Approximate Duality of Multicommodity Multiroute Flows and Cuts: Single Source Case

Petr Kolman* Christian Scheideler†

Abstract

Given an integer h, a graph $G = (V, E)$ with arbitrary positive edge capacities and k pairs of vertices $(s_1, t_1), (s_2, t_2), \dots, (s_k, t_k)$, called terminals, an h-route cut is a set $F \subseteq E$ of edges such that after the removal of the edges in F no pair $s_i - t_i$ is connected by h edge-disjoint paths (i.e., the connectivity of every $s_i - t_i$ pair is at most $h - 1$ in $(V, E \setminus F)$). The h-route cut is a natural generalization of the classical cut problem for multicommodity flows (take $h = 1$). The main result of this paper is an $O(h^7 2^{2h} \log^2 k)$-approximation algorithm for the minimum h-route cut problem in the case that $s_1 = s_2 = \cdots = s_k$, called the single source case. As a corollary of it we obtain an approximate duality theorem for multiroute multicommodity flows and cuts with a single source. This partially answers an open question posted in several previous papers dealing with cuts for multicommodity multiroute problems.

1 Introduction

The theory of networks and cuts and flows in networks is among the oldest areas of combinatorial optimization, dating back at least to the time of Gustav Kirchoff. Almost every student of computer science or mathematics has to come across Menger's theorem or the more general maximum flow – minimum cut theorem (Ford and Fulkerson [7], Elias, Feinstein and Shannon [6]): The maximum value of a flow from a source vertex to a sink vertex in a network equals the minimum capacity among all cuts between the source and the sink. Although this theorem perfectly describes the relation between classical flows and cuts in networks with a single commodity, there are many other (more general) reasonable settings where the theorem is not applicable. A natural generalization of the classical flows and cuts are the *multicommodity* flows and cuts; although an exact duality does not hold for them, an approximate max-flow min-cut relationship does [8], cf. [12].

Another generalization of the classical flows and cuts are the *multiroute* flows and cuts [10], cf. [1]. A classical flow is a non-negative linear combination of unit flows along simple paths (after removing any cycles). An *h-route flow* is a non-negative linear combination of elementary h-route flows where an *elementary h-route flow* is a set of h edge-disjoint paths between a source and sink, each path carrying a unit of flow. An *h-route cut* is a set of edges whose removal decreases the maximum h-route flow to zero.

There are many papers dealing with the problem of multiroute flows (also known, e.g., as multi-path routing) and multiroute cuts. One reason for this interest are the possible applications of multiroute flows in communication networks (multiroute flows are more resilient against link failures and adversaries [2, 9, 13]). Also, it has been considered an important open problem whether the approximate duality of multicommodity flows and cuts does extend to *multiroute* multicommodity flows and cuts. There are several partial results in this respect. Bruhn et al. [4] proved an approximate duality theorem for single-source multicommodity h-route flows on uniform capacity networks that applies for any $h > 1$; the approximation factor is linear in h and does not depend on k. Chekuri and Khanna [5] described an approximation algorithm for the minimum multicommodity 2-route cut problem whose corollary is an approximate duality theorem for multicommodity 2-route cuts and flows on networks with arbitrary edge capacities. An analogous result was recently shown by Kolman and Scheideler [11] for 3-route multicommodity flows and cuts. The approximation factors in the last two results mentioned above are polylogarithmic in k. Another polylogarithmic approximation for the minimum multicommodity 2-route cut problem (without any explicit implications for the approximate duality) was given by Barman and Chawla [3]; they also provided several bicriteria approximations.

On the lower bound side, the known $\Omega(\log k)$ gap between the classical maximum multicommodity flow and the minimum cut carries over to the multiroute setting. Apart from this, a lower bound $\Omega(h)$ for the same gap is known [3].

*Fakulty of Mathematics and Physics, Charles University of Prague, Czech Republic. Partially supported by the grants GA ČR 201/09/0197 and 1M0021620808 (ITI).

†Dept. of Computer Science, University of Paderborn, Germany. Partially supported by the grants DFG SCHE 1592/1-1 and DFG SFB 901.

It is worth mentioning that in general there is no relationship between the sizes of a minimum h-route cut and a minimum $(h-1)$-route cut, apart from the obvious fact that the former is not larger then the latter. In particular, it may happen that the size of a minimum h-route cut is zero while the size of a minimum $(h-1)$-route cut is arbitrarily large.

1.1 New results. The main result of this paper is an $O(h^7 2^{2h} \log^2 k)$-approximation algorithm for the minimum h-route cut problem with single source for any $h \geq 1$. As a corollary of it we obtain an approximate duality theorem for multiroute multicommodity flows and cuts for that case. This partially answers an open question posted in several previous papers dealing with cuts for multicommodity multiroute problems [5, 4, 3, 11]. Note that we focused on showing an approximation ratio that is polylogarithmic in k, at the cost of obtaining an exponential dependency on h. However, we suspect that a much lower dependency on h is possible.

1.2 Comparison with previous related results. At a high level, the structure of our approximation algorithm is the same as the structure of the approximation algorithm by Garg et al. [8] for classical multicommodity cuts (cf. [14, 3, 11]): while there exists a commodity whose terminals are still h-connected, perform an h-route cut separating the terminals of this commodity, charge the h-route cut to a certain part (volume) of the network (roughly, to the region that was used to define the cut) and proceed with the next iteration. The h-route cut in each iteration is derived from the fractional solution of the LP relaxation of the minimum h-route cut problem (see below) via the ball growing technique. The fractional solution of the LP relaxation is computed only once at the very beginning of the algorithm and the same solution (with some minor modifications due to local structural changes of the graph G during the algorithm) is used in all iterations.

Though the three above mentioned approximation algorithms [8, 3, 11] and our new algorithm share the same structure, including the usage of the ball growing technique, on the low level, our new algorithm substantially differs from all of the three older algorithms.

In the classical setting (i.e., for $h = 1$), after performing a cut around an appropriate ball with center in t_i, the vertices s_i and t_i belong to different components of connectivity. This way the problem is split by the cut into two independent subproblems.

For $h = 2$, after performing a 2-route cut for the commodity i, the vertices s_i and t_i will typically still belong to the same connected component but there always exists a cut consisting of a single edge e that separates s_i from t_i. Let S and T be the two components of the graph after removing e. We note that even without removing e, there is no simple path in G originating and terminating in S and going through T (and vice versa). Thus, similar to $h = 1$, the two parts S and T of the graph induce two independent subproblems (with respect to 2-route flows and cuts) even though S and T belong to the same connected component. It was this independence that made it possible to design the charging scheme for $h = 2$ [3].

