $g_{\mathrm{in}} + g_{\mathrm{out}} \leqslant g$. To see that, remove small disks D_{in} and D_{out} in $\mathbb{S}_{g_{\mathrm{in}}}$ and $\mathbb{S}_{g_{\mathrm{out}}}$, respectively, around the point that corresponds to vertex C and that intersects only edges incident to C, and glue them together by identifying boundaries of D_{in} and D_{out}. The surface obtained is homeomorphic to $\mathbb{S}_{g_{\mathrm{in}}+g_{\mathrm{out}}}$ It is easy to see that C, and the edges incident to C, can be re-embedded in this surface without intersecting any other edges. This terminates the proof of Lemma 13. ◄

Using Theorem 3, this lemma ensures that one can compute in polynomial time a subset $\mathcal{C}' \subseteq \mathcal{C}^{\mathrm{half}}$ of at least $|\mathcal{C}^{\mathrm{half}}|/\chi_g$ pairwise edge-disjoint D-cycles. From this set, we define an integral multiflow by setting $f'_C = 1$ for $C \in \mathcal{C}'$ and $f'_C = 0$ for $C \in \mathcal{C} \setminus \mathcal{C}'$. It is easy to check that f' is a multiflow that satisfies the properties of Lemma 11.

7   Non-separating cycles: routing an integral multiflow (Step 4)

If the separating cycles contribute less than half to the total value of the multiflow $\overline{f}$ obtained by Lemma 7, we consider the non-separating cycles in the support of $\overline{f}$. We first partition them into *free homotopy classes*. The next theorem gives an upper bound on the number of such classes.

▶ **Theorem 14** ([19]). *Let $\mathbb{S}_g$ be an orientable surface of genus g. Then there are at most $O(g^2 \log g)$ topological cycles such that any two of them are in different free homotopy classes and cross each other at most once.*

▶ **Corollary 15.** *The D-cycles in the support of $\overline{f}$ can be partitioned into $O(g^2 \log g)$ free homotopy classes in polynomial time.*

Proof. Take pairs of cycles in the support of $\overline{f}$ and check whether they are freely homotopic, for example as in [13, 26]. ◄

7.1 Greedy algorithm

Let $\mathcal{H}$ be a free homotopy class of non-separating cycles whose total flow value $|\overline{f}(\mathcal{H})|$ is largest. We will run the following simple greedy algorithm (Algorithm 1) on $\mathcal{H}$ to get an integral multiflow.

■ **Algorithm 1** Greedy algorithm for integral multiflows.

Input: a sequence $C_1, \ldots, C_k$ of D-cycles of $\mathcal{C}(\overline{f})$.
Output: an integral multiflow f.
$f \leftarrow$ the all-zero multiflow;
for $i = 1$ *to* k **do**
$\quad\lfloor$ Set f_{C_i} to be the greatest integer such that f remains feasible.

The value of the integral multiflow returned by this algorithm depends on the order of the D-cycles in the input. If it is ordered according to the following definition, then we show that we lose only a constant fraction of the flow value.

▶ **Definition 16.** *A family of cycles* $\{C_1, C_2, \ldots, C_k\}$ *is* cyclically ordered, *or has a* cyclic order *if, whenever two cycles C_a and C_b share an edge, where $a < b$, then this edge is:*

1. *shared by all cycles $C_a, C_{a+1}, \ldots, C_{b-1}, C_b$,*
2. *or shared by all cycles $C_b, C_{b+1}, \ldots, C_k, C_1, \cdots, C_{a-1}, C_a$.*

The following lemma establishes the approximation ratio of Algorithm 1 on cyclically ordered input.

▶ **Lemma 17.** *Let $\overline{f}$ be a multiflow and $\mathcal{H} = \{C_1, C_2, \ldots, C_k\}$ a cyclically ordered family of $\mathcal{C}(\overline{f})$. Then Algorithm 1 returns in polynomial time an integral multiflow of value at least $|\overline{f}(\{C_1, \ldots, C_k\})|/2$.*

Proof. Let $\overline{f}$ be a multiflow and $\mathcal{H} = \{C_1, C_2, \ldots, C_k\}$ a cyclically ordered family of $\mathcal{C}(\overline{f})$. It is clear that Algorithm 1 runs in polynomial time and returns an integral multiflow. Let f be this flow. We show that its value is at least $|\overline{f}(\mathcal{H})|/2$.

