Delay-constrained Unicast and the Triangle-Cast Problem

Chandra Chekuri
Dept. of CS
UIUC
chekuri@illinois.edu

Sudeep Kamath
ECE Department
Princeton University
sukamath@princeton.edu

Sreeram Kannan
Dept. of EE
Univ. of Washington, Seattle
ksreeram@uw.edu

Pramod Viswanath
Dept. of ECE
UIUC
pramodv@illinois.edu

***Abstract*—We consider the single-unicast communication problem in a network with a delay constraint. For this setting, it has recently been shown that network coding offers an advantage over routing. We show that the existing upper bound in the literature on the capacity offered by network coding can be a factor of $\Theta(D)$ larger than the true capacity where D is the delay bound.**

In this work, we tighten this gap significantly to $8\log(D+1)$ by proving a new upper bound. The key insight is a connection to a new traffic model that we call triangle-cast (or degraded multiple-unicast), for which we obtain a logarithmic flow-cut gap by suitably adapting the techniques from the approximation algorithms literature.

***Index Terms*—delay-constrained unicast, generalized network sharing bound, triangle-cast, region-growing lemma**

I. Introduction

Understanding the information capacity of networks is a central object of study in network information theory. The simplest family of networks is the class of wireline networks. Network coding has been shown to have an advantage over routing in networks with multiple sources or multiple destinations [1]–[3]. This is in contrast to the single-unicast setting where communication between a *single* source to a *single* destination; here routing (without any need for coding) achieves the optimal capacity. This follows from the max-flow min-cut theorem and the fact that the min-cut is also an upper bound on the capacity [4].

The single-unicast problem takes a surprising twist when the information is perishable with a delay constraint, namely, if a message from the source needs to be received by the destination within D time instants from its origin at the source (for simplicity we assume that each edge has a unit delay). It was recently demonstrated that network coding can beat routing for delay-constrained unicast by means of an example [5]. A natural question then is to quantify the advantage of coding. We set up some notation to address this more formally. Given a network G and a source node s and destination node t and a delay constraint D, let F denote the maximum flow from s to t over paths of length at most D. Let C be the delay D constrained capacity between s and t. Examples from [5] demonstrate that C can be a factor of 2 larger than F.

How large can C be compared to F in the worst case? F can be computed in polynomial-time via a linear programming formulation and hence is reasonably well understood. On the other hand, we do not have a good understanding of C. We can attempt to upper bound C by considering a cut in the delay-constrained setting. Abusing notation, we can define Min-Cut to be the capacity of the smallest capacity set of edges whose removal disconnects all paths of length at most D from the source to the destination. [5] shows that the Min-Cut thus defined is actually an upper bound on the information capacity, even in the delay-limited setting. However, we demonstrate, via a family of networks, that the Min-Cut bound can be $\Theta(D)$ factor larger than the capacity.

Our main result in this paper is a new upper bound on C that significantly tightens the relationship between F and C and is captured by the theorem below.

Theorem 1. *In any network G and for any source-destination pair in G,*

$$\frac{C}{8\log(D+1)} \leq F \leq C. \tag{1}$$

Our result provides a converse proof