For $h = 3$ the situation is different. After performing a 3-route cut for commodity i, we know that there exists a cut consisting of at most two edges whose removal disconnects s_i and t_i. Similarly to the previous case, let S and T be the two parts of G obtained by removing the two edges. In contrast to $h \leq 2$, now there may exist paths originating in S and terminating in S that are going through T (and vice versa). Thus, the parts S and T do not induce independent subproblems any more. To cope with this problem, the approximation algorithm keeps track of the balls constructed in previous iterations. Fortunately, as S and T are connected by two edges only, the interactions between S and T are of a very specific form which makes it possible to cope with the minimum h-route cut problem for $h = 3$ [11]. The *multilevel* ball growing technique was introduced here as the main tool. On the technical level, after dealing recursively with all commodities with both terminals in T (say), it is possible to replace (the ball) T by a single edge (with endpoints being equal to the starting points of the two edges leading from S into T) which substantially simplifies the situation for later iterations.

For $h > 3$, the interactions between S and T (defined as above) are of much more complicated form and so are the interactions between the balls from previous iterations. One difficulty is that the balls cannot be substituted by single edges any more (not even by a tree, due to the many restrictive constraints: lengths of paths, sizes of cuts, etc.). A related difficulty is the following: occasionally, for an appropriate ball (i.e., an edge, after the substitution), the algorithm for $h = 3$ needs to decrease the connectivity between two *entry* nodes of the ball (i.e., between the nodes that connect the ball to the rest of the graph). As the maximum number of edge-disjoint paths in the ball between the two entry nodes of the ball is just one, decreasing the connectivity means performing an ordinary cut of the ball. After doing so, each of the two resulting components of the ball is connected to the rest of the graph by a single edge only, and thus, as explained earlier, the components correspond to independent subproblems and can be removed from the graph. However, for $h > 3$, after decreasing the connectivity between two subsets of entry nodes of a given ball, typically the ball will not split into two separated parts (think about a multiroute cut of the ball), and even if it does, both parts of the ball may contain two or more entry edges. Thus, we cannot remove them from the graph. However, this significantly complicates the charging mechanism.

The main technical contribution of this paper is an extension of the multilevel ball growing technique to the case of $h > 3$ with single source. Among various ideas is a technique to prevent edges within certain subgraphs (consisting of balls from previous iterations) to be cut or charged even though the original ball growing argument would demand that. To make the treatment of the balls possible, we maintain them in a laminar set system.

1.3 Preliminaries.

Suppose we are given a graph $G = (V, E)$ with edge capacities $c : E \to \mathbb{R}_+$ and commodities $(s_1, t_1), \ldots, (s_k, t_k)$. Let $\mathcal{P}_i$ denote the set of all paths in G between s_i and t_i. In the single source version of the problem, the *sources* of all commodities are in the same vertex s, that is, $s_i = s$ for each commodity i. Our approximation algorithm is based on the linear programming relaxation of the minimum h-route cut problem that was recently used to deal with the problem for $h = 2$ [3] and $h = 3$ [11]. For every edge e there is a variable $x(e) \in \{0, 1\}$ indicating whether this edge belongs to the h-route cut, and for each commodity and each edge there is a variable $x_i(e) \in \{0, 1\}$ serving as a proof that the edges with $x(e) = 1$ form really an h-route cut for the commodity i; the proof consists in showing that $x + x_i$ corresponds to a classical cut with at most $h - 1$ edges with $x_i(e) = 1$. The minimum h-route cut problem can be stated as an integer linear program (ILP) as follows:

$$(1.1) \qquad \min \sum_{e \in E} c(e)x(e)$$

$$\begin{aligned} \sum_{e \in p} (x(e) + x_i(e)) &\geq 1 & \forall i \in [k], p \in \mathcal{P}_i \\ \sum_{e \in E} x_i(e) &\leq h - 1 & \forall i \in [k] \\ x(e) &\in \{0, 1\} & \forall e \in E \\ x_i(e) &\in \{0, 1\} & \forall i \in [k], \forall e \in E \end{aligned}$$

In order to find a good approximate solution for this ILP, we work with the linear programming (LP) relaxation of it where the constraints $x(e) \in \{0, 1\}$ and $x_i(e) \in \{0, 1\}$ are replaced by $x(e) \geq 0$ and by $x_i(e) \geq 0$, resp., for every edge e and every commodity i. We denote by Φ the value of the objective function for the optimal solution of the LP relaxation.

Given a fractional solution x and x_i, shortest paths with respect to x or $x + x_i$ naturally induce a metric on V; previous papers that exploit the ball growing technique are based on these metrics. In this paper we again work with metrics that are derived from the LP relaxation but we loosen the connection with the shortest paths. More precisely, throughout the algorithm and for each $i \in [k]$, $y_i(u)$ is the length of the shortest path in the input graph G, with respect to $x + x_i$, between t_i and the node $u \in V$. Important properties of the mappings y_i are that for each edge $uv \in E$, $|y_i(u) - y_i(v)| \leq x(uv) + x_i(uv)$ and that $y_i(s_i) \geq 1$ for all i.

We will need the following simple lemma (implicitly used in a simpler form by Garg et al. [8]).

Lemma 1.1. (Ball Growing Lemma [11]) *Let $[l_1, r_1], [l_2, r_2], \ldots, [l_z, r_z]$ be internally disjoint intervals of real numbers such that $l_1 < l_2 < \cdots < l_z$ and let $\mathcal{R} = \bigcup_{i=1}^{z} [l_i, r_i]$. Assume that the following holds:*

- *f is a nondecreasing function on $\mathcal{R}$ and $f(l_1) > 0$,*
- *f is differentiable on $\mathcal{R}$, except for finitely many points,*
- *g is a function on $\mathcal{R}$ such that $\forall r \in \mathcal{R}$, $g(r) \leq f'(r)$, except for finitely many points.*

Let $\gamma = f(r_z)/f(l_1)$. Then there exists $r \in \mathcal{R}$ such that $g(r) \leq \frac{1}{|\mathcal{R}|} \log \gamma \cdot f(r)$. Also, there exists $r \in \mathcal{R}$ such that $g(r) \leq \frac{1}{|\mathcal{R}|} \cdot f(r_z)$.

Kolman and Scheideler [11] proved that if I is an upper bound on the integrality gap of the linear program (1.1), F is the size of the maximum multicommodity h-route flow and C is the size of the minimum multicommodity h-route cut, then

$$(1.2) \qquad \frac{F}{h} \leq C \leq I \cdot F \quad .$$

By describing, for the single source version of the problem, an $O(h^7 2^{2h} \log^2 k)$-approximation algorithm based on the LP (1.1), we also show that the integrality gap of the LP for the single source case is $O(h^7 2^{2h} \log^2 k)$ which in turn implies the approximate duality theorem.