Let us define $\mathcal{H}_{a,b} = \{C_a, C_{a+1}, \ldots, C_{b-1}\}$ and $\mathcal{H}_{b,a} = \{C_b, C_{b+1}, \ldots, C_k, C_1, \ldots, C_{a-1}\}$ for all $1 \leqslant a \leqslant b \leqslant k$. Additionally, for all edges $e \in \bigcup_{C \in \mathcal{H}} C$, we define $\mathcal{H}^e := \{C \in \mathcal{H} \mid e \in C\}$. Since we assumed that $\mathcal{H}$ is cyclically ordered, we know that for each $e \in \bigcup_{C \in \mathcal{H}} C$, there are indexes $1 \leqslant a, b \leqslant k$, such that $\mathcal{H}^e = \mathcal{H}_{a,b}$.

We call i_0 the smallest index $1 \leqslant i \leqslant k$ such that there exists an edge $e \in C_i$ such that $f(\mathcal{H}_{1,i+1})(e) = u(e)$ and $\mathcal{H}_{i,1} \subseteq \mathcal{H}^e$. Remark that in particular, for all $i > i_0$, we must have $f_{C_i} = 0$, and thus $|f| = |f(\mathcal{H}_{1,i_0+1})|$.

We first show by induction that for all $1 \leqslant i < i_0$ we have $|f(\mathcal{H}_{1,i+1})| \geqslant |\overline{f}(\mathcal{H}_{1,i+1})|$. For $i = 1$, we have $|f(\mathcal{H}_{1,i+1})| = |f(\mathcal{H}_{1,2})| = f_{C_1} = \min\{u(e) | e \in C_1\} \geqslant \overline{f}_{C_1} = |\overline{f}(\mathcal{H}_{1,2})|$.

Assume now that at some iteration $1 < i < i_0$ of the algorithm we set $f_{C_i} = x$. By the choice of x, we know that there is an edge $e \in C_i$ such that $u(e) = f(\mathcal{H}_{1,i+1})(e)$. In particular, notice that $|f(\mathcal{H}^e)| = |f(\mathcal{H}^e \cap \mathcal{H}_{1,i+1})| = u(e)$. By feasibility of $\overline{f}$, we have

$$|f(\mathcal{H}^e \cap \mathcal{H}_{1,i+1})| = u(e) \geqslant |\overline{f}(\mathcal{H}^e)|. \tag{5}$$

Now, let a, b be the two indexes such that $\mathcal{H}_{a,b} = \mathcal{H}^e$. Since we assumed that $i < i_0$, we must have $i < b \leqslant k$. There are two cases: either $1 \leqslant a \leqslant i < b$ or $1 < i < b < a$.

If $1 \leqslant a \leqslant i < b$, then equation (5) becomes $|f(\mathcal{H}_{a,i+1})| \geqslant |\overline{f}(\mathcal{H}^e)| \geqslant |\overline{f}(\mathcal{H}_{a,i+1})|$. Together with the induction hypothesis we obtain:

$$|f(\mathcal{H}_{1,i+1})| = |f(\mathcal{H}_{1,a})| + |f(\mathcal{H}_{a,i+1})| \geqslant |\overline{f}(\mathcal{H}_{1,a})| + |\overline{f}(\mathcal{H}_{a,i+1})| = |\overline{f}(\mathcal{H}_{1,i+1})|.$$

Otherwise if $1 < i < b < a$, then $\mathcal{H}_{1,i+1} \subseteq \mathcal{H}^e$, and thus the inequality claimed follows directly from equation (5). We have established the induction. In particular, we have proved that $|f| = |f(\mathcal{H}_{1,i_0+1})| \geqslant |f(\mathcal{H}_{1,i_0})| \geqslant |\overline{f}(\mathcal{H}_{1,i_0})|$. To conclude the proof of Lemma 17, it remains to show that $|f| \geqslant |\overline{f}(\mathcal{H}_{i_0,1})|$.