Without loss of generality we assume that $h \leq k$ and $k \geq 2$ (otherwise an $O(h^2)$-approximation is trivial [11]). For a graph H, we use $E(H)$ to denote the edge set and $V(H)$ to denote the node set of H. Similarly, for a set D of (sub)graphs, we define $E(D) = \bigcup_{H \in D} E(H)$ and $V(D) = \bigcup_{H \in D} V(D)$. For a set F of edges we define $c(F) = \sum_{e \in F} c(e)$.

2 Rounding Algorithm

In the single source version of the problem, the *sources* of all commodities are in the same vertex s, that is, $s_i = s$ for each commodity i. As we already mentioned in the introduction, the algorithm works in iterations and in iteration i an h-route cut for commodity i is constructed (if it does not already exist). Similarly to the previous papers on multi-cut approximations, the h-route cut for commodity i is obtained by the ball-growing technique; roughly, it consists of cutting all but $h-1$ edges leaving a ball around t_i with a properly chosen radius r. We recall that the cuts constructed by the multicut algorithm for classical multicommodity flows [8] are of the form

$$\delta(r) \quad = \quad \{uv \in E \ \mid \ y_i(u) \le r < y_i(v)\} \ .$$

In iteration i, we view every edge $uv \in E$ as a segment of length $|y_i(u) - y_i(v)|$ consisting of two parts: an x_i-part of length (at most) $x_i(uv)$ followed by an x-part of length (at most) $x(uv)$ on the way from t_i. (Note that the ordering of x and x_i is different from our previous paper [11] since this particular ordering is more convenient for our analysis here.) We define

$$\delta_x(r) \quad = \quad \{uv \in E \ \mid \ y_i(u) + x_i(uv) \le r < y_i(v)\} \ .$$

Note that for every $r \in (0, 1)$, if $|\delta(r) \setminus \delta_x(r)| \le h - 1$, then $\delta_x(r)$ is an h-route cut between t_i and s. Our intention is, roughly, for each commodity i to find a radius r such that $\delta_x(r)$ is a cheap h-route cut between s_i and t_i. Before describing the algorithm in more detail, we introduce some notation and several ingredients that are important for our algorithm.

2.1 Levels. In our algorithm, edges may be charged several times for different h-route cuts. Therefore we maintain for every edge e a counter $\ell(e)$ called the *level* of the edge, which is an upper bound on the number of times the edge was charged for a cut (i.e., for a removal of some edges). Initially, $\ell(e) = 0$ for every edge $e \in E$. We will organize the cutting and charging in such a way that the level of every edge is upper bounded by $2h + \log k$ (see Lemma 2.1 bellow). The use of levels was the main innovation of our multilevel ball growing technique [11] but here it is much more tricky to apply.

2.2 Restricted structures. In order to keep the edge levels small, when constructing a new h-route cut, we try to avoid charging the cost of the new cut to edges of high level. For this we maintain a set D of *restricted structures* which are subgraphs of G. Roughly speaking, the restricted structures in iteration i will coincide with balls constructed by the algorithm around the terminals $t_1, \ldots, t_{i-1}$ in previous iterations. The restricted structures in D form a laminar family, that is, for any two subgraphs $H_1, H_2 \in D$, $H_1 \subset H_2$, or $H_2 \subset H_1$, or $H_1 \cap H_2 = \emptyset$. For any $H \in D$ we denote by $D(H)$ the restricted structures from D that are subgraphs of H, that is, $D(H) = \{H' \in D \ \mid \ H' \subsetneq H\}$. Initially, $D = \emptyset$. The set of all edges that connect $H \in D$ to the rest of G are called its *entry edges* and are denoted by $in_E(H)$, and the set of endpoints of the entry edges that are inside of H are called *entry nodes* and denoted by $in_V(H)$. We define the *level of* H as

$$\ell(H) \quad = \quad \max\{\ell(e) \ \mid \ e \in E(H) \setminus \bigcup_{H' \in D(H)} E(H')\} \ .$$

If $E(H) \setminus \bigcup_{H' \in D(H)} E(H') = \emptyset$ then H can be removed from D as all of its edges are already covered by other restricted structures and the level of these structures is guaranteed by Invariant 2.1 to be larger than H.

A related notion of *restricted edge* appeared in our paper about 3-route cuts [11]. In that setting it was possible to keep the restricted structures disjoint and to reduce them to individual edges which significantly simplified their treatment. Here, to control the interactions between the restricted structures, we associate with every $H \in D$ a *credit*

$$\phi(H) = 2^{2|in_E(H)| + \ell(H)} \ , \qquad (2.3)$$

and the restricted structures must satisfy the following invariant.

INVARIANT 2.1. *At the end of iteration i the restricted structures in D form a laminar family of subgraphs of G such that for all $H \in D$,*

$$2 \le \quad |in_E(H)| \quad \le h - 1 \ , \ \textit{and} \qquad (2.4)$$

$$\ell(H) \quad < \quad \ell(H') \ \ \textit{for all } H' \in D(H) \ , \ \textit{and} \qquad (2.5)$$

$$\sum_{H \in D} \phi(H) \quad \le \quad i \cdot 2^{2h} \ . \qquad (2.6)$$

Note that if $|in_E(H)| \le 1$, we can remove H from G as no simple path can enter H and leave it again. If $\ell(H) \ge \ell(H')$ for some $H' \in D(H)$, then we can remove H' from D and raise the level of its edges to $\ell(H)$ without affecting $\phi(H)$. Under the assumption that Invariant 2.1 holds, we observe a simple fact.

LEMMA 2.1. *For every edge e it holds that $\ell(e) \le 2h + \log i$ at the end of iteration i.*

Proof. The bound is implied by the definition (2.3) and constraints (2.4) and (2.6) in Invariant 2.1 which require that $2^{2 \cdot 2 + \ell(e)} \le i \cdot 2^{2h}$ for any edge e. □

In the rest of this section we explain iteration i of our algorithm step by step.

2.3 Partitioning of D. Earlier in this section we outlined our plan to avoid cutting and charging edges of high level. This was an (unrealistic) oversimplification. Instead, roughly speaking, we will allow cutting and charging edges in restricted structures of level ℓ if we ensure that we cut or charge at least *two* restricted structures of level ℓ. By doing so we obtain a tree of restricted structures with logarithmic depth. This is the motivation for partitioning the set D into two parts as described below; the set D_1 is prohibited for cuts and charges while the set D_2 is allowed for cuts and charges. The exact partitioning procedure is not based on the number of restricted structures of certain level but, instead, on the sum of potentials of restricted structures of this level.

At the beginning of iteration i we partition the set D of restricted structures into two parts, D_1 and D_2. For each level ℓ we check whether $\sum_{H\in D:\ell(H)=\ell}\phi(H)\geq 2h^2 2^{2h+\ell}$. If so, we put to D_1 the minimum number of restricted structures of level ℓ whose credit sums up to at least $2h^2 2^{2h+\ell}$, starting with the closest to t_i (w.r.t. $\min_{v\in V(H)} y_i(v)$). If not, we put to D_1 all restricted structures of level ℓ. This is done for every level. After processing all levels, we put all remaining restricted structures to D_2, that is, we set $D_2 = D\setminus D_1$. Based on the restriction on D_1 we get:

LEMMA 2.2. *For each level ℓ, $\sum_{H\in D_1:\,\ell(H)=\ell}|in_E(H)| \leq h^2 2^{2h-1}$.*

Proof. From the definition of the credit it follows that $\sum_{H\in D_1:\ell(H)=\ell}\phi(H) \leq (2h^2+1)2^{2h+\ell}$ and thus $\sum_{H\in D_1:\ell(H)=\ell} 2^{2|in_E(H)|} \leq (2h^2+1)2^{2h}$. It is easy to check that the maximum value of $\sum_{H\in D_1:\,\ell(H)=\ell}|in_E(H)|$ is reached when $|in_E(H)|=2$ for every H, which results in an upper bound of $2\cdot(2h^2+1)2^{2h}/2^{2\cdot 2}\leq h^2 2^{2h-1}$ for $\sum_{H\in D_1:\,\ell(H)=\ell}|in_E(H)|$. □

2.4 Preprocessing of D_1. Assume that we are in iteration i and H is a restricted structure from D_1. Let $R_H = [\min_{u\in in_V(H)} y_i(u), \max_{u\in in_V(H)} y_i(u))$. For any subgraph H' of H with the same vertex set and for $r\in R_H$ let $\text{mincut}_{H'}(r)$ be the edge set of the classical single-commodity cut of minimum *cardinality* in H' between $S(r)=\{v\in in_V(H)\mid y_i(v)\leq r\}$ and $T(r)=in_V(H)\setminus S(r)$ (i.e., all edges have cost 1 when evaluating the cost of cuts). For notational simplicity, for $r\notin R_H$ we define $\text{mincut}_{H'}(r)=\emptyset$. The following lemma shows that for every $H\in D_1$, by removing a few edges from $E(H)$ (the set F_H that we can pay for by the volume of H) and excluding a short subinterval from R_H (the interval I_H) we ensure that on average the mincuts of $H\setminus F_H$ are "small" (as the sum of x_is is small). This will allow us to avoid cutting edges in restricted subgraphs in D_1 even though the standard ball growing technique would demand that.

LEMMA 2.3. *For every $H\in D_1$ there is a subset $F_H\subseteq E(H)$ and a subset $I_H\subseteq R_H$ so that for $H'=H\setminus F_H$,*

$$\int_{r\in R_H\setminus I_H}|\text{mincut}_{H'}(r)|\,dr \leq \sum_{e\in E(H)} x_i(e)\ , \tag{2.7}$$

$$\sum_{e\in F_H} c(e) = O(h^5 2^{2h})\sum_{e\in E(H)} x(e)c(e)\ , \tag{2.8}$$

$$|I_H|\leq |in_V(H)|/(2h^4 2^{2h}\log k)\ . \tag{2.9}$$

Moreover, the sets I_H and F_H can be computed in polynomial time.

Proof. For $h'=|in_V(H)|$, let $u_1,\ldots,u_{h'}$ be the entry nodes of H in increasing order of $r_j=y_i(u_j)$ and let $I_j=[r_j,r_{j+1})$. We proceed through the intervals I_j one by one, starting with I_1, and for each of them we update the sets I_H and F_H. Initially, we set $I_H=\emptyset$ and $F_H=\emptyset$. We use H' as a shortcut for $H\setminus F_H$ and $\delta_x^{H'}(r)$ as a shortcut for $\delta_x(r)\cap E(H')$.

Assume that we are dealing with the interval I_j. Let $l=|\text{mincut}_{H'}(r_j)|$, and for any $l'\leq l$ let $\mathcal{F}_j^{l'}$ be the set of all tuples of l' edge-disjoint paths in H between the node sets $S(r_j)$ and $T(r_j)$. For each $B\in\mathcal{F}_j^{l'}$ we define $I_B=\{r\in I_j\mid \delta_x^{H'}(r)\cap B\neq\emptyset\}$, that is, I_B is the set of radii for which $\delta(r)$ hits some edge from $B\cap E(H')$ in its x-part. We distinguish between two cases.

1. If $\min_{B\in\mathcal{F}_j^l}|I_B|\leq 1/(2h^4 2^{2h}\log k)$, we extend I_H by $I_{B'}$ where B' is a set from $\mathcal{F}_j^l$ for which the minimum is achieved. Clearly,

   $$|I_{B'}|\leq 1/(2h^4 2^{2h}\log k)\ . \tag{2.10}$$

   This fact will be used to guarantee the property (2.9). Also notice that

   $$\begin{aligned}\int_{r\in I_j\setminus I_{B'}}\delta_{x_i}^{H'}(r)\ dr &\geq \int_{r\in I_j\setminus I_{B'}} l\ dr\\ &= \int_{r\in I_j\setminus I_{B'}}|\text{mincut}_{H'}(r)|\ dr\end{aligned} \tag{2.11}$$

   so property (2.7) holds for the subinterval $I_j\setminus I_H$.

2. If $\min_{B\in\mathcal{F}_j^l}|I_B| > 1/(2h^4 2^{2h}\log k)$, we let l' be the minimum index for which $\min_{B\in\mathcal{F}_j^{l'}}|I_B| > 1/(2h^4 2^{2h}\log k)$. Suppose that $l'\geq h$. In this case we define $\text{mincut}_{H'}(r_j)$ to be equal to the entry edges of $T(r_j)$. Thus, instead of going through $S(r_j)$ and $T(r_j)$, we would fully include H in a

ball with radius in I_j, which is sufficient for all of our arguments later in this section to apply. Then it holds that

$$\int_{r\in I_j\setminus I_{B'}} \delta_{x_i}^{H'}(r)\ dr \geq \int_{r\in I_j\setminus I_{B'}} |\mathrm{mincut}_{H'}(r)|\ dr$$

where B' is a set from $\mathcal{F}_j^{l'-1}$ with $|I_{B'}| = \min_{B\in\mathcal{F}_j^{l'-1}} |I_B| \leq 1/(2h^4 2^{2h}\log k)$.