By definition of i_0, we know that there exists an edge $e \in C_{i_0}$ such that $f(e) = u(e)$ and such that $\mathcal{H}_{i_0,1} \subseteq \mathcal{H}^e$. By feasibility of $\overline{f}$, we deduce that $|\overline{f}(\mathcal{H}_{i_0,1})| \leqslant u(e) = f(e) \leqslant |f|$. This concludes the proof.   ◄

▶ **Remark 18.** The analysis of Algorithm 1 for cyclically ordered inputs is tight. To see this, imagine that $\mathcal{H} = \{C_1, \ldots, C_{2k-1}\}$, and there are two edges e_1, e_2, both of capacity k, such that $\{C \in \mathcal{H} \mid e_1 \in C\} = \{C_1, \ldots, C_k\}$ and $\{C \in \mathcal{H} \mid e_2 \in C\} = \{C_{k+1}, \ldots, C_{2k-1}, C_1\}$. Then Algorithm 1 may only set $f_{C_1} = k$ while $\overline{f}$ could be such that $\overline{f}_C = 1$ for all $C \in \mathcal{H}$, for a total value $2k - 1$.

7.2   Computing a cyclic order

Lemma 20, the second main result of the section, states that a family $\mathcal{H}$ of pairwise freely homotopic cycles crossing at most once can be cyclically ordered in polynomial time. One key ingredient in the proof is that cycles in $\mathcal{H}$ are pairwise non-crossing. This fact uses the assumption that the surface is orientable. In a non-orientable surface, two freely homotopic cycles may cross exactly once.

Recall that $\overline{f}$ denotes the minimally-crossing multiflow obtained by Lemma 7.

▶ **Lemma 19.** *Two freely homotopic cycles in $\mathcal{C}(\overline{f})$ do not cross.*

For the proof of this simple topological fact, see the full version.

▶ **Lemma 20.** *A family of non-separating, pairwise non-crossing and freely homotopic cycles of a graph embedded in an orientable surface can be cyclically ordered. Such a cyclic order can be found in polynomial time.*

This result holds more generally for a family of *non-contractible*[3], pairwise non-crossing and freely homotopic cycles. For simplicity, we only consider the special case of non-separating cycles, which is sufficient for our main result.

Proof. Let $\mathcal{H}$ be a set of non-separating, pairwise freely homotopic and non-crossing cycles. We first order the cycles in $\mathcal{H}$ and then prove that this is a cyclic order. We assume that $|\mathcal{H}| \geqslant 3$, otherwise any order on $\mathcal{H}$ is a cyclic order.

In topology it is usually more convenient to work with disjoint cycles. If two (graph) cycles do not cross, but may share common edges, it is possible to continuously deform by free homotopy one of them, into an arbitrarily small open neighborhood so that the two resulting (topological) cycles are now disjoint.

In the context of graph cycles, we now give a reduction from the setting of Lemma 20 to the special case where the cycles are disjoint. Initially, $Q = G + H$.

---

[3] all cycles that are not freely homotopic to a point on the surface.
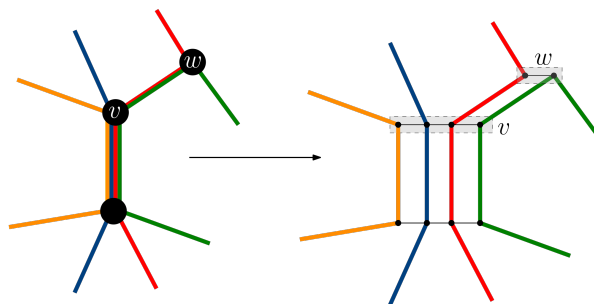
Figure 5 Construction of Q.

Step 1: If an edge is shared by s cycles, replace it s parallel edges. Each of these edges corresponds to a different cycle so that the resulting set of cycles is still pairwise non-crossing. Now the cycles are pairwise edge-disjoint but may still share some vertices.

Step 2: Let v be a vertex shared by two cycles C and C'. Edges incident to v are embedded around v in the cyclic order $e_1, a_1, \ldots, a_i, e_2, b_1, \ldots, b_j$ where $C \cap \delta(v) = \{e_1, e_2\}$. Since C and C' do not cross, we have $C' \cap \delta(v) \subseteq \{a_1, \ldots, a_i\}$ or $C' \cap \delta(v) \subseteq \{b_1, \ldots, b_j\}$. Then replace v by two adjacent vertices v', v'' and distribute the incident edges so that $\delta(v') = (e_1, a_1, \ldots, a_i, e_2, \{v', v''\})$ and $\delta(v'') = (\{v', v''\}, b_1, \ldots, b_j)$. Repeat step 2 until all cycles are vertex-disjoint.

It is easy to see that this graph is connected (since $G + H$ is connected) and can be embedded in the same surface $\mathbb{S}_g$. Figure 5 illustrates the construction of Q. Moreover, a cyclic ordering of the resulting cycles naturally induces a cyclic ordering of H. This completes the reduction. For simplicity, let us also call $\mathcal{H}$ the family of cycles in Q.