If $l' < h$, then we perform a cut through $S(r_j)$ and $T(r_j)$ in the following way. The vector x scaled by $h\cdot 2h^4 2^{2h}\log k$ represents a feasible fractional solution of a linear programming relaxation, analogous to the linear program (1.1), of an l'-route cut between $S(r_j)$ and $T(r_j)$ in H' (see the paper [11] for a detailed description how to obtain the actual fractional solution). Combining Theorem 14 in [11] and the last statement of the Ball growing lemma together with Lemma 2 in [11], we obtain in polynomial time an l'-route cut F_j between $S(r_j)$ and $T(r_j)$ of size

$$(2.12)\ \sum_{e\in F_j} c(e) = O(h^6 2^{2h}\log k)\int_{r\in I_j} c(\delta_x^H(r))\ dr\ .$$

We add the edges from F_j to F_H and adjust H' accordingly. Since now $\mathrm{mincut}_{H'}(r_j) \leq l'-1$ and since, by the assumption of this case, $\min_{B\in\mathcal{F}_j^{l'-1}} |I_B| \leq 1/(2h^4 2^{2h}\log k)$, we can continue as in the case 1 to complete our dealing with the interval I_j.

Combining the inequalities (2.10), (2.11) and (2.12) over all intervals I_j yields the lemma. □

Lemma 2.3 ignores the fact that we may have restricted structures inside of H. As we have a laminar family of restricted structures, we have to be a bit more careful in choosing F_H so that the charging in the algorithm only affects structures that are actually cut. Consider any $H \in D_1$. Instead of just focusing on the entry nodes of H, we consider all entry nodes in $\bar{in}_V(H) = in_V(H) \cup \bigcup_{H'\in D_1(H)} in_V(H')$ (where $D_1(H)$ denotes the set of maximal D_1-structures contained in H, i.e., they are not contained in any other D_1-structure in H); by Lemma 2.2 we know that their number is at most $O((h+\log k)\cdot h^2 2^{2h-1})$. We order the nodes in $\bar{in}_V(H)$ according to increasing y_i-distance from t_i, call the resulting distances $r_1,\ldots,r_{|\bar{in}_V(H)|}$, and then apply the proof of Lemma 2.3 to them. Taking over the notation in that proof and renaming F_j to $\bar{F}_j$, F_H to $\bar{F}_H$, I_H to $\bar{I}_H$ and I_j to $\bar{I}_j$ to avoid confusion with the parameters in Lemma 2.3, this results in the following property.

Corollary 2.1. *For every $H \in \bar{D}_1$ with its edge set reduced to $E'(H) = E(H)\setminus E(D_1(H))$ there is a subset $\bar{F}_H = \bigcup_{j=1}^{|\bar{in}_V(H)|-1} \bar{F}_j \subseteq E'(H)$ and a subset $\bar{I}_H = \bigcup_{j=1}^{|\bar{in}_V(H)|-1} \bar{I}_j \subseteq R_H$ so that for $H' = H\setminus \bar{F}_H$,*

$$(2.13)\quad \int_{r\in R_H\setminus \bar{I}_H} |\mathrm{mincut}_{H'}(r)|\, dr \leq \sum_{e\in E'(H)} x_i(e)\ ,$$

$$(2.14)\quad \sum_{e\in\bar{F}_j} c(e) = O(h^5 2^{2h}\log k)\int_{r\in\bar{I}_j} c(\delta_x^H(r))\, dr\ ,$$

$$(2.15)\quad |\bar{I}_H| \leq |\bar{in}_V(H)|/(2h^4 2^{2h}\log k)\ .$$

Moreover, all of the involved sets can be computed in polynomial time.

Note that by the choice of the strips $\bar{I}_j$ in Corollary 2.1 it holds for each strip $\bar{I}_j$ that a structure $H' \in D(H)$ either has nodes on both sides of the strip or none of the edges in $E(H')$ will be cut by $\bar{F}_j$ (which also means it does not have to be charged for that cut).

In the preprocessing phase, we temporarily remove from every $H \in \bar{D}_1$ the subset of edges $\bar{F}_H$ as given by Corollary 2.1 in order to be able to apply the ball growing technique. In particular, we define $\bar{F}_i = \bigcup_{H\in\bar{D}_1} \bar{F}_H$ and $I = \bigcup_{H\in\bar{D}_1} \bar{I}_H$ and $\bar{G}_i = G_i\setminus\bar{F}_i$ where G_i denotes the graph G at the beginning of iteration i. The cost of these edges is charged to the volume of the edges that contribute to the integral on the righthand side of the equation (2.14). Later, for most $H \in \bar{D}_1$, we will put the set $\bar{F}_H$ back into $\bar{G}_i$; the details are provided later in this section. We stress that we do not increase the level of any edge at this point.

2.5 h-route cuts. The crucial idea for the 3-route cut algorithm [11] was to consider only radii that do not cut edges of structures in D_1 in their x-parts; as D_1 was just an edge set in that case, it was relatively easy to show that this restriction does not reduce the set of possible radii too much. Here, Corollary 2.1 will help us to mimic the same approach. We are looking for h-route cuts in $\bar{G}_i$ of the following form: $\delta_h(r) = \delta_x(r)\setminus E(D_1)$. We also define $\bar{\delta}_h(r) = (\delta_{x_i}(r)\setminus E(D_1))\cup \bigcup_{H\in\bar{D}_1} \mathrm{mincut}_H(r)$. Note that $\delta_h(r)$ need not be an h-route cut between s and t_i, and that $\delta_h(r)\cup\bar{\delta}_h(r)$ is a classical cut between s and t_i. Thus, if $|\bar{\delta}_h(r)| \leq h-1$ then $\delta_h(r)$ is an h-route cut. Let $\mathcal{R}$ be the set

$$\mathcal{R} = \{r \in [0,1]\setminus I \ \mid\ \delta_h(r) \text{ is an } h\text{-route cut}\}\ .$$

We say that $\mathcal{R}$ is the set of *good radii* for $\bar{G}_i$. Below we prove that the measure of the set $\mathcal{R}$ is large which will make it possible to apply the Ball Growing Lemma (Lemma 1.1), and thus, to find a cheap h-route cut between t_i and s.

Lemma 2.4. (Good radii)
$|\mathcal{R}| \geq 1/(2h)$.