In the dual Q^*, let $\mathcal{K}$ denote the set of connected components of $Q^* \setminus \left(\bigcup_{C \in \mathcal{H}} C^* \right)$. They correspond to the connected components of $\mathbb{S}_g \setminus \left(\bigcup_{C \in \mathcal{H}} C \right)$. We say that a cycle $C \in \mathcal{H}$ is *incident* to a connected component $K \in \mathcal{K}$ if there is an edge in C^* with one endpoint in K. Consider the bipartite graph B that has a vertex for each cycle in $\mathcal{H}$ and a vertex for each element of $\mathcal{K}$, and whose edges represent the incidence relation. Next we show that the graph B is a cycle, and we order the D-cycles in $\mathcal{H}$ according to the cyclic order induced by B.

$\triangleright$ **Claim 21.** B is a cycle.

The connectivity of B follows by construction from the connectivity of Q. Then it is enough to prove that this graph is 2-regular.

We first prove that each vertex of B that corresponds to a cycle in $\mathcal{H}$ has degree two in B. Since the cycles in $\mathcal{H}$ are disjoint, each cycle C has one component on its left, and one on its right, when we walk along the cycle. Assume, for a contradiction, that they are the same component: C is incident to only one component of $\mathbb{S}_g \setminus \left(\bigcup_{C \in \mathcal{H}} C \right)$. This cycle is also incident to only one component of $\mathbb{S}_g \setminus (C \cup C')$ where C' is any other cycle in $\mathcal{H}$. By Fact 4, we know that $\mathbb{S}_g \setminus (C \cup C')$ has two connected components. But since C is incident to only one connected component of $\mathbb{S}_g \setminus (C \cup C')$, $\mathbb{S}_g \setminus C'$ must also have two connected components, which contradicts the assumption that C' is non-separating. Thus, each cycle in B must have degree two.

Now we prove that each element of $\mathcal{K}$ has degree two. For a contradiction, assume that an element of $\mathcal{K}$ is incident to three cycles C, C', C'' or more. Then one component of $Q^* \setminus (C \cup C' \cup C'')$ is also incident to C, C' and C'', and $Q^* \setminus (C \cup C' \cup C'')$ has two or three components in total. If it has three components, then one of the other two components

would be incident to exactly one cycle, which would mean that this cycle is separating, a contradiction. If $Q^* \setminus (C \cup C' \cup C'')$ has exactly two connected components, then $Q^* \setminus (C \cup C')$ must be connected which contradicts Fact 4. Thus, each component is incident to exactly two cycles. This concludes the proof of the claim.

It remains to show that the order induced by B satisfies the property of Definition 16. If an edge $e = \{u, v\}$ of $G + H$ is shared by some cycles $C'_1, \ldots, C'_\ell$, then the vertex v can be mapped to a path $P = (v_1, \ldots, v_\ell)$ in Q, so that $C'_i \cap P = \{v_i\}, 1 \leqslant i \leqslant \ell$. See Figure 5. It follows that for all $1 \leqslant i \leqslant \ell - 1$, C'_i and C'_{i+1} are both incident the same connected component of $Q^* \setminus \left(\bigcup_{C \in \mathcal{H}} C \right)$ that contains the edge $\{v_i, v_{i+1}\}^*$. In particular, C'_i and C'_{i+1} are consecutive in the order induced by B.   ◀

8   Proof of Theorem 1

By construction, the output of the algorithm is a feasible solution. We now analyze the value of the output. Since (1) is a relaxation of the maximum integral multiflow problem, $|f^*| \geqslant \mathrm{OPT}$. By Lemma 7, $|\overline{f}| \geqslant (1 - \epsilon)|f^*|$. For $\epsilon = \frac{1}{2}$ we have $|\overline{f}| \geqslant \frac{1}{2}|f^*|$.

Consider the multiflow restricted to separating cycles, $\overline{f}_{\mathrm{sep}}$. If $|\overline{f}_{\mathrm{sep}}| \geqslant \frac{1}{2}|\overline{f}|$, then by Theorem 10, Lemma 11, and Theorem 3 we obtain an integral flow of value at least $|\overline{f}_{\mathrm{sep}}|/\Theta(\sqrt{g})$.

Otherwise, by Theorem 14 there exists a free homotopy class $\mathcal{H}$ of non-separating cycles such that $|\overline{f}(\mathcal{H})| \geqslant |\overline{f}|/\Theta(g^2 \log g)$. Use Lemmas 17 and 20 to obtain that the output has value at least $|f^*|/\Theta(g^2 \log g)$.

Finally, we analyze the running time. As observed in Section 4, an optimum fractional multiflow f^* can be found in polynomial time. (Discretizing and) uncrossing is done in time polynomial in $|E||D|$ by Lemma 7. Partitioning into free homotopy classes is done by Corollary 15. Finally, the operations of Theorem 10, Theorem 3, Lemma 11, Lemma 17 and Lemma 20 can all be done in polynomial time, hence polynomial running time overall.

9   Proof of Corollary 2

In this section, we observe how Corollary 2 follows from Theorem 1 and the following result by Tardos and Vazirani [36] (based on work by Klein, Plotkin and Rao [24]).

▶ **Theorem 22** ([36]). *Let (G, H, u) be a multiflow instance and $\gamma > 1$ such that the supply graph G does not have a $K_{\gamma,\gamma}$ minor. Then the minimum capacity of a multicut is $O(\gamma^3)$ times the maximum value of a (fractional) multiflow.*

The following is well known.

▷ Claim 23.   If a graph G has genus at most g, where $g \geqslant 1$, then it has no $K_{\gamma,\gamma}$ minor for any $\gamma > 2(\sqrt{g} + 1)$.

Proof. Suppose that such a minor $K_{\gamma,\gamma}$ exists in G. As the three operations for obtaining a minor (deleting edges/vertices and contracting edges) do not increase the genus, $K_{\gamma,\gamma}$ has genus at most g. Furthermore, K has 2γ vertices, γ^2 edges, and at most $\frac{\gamma^2}{2}$ faces (since there is no odd cycle in a bipartite graph). By Euler's formula, $2 - 2g \leqslant 2\gamma - \gamma^2 + \frac{\gamma^2}{2}$, which implies $\gamma \leqslant 2(\sqrt{g} + 1)$.   ◁

By Claim 23 and Theorem 22, the ratio between the minimum capacity of a multicut and the maximum value of a (fractional) multiflow is $O(g^{1.5})$. This, combined with Theorem 1, proves Corollary 2.

References

1 Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network Flows*. Prentice-Hall, 1993.

2 Matthew Andrews, Julia Chuzhoy, Sanjeev Khanna, and Lisa Zhang. Hardness of the undirected edge-disjoint paths problem with congestion. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 226–244, 2015.

3 Shuchi Chawla, Robert Krauthgamer, Ravi Kumar, Yuval Rabani, and D. Sivakumar. On the hardness of approximating multicut and sparsest-cut. *Computational Complexity*, 15(2):94–114, 2006.

4 Chandra Chekuri, Sanjeev Khanna, and F. Bruce Shepherd. An $O(\sqrt{n})$ approximation and integrality gap for disjoint paths and unsplittable flow. *Theory of Computing*, 2:137–146, 2006.

5 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.

6 Julia Chuzhoy, David H. K. Kim, and Rachit Nimavat. New hardness results for routing on disjoint paths. In *Proceedings of the 49th Annual ACM Symposium on Theory of Computing Conference (STOC)*, pages 86–99, 2017.

7 Julia Chuzhoy, David H. K. Kim, and Rachit Nimavat. Almost polynomial hardness of node-disjoint paths in grids. In *Proceedings of the 50th Annual ACM Symposium on Theory of Computing Conference (STOC)*, pages 1220–1233, 2018.

8 Vincent Cohen-Addad, Éric Colin de Verdière, and Arnaud de Mesmay. A near-linear approximation scheme for multicuts of embedded graphs with a fixed number of terminals. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1439–1458, 2018.

9 Éric Colin de Verdière. Topological algorithms for graphs on surfaces. In J.E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*. CRC Press, 2017. Chapter 23.

10 Marie-Christine Costa, Lucas Letocart, and Frederic Roupin. Minimal multicut and maximal integer multiflow: A survey. *European Journal of Operational Research*, 162(1):55–69, 2005.

11 Elias Dahlhaus, David S. Johnson, Christos H. Papadimitriou, Paul D. Seymour, and Mihalis Yannakakis. The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23(4):864–894, 1994.

12 David B. A. Epstein. Curves on 2-manifolds and isotopies. *Acta Mathematica*, 115:83–107, 1966. `doi:10.1007/BF02392203`.