Proof. We first prove that

$$\int_{r\in[0,1]\setminus I} |\bar{\delta}_h(r)|\ dr \leq h-1\ .$$

Think about the edges that contribute to the value of the integral. They can be divided into two subsets: edges from $E(D_1)$, and edges from $E \setminus E(D_1)$. By Corollary 2.1, the contribution of the first group is at most $\sum_{e\in E(D_1)} x_i(e)$; the contribution of the other group is at most $\sum_{e\in E\setminus E(D_1)} x_i(e)$. Since the vectors x and x_i constitute a fractional solution of the linear program (1.1), the sum of these two upper bounds is at most $h-1$.

Once we have the above bound, we easily conclude that the measure of the set $X = \{r \in [0,1] \setminus I \mid |\bar{\delta}_h(r)| > h-1\}$ is at most $1-1/h$; just observe that $h|X| = \int_{r\in X} h\ dr \leq \int_{r\in[0,1]\setminus I} |\bar{\delta}_h(r)| \leq h-1$.

Now, it follows from Corollary 2.1 and Lemmas 2.1 and 2.2 that

$$\begin{aligned}\sum_{H\in\bar{D}_1} |\bar{I}_H| &\leq \sum_{H\in\bar{D}_1} |\bar{in}_V(H)|/(2h^4 2^{2h}\log k)\\ &\leq (\sum_{H\in D_1} |in_V(H)|)/(2h^4 2^{2h}\log k)\\ &\leq (2h\log k)\cdot(h^2 2^{2h-1})/(2h^4 2^{2h}\log k)\\ &= 1/(2h)\end{aligned}$$

Hence, $|I| \leq 1/(2h)$. Since $[0,1]\setminus(X\cup I) \subseteq \mathcal{R}$, the claim of the lemma follows. □

We are almost ready to apply the ball growing lemma. We define the volume function

$$Vol(r) = \frac{\Phi}{k} + \int_{\rho\in\mathcal{R}\cap[0,r]} c(\delta_h(\rho))\ d\rho\ ,$$

where $\Phi = \sum_{e\in E} c(e)x(e)$. Since the functions $f(r) := Vol(r)$ and $g(r) := c(\delta_h(r)) = \sum_{e\in\delta_h(r)} c(e)$ satisfy the assumptions of the ball growing lemma on the set of good radii $\mathcal{R}$, we obtain the following lemma (cf. [11]).

Lemma 2.5. (Cheap h-route cut) *There exists $r \in \mathcal{R}$ such that $c(\delta_h(r)) \leq 2h\log(2k)\cdot Vol(r)$. Moreover, such a radius can be computed in polynomial time.*

As the h-route cut between s and t_i in $\bar{G}_i$ we use the set $F_i = \delta_h(r)$ where r is the radius from the lemma. Note that according to the definition of $\delta_h(r)$ no edges in $\bar{\delta}_h(r)\cup E(D_1)$ contribute towards the volume $Vol(r)$ (the edges in $\bar{\delta}_h(r)\setminus E(D_1)$ are cut in their x_i part, and by our assumption on the ordering of the x- and x_i-parts, the x_i-parts precede the x-parts on the way from t_i). We remove F_i from $\bar{G}_i$ (which may only remove edges in D_2-structures that are not contained in any D_1-structure) and proceed with updating D so that Invariant 2.1 is preserved.

In the following let r be the radius from Lemma 2.5, let $B(r) = \{u \in V \mid y_i(u) \leq r\}$ and let $H(r)$ be the subgraph of $\bar{G}_i$ induced by the node set $B(r)$. We define the entry edges of $H(r)$ as $in_E(H(r)) = \bar{\delta}_h(r)$. Our intention is to add $H(r)$ to D. However, when doing this, we may no longer have a laminar family of restricted structures, and levels of some structures in D may have to increase as they get charged for establishing an h-route cut for $H(r)$. We handle these issues in three stages: first, we update the credit of restricted structures to reflect their increased level for certain cases in which this is directly possible. Then we repair the laminar property of the restricted structures, and finally we update the credit of the remaining restricted structures whose level got increased.

2.6 Pre-updating the credit. Due to (2.6) in Invariant 2.1 we can afford to set the level of $H(r)$ to 1 (see the level definition) as the total credit can increase by 2^{2h} in each iteration and $|in_E(H(r))| \leq h-1$. Hence, if there are no restricted structures that intersect with $H(r)$, we are done.

Suppose now that there is a restricted structure $H' \in D$ with $V(H') \cap V(H(r)) \neq \emptyset$ and $V(H') \not\subseteq V(H(r))$ (i.e., H' intersects with $H(r)$ but is not fully contained in it). In this case, there is also a maximal restricted structure $H \in \bar{D}$ (for example, the one containing H) with that property. Suppose that H is cut by $H(r)$ into two disconnected subgraphs H_1 and H_2 (i.e., there is no edge connecting H_1 and H_2), where H_1 is inside of $H(r)$ and H_2 is outside. Then we consider the following cases:

1. Both H_1 and H_2 have at least two entry edges. Then it follows from $|in_E(H_1)| + |in_E(H_2)| \leq |in_E(H)|$ that $|in_E(H_1)| \leq |in_E(H)| - 2$ and $|in_E(H_1)| \leq |in_E(H)| - 2$, which implies that

$$\begin{aligned}&2^{2|in_E(H_1)|+(\ell+1)} + 2^{2|in_E(H_2)|+(\ell+1)}\\ &\leq (2^{2(|in_E(H)|-2)} + 2^{2(|in_E(H)|-2)})2^{\ell+1}\\ &\leq 2^{2|in_E(H)|-2+(\ell+1)} = \phi(H)/2\end{aligned}$$

   Thus, H can be split into H_1 and H_2 with levels by one larger than H, which can be paid for by the credit of H. In fact, half of the credit of H still remains, which we will make use of below.

2. H_1 (resp. H_2) has exactly one entry edge. Then we can remove H_1 (resp. H_2) from D and from G

as it is not relevant any more for future h-route cuts and we can raise the level of the remaining H_2 (resp. H_1) by 1 with the credit of H. In fact, also here half of the credit of H still remains, which we will make use of below.

3. H_1 does not contain any entry edge. Then we can remove H_1 from D and from G as well as it is isolated. If H is a D_1-structure, we undo all mincuts in H_2 to make sure that H_2 is not charged for cuts. If H is a D_2-structure, H_2 is not charged because it is outside of $H(r)$. Hence, in both cases we can leave the level of H_2 at the level of H.