13 Jeff Erickson and Kim Whittlesey. Transforming curves on surfaces redux. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1646–1655. SIAM, 2013. `doi:10.1137/1.9781611973105.118`.

14 Samuel Fiorini, Nadia Hardy, Bruce Reed, and Adrian Vetta. Approximate min–max relations for odd cycles in planar graphs. *Mathematical Programming*, 110(1):71–91, 2007.

15 Lester R. Ford and Delbert R. Fulkerson. *Flows in Networks*. Princeton University Press, 1962.

16 Naveen Garg, Nikhil Kumar, and András Sebő. Integer plane multiflow maximisation: Flow-cut gap and one-quarter-approximation. In *Proceedings of IPCO*, pages 144–157, 2020.

17 Naveen Garg, Vijay Vazirani, and Mihalis Yannakakis. Approximate max-flow min-(multi)cut theorems and their applications. *SIAM Journal on Computing*, 25(2):235–251, April 1996.

18 Naveen Garg, Vijay Vazirani, and Mihalis Yannakakis. Primal-dual approximation algorithms for integral flow and multicut in trees. *Algorithmica*, 18(1):3–20, 1997.

19 Joshua Evan Greene. On curves intersecting at most once, 2018. `arXiv:1807.05658`.

20 Percy J. Heawood. Map colour theorem. *Quarterly Journal of Mathematics*, 24:332–338, 1890.

21 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*, 35:752–769, 2021.

22   Richard M. Karp. On the computational complexity of combinatorial problems. *Networks*, 5(1):45–68, 1975.

23   Ken-ichi Kawarabayashi and Yusuke Kobayashi. An $O(\log n)$-approximation algorithm for the edge-disjoint paths problem in Eulerian planar graphs. *ACM Transactions on Algorithms*, 9(2):16:1–16:13, 2013.

24   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)*, pages 682–690, 1993. `doi:10.1145/167088.167261`.

25   Philip N. Klein, Claire Mathieu, and Hang Zhou. Correlation clustering and two-edge-connected augmentation for planar graphs. In *Proceedings of the 32nd International Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 554–567, 2014.

26   Francis Lazarus and Julien Rivaud. On the homotopy test on surfaces. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science, (FOCS)*, pages 440–449. IEEE Computer Society, 2012. `doi:10.1109/FOCS.2012.12`.

27   Justin Malestein, Igor Rivin, and Louis Theran. Topological designs. *Geometriae Dedicata*, 168:221–233, August 2010. `doi:10.1007/s10711-012-9827-9`.

28   Matthias Middendorf and Frank Pfeiffer. On the complexity of the disjoint paths problem. *Combinatorica*, 13(1):97–107, March 1993. `doi:10.1007/BF01202792`.

29   Bojan Mohar and Carsten Thomassen. *Graphs on Surfaces*. Johns Hopkins Series in the Mathematical Sciences. Johns Hopkins University Press, 2001. URL: `http://jhupbooks.press.jhu.edu/ecom/MasterServlet/GetItemDetailsHandler?iN=9780801866890&qty=1&source=2&viewMode=3&loggedIN=false&JavaScript=y`.

30   Guyslain Naves and András Sebő. Multiflow feasibility: an annotated tableau. In W. Cook, L. Lovász, and J. Vygen, editors, *Research Trends in Combinatorial Optimization*, pages 261–283. Springer, 2009.

31   Piotr Przytycki. Arcs intersecting at most once. *Geometric and Functional Analysis*, 25:658–670, February 2015. `doi:10.1007/s00039-015-0320-0`.

32   Neil Robertson, Daniel P. Sanders, Paul Seymour, and Robin Thomas. Efficiently four-coloring planar graphs. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing (SODA)*, pages 571–575, 1996.

33   Neil Robertson and Paul D. Seymour. Graph minors. XIII. The disjoint paths problem. *Journal of Combinatorial Theory, Series B*, 63(1):65–110, 1995.

34   Alexander Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*. Springer, 2003.

35   András Sebö. Integer plane multiflows with a fixed number of demands. *Journal of Combinatorial Theory, Series B*, 59(2):163–171, 1993. `doi:10.1006/jctb.1993.1062`.

36   Éva Tardos and Vijay V. Vazirani. Improved bounds for the max-flow min-multicut ratio for planar and $K_{r,r}$-free graphs. *Information Processing Letters*, 47:77–80, August 1993.