4. H_2 does not contain any entry edge. Then we undo any cutting inside of H and extend $H(r)$ by H. If H is a D_1-structure, then H (as well as every $H' \in D(H)$) is not charged in this case, so we do not have to raise its level (or any level of its substructures). If H is a D_2-structure, we deal with it below.

If $D(H) \neq \emptyset$ and cases 1-3 were applied to H, we deal with any $H' \in D(H)$ that is cut by $H(r)$ into two disconnected subgraphs in the order of increasing level in the same way as we dealt with H, i.e., we apply the cases above to them. If a part of a structure got thrown away due to case 2 or 3 of some previously considered structure, we treat it as having no entry edge in that part. In this way, all of these structures receive a proper credit update reflecting their current level (except for those D_2-structures satisfying case 4). Also, we repaired the laminar set property for these structures. Note that when using the cases above, we never shrink $H(r)$ but possibly extend it (when case 4 is applied). This will be important when dealing with the D_2-structures that are (partly) charged for the cut F_i, which we are considering next.

Suppose that there is a D_2-structure H that is (partly) charged for the cut F_i. Let $\ell = \ell(H)$ and let $D(\ell)$ (resp. $D_1(\ell)$) be the set of all $H' \in D$ (resp. $H' \in D_1$) of level ℓ with $V(H') \cap B(r) \neq \emptyset$. We know from our rule of partitioning D that $\sum_{H \in D_1(\ell)} \phi(H) \geq 2h^2 2^{2h+\ell}$. Now, every structure $H' \in D_1(\ell)$ satisfies one of the following properties: (a) H' is inside of $H(r)$, (b) H' is inside of some maximal structure $H'' \in \bar{D}$ that is not cut by $H(r)$ into disconnected subgraphs, or (c) H' is inside of some maximal structure $H'' \in \bar{D}$ that is cut into disconnected subgraphs by $H(r)$. In the first two cases, we remove H' from D and thereby free up all of its credit, and in case (c) we make use of the property that at least one entry edge of H' must be inside of $H(r)$ (otherwise the D_2-structure H cannot be (partly) charged for $H(r)$ because for this at least one entry node of H must be closer to t_i than r, which then also applies to H'). Hence, in case (c) we can make use of the fact that either case 1 or 2 above applied to H', so at least half of its credit can be taken away from it. Hence, we can free up a total credit of at least

$$h^2 \cdot 2^{2h+(\ell+1)}$$

For $h \geq 3$ this is enough credit to increase the level of $H(r)$ as well as all maximal structures $H'' \in \bar{D}$ containing some $H' \in D(\ell)$ that are not disconnected into two subgraphs by $H(r)$, which can be at most $h-1$ many, to $\ell+1$. This is why we can remove all $H' \in D_1(\ell)$ that satisfy property (a) or (b) above because now they are contained in a restricted structure with a larger level then themselves. Also, note that we can remove all D_1- and D_2-structures of level at most ℓ that are inside of $H(r)$ (such as those to which we applied case 4 above) for the same reason, which resolves the case 4.

At the end of this phase, we have dealt with updating the levels of all D_2-structures intersecting with $H(r)$, and all D_1-structures that are inside of H have also been dealt with (by either undoing any cutting in them so that they do not need to be charged or by removing them from D). Thus, it remains to consider the D_1-structures that intersect with $H(r)$ but are not fully contained in it. Also, we still need to repair the laminar set property (which we have only done so far for maximal structures that are cut into two disconnected subgraphs by $H(r)$ and their substructures). This is done next.

2.7 Getting back to a laminar family of restricted structures. Let $D' = \{H \in D \mid V(H) \cap B(r) \neq \emptyset$ and $V(H) \not\subseteq B(r)\}$. In order to avoid the intersection of $H(r)$ with the restricted structures from D', we process the restricted structures in D' in the order of their (original) level, starting with restricted structures of the lowest level, and for each such problematic $H \in D'$ we perform either a local change of H or a local change of $H(r)$. For notational simplicity, for a restricted structure H we define $\bar{E}(H) = E(H) \cup in_E(H)$, and similarly $\bar{E}(H(r)) = E(H(r)) \cup in_E(H(r))$. For each restricted structure H from D' we define three sets (see Figure 1): $A_H = in_E(H) \setminus \bar{E}(H(r))$, $B_H = in_E(H(r)) \cap \bar{E}(H)$, $C_H = in_E(H) \cap \bar{E}(H(r))$. Based on the quantities of A_H, B_H and C_H, we distinguish between three types of intersections of H with $H(r)$.

1) $|B_H| < |A_H|$ and $|B_H| < |C_H|$: In this case, H is replaced in D by its two subgraphs: $H_1 = H \cap H(r)$ with $in_E(H_1) = C_H \cup B_H$ and $H_2 = H \setminus H_1$ with $in_E(H_2) = B_H \cup A_H$. If in addition $H \in \bar{D}_1$, then the edges belonging to $mincut_H(r)$ in $\bar{F}_H$ are left removed and all other edges in $\bar{F}_H$ are given

Figure 1: How to deal with different types of intersection of $H(r)$ and $H \in D'$.

back to $\bar{G}_i$, i.e., $\bar{F}_i$ is updated to $(\bar{F}_i \setminus \bar{F}_H) \cup mincut_H(r)$ (with $E'(H)$ being still equal to $E(H)$ in Corollary 2.1).

2) $|A_H| \leq |B_H|$: In this case, $H(r)$ is extended by the nodes and edges of H, the edges in B_H are removed from $in_E(H(r))$ and the edges in A_H are added there. We also remove from D' all restricted structures belonging to $D(H)$. For any structure $H' \in D$ with $H \in D(H')$ that was previously applied to case 1), we extend $V(H'_1)$ by $V(H)$ and remove all nodes in $V(H)$ from $V(H'_2)$ to repair the laminar property of D. Note that this does not increase the number of entry edges of H'_1 or H'_2.

3) $|C_H| \leq |B_H| < |A_H|$: In this case, the nodes and edges from $H(r) \cap H$ are removed from $H(r)$ (i.e., $H(r) = H(r) \setminus H$). We also remove from D' all restricted structures belonging to $D(H)$. For any structure $H' \in D$ with $H \in D(H')$ that was previously applied to case 1), we extend $V(H'_2)$ by $V(H)$ and remove all nodes in $V(H)$ from $V(H'_1)$. Note that this does not increase the number of entry edges of H'_1 or H'_2.

Clearly, after we have processed D', we again have a laminar family of restricted structures. Let D'' denote the set of restricted structures for which the removal of some edges in the set $\bar{F}_H$ was made permanent in case 1), and let D''' denote the set of restricted structures that were excluded from $H(r)$ as described in case 3). Let $\bar{F}_i = \bigcup_{H \in D''} \text{mincut}_H(r)$. With this notation, the set of edges added to the global h-route cut in iteration i corresponds to $F_i \cup \bar{F}_i$, and we note that $F_i \cup \bar{F}_i$ is an h-route cut between t_i and s in G_i. We denote by $G_{i+1} = G_i \setminus (F_i \cup \bar{F}_i)$ the graph entering the next iteration.

2.8 Post-updating the credit. Finally, we show how to handle the level increase of those D_1-structures that are still cut by $\bar{F}_i$. For this we need the following lemma.

LEMMA 2.6. *For every $H \in D_1$ that was replaced by H_1 and H_2 in D as described in case 1) it holds for the original level ℓ of H that*

$$2^{2|in_E(H_1)|+(\ell+1)} + 2^{2|in_E(H_2)|+(\ell+1)} \leq 2^{2|in_E(H)|+\ell}$$

Proof. Note that in case 1) it holds that $|in_E(H_1)| + |in_E(H_2)| < |in_E(H)|$. Hence,

$$\begin{aligned} & 2^{2|in_E(H_1)|+(\ell+1)} + 2^{2|in_E(H_2)|+(\ell+1)} \\ = \; & (2^{2|in_E(H_1)|} + 2^{2|in_E(H_2)|})2^{\ell+1} \\ \leq \; & 2^{2|in_E(H)|-1+(\ell+1)} \end{aligned}$$

As the number of entry edges of H_1 and H_2 does not increase due to repairs of the laminar property for other restricted structures, the lemma follows. □

Therefore, we can give enough credit to H_1 and H_2 from the credit of H to pay for the level increase.

Finally, we show how to make sure that all restricted structures in D are connected at the end. Consider some restricted structure $H \in D$ that is cut into two or more connected components due to the removal of edges in $F_i \cup \bar{F}_i$. If $H_1, \ldots, H_d$ are its connected components with at least 2 entry edges, then H is replaced in D by $H_1, \ldots, H_d$ (connected components with at most 1 entry edge can be removed from consideration). The entry edges of the connected components are inherited from H and therefore, $\sum_{j=1}^{d} in_E(H_j) \leq in_E(H)$. Hence,

$$\sum_{j=1}^{d} 2^{2|in_E(H_j)|+\ell(H)} \leq 2^{2|in_E(H)|+\ell(H)} = \phi(H)$$

which implies that we can keep the level of these structures at $\ell(H)$ with the credit of H. Therefore, we can preserve the levels of the edges while making sure that all restricted structures in D are connected.

2.9 Summing it up. Every time we add a subset F of edges to the global cut, we identify a set of edges whose x-volume is "large" (i.e., proportional to $\sum_{e \in F} c(e)$, up to a factor of $O(h^6 2^{2h} \log k)$) and we increase by one the level of each of them. This is implied by Lemma 2.5, Lemma 2.3 and description of the algorithm in the previous paragraphs. At the same time we guarantee that the level of every edge is at most $2h + \log k$ (Invariant 2.1 and Lemma 2.1).

THEOREM 2.1. *The approximation ratio of the algorithm for the h-route single-source cut problem is $O(h^7 2^{2h} \log^2 k)$.*

COROLLARY 2.2. *For every instance of the multicommodity single-source h-route flow problem,*

$$\frac{F}{h} \leq C \leq O(h^7 2^{2h} \log^2 k) F \tag{2.16}$$

where F is the size of the maximum h-route flow for the instance and C is the size of the minimum h-route cut.

3 Conclusion

In this paper we showed an approximate duality theorem for multiroute multicommodity flows and cuts for the single source case. It remains a challenging open problem whether it is possible to obtain an analogous result in the general setting with multiple sources. Currently, the best bound we have been able to obtain is of the order $O(\log^h k)$ (when ignoring h-factors).

References

[1] Charu C. Aggarwal and James B. Orlin. On multiroute maximum flows in networks. *Networks*, 39:43–52, 2002.

[2] Amitabha Bagchi, Amitabh Chaudhary, Michael T. Goodrich, and Shouhuai Xu. Constructing disjoint paths for secure communication. In *Proceedings of the 17th International Conference on Distributed Computing*, volume 2848 of *Lecture Notes in Computer Science*. Springer Verlag, 2003.

[3] Siddharth Barman and Shuchi Chawla. Region growing for multi-route cuts. In *Proceedings of the 21th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2010.

[4] Henning Bruhn, Jakub Černý, Alexander Hall, Petr Kolman, and Jiří Sgall. Single source multiroute flows and cuts on uniform capacity networks. *Theory of Computing*, 4(1):1–20, 2008. Preliminary version in Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 2007.

[5] Chandra Chekuri and Sanjeev Khanna. Algorithms for 2-route cut problems. In *Proceedings of the 35th International Colloquium on Automata (ICALP)*, volume 5125 of *Lecture Notes in Computer Science*, pages 472–484, 2008.

[6] P. Elias, A. Feinstein, and C. Shannon. A note on the maximum flow through a network. *Information Theory, IRE Transactions on*, 2(4):117 –119, 1956.

[7] L. R. Ford and D. R. Fulkerson. Maximum flow through a network. *Canad. J. Math.*, 8:399–404, 1956.

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

[9] Koushik Kar, Murali Kodialam, and T. V. Lakshman. Routing restorable bandwidth guaranteed connections using maximum 2-route flows. *IEEE/ACM Transactions on Networking*, 11:772–781, 2003.

[10] Wataru Kishimoto. A method for obtaining the maximum multiroute flows in a network. *Networks*, 27:279–291, 1996.

[11] Petr Kolman and Christian Scheideler. Towards duality of multicommodity multiroute cuts and flows: Multilevel ball-growing. In *28th International Symposium on Theoretical Aspects of Computer Science (STACS)*, Leibniz International Proceedings in Informatics (LIPIcs), Dagstuhl, Germany, 2011. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[12] Tom Leighton and Satish Rao. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *Journal of the ACM*, 46(6):787–832, November 1999. Preliminary version in Proceedings of the 29th Annual IEEE Symposium on Foundations of Computer Science (FOCS), 1988.

[13] M. Martens. A simple greedy algorithm for the k-disjoint flow problem. In *6th Annual Conference on Theory and Applications of Models of Computation (TAMC)*, volume 5532 of *Lecture Notes in Computer Science*, pages 291–300, 2009.

[14] David B. Shmoys. Cut problems and their application to divide-and-conquer. In Dorith S. Hochbaum, editor, *Approximation Algorithms for NP-hard Problems*, pages 192–235. PWS Publishing Company, 1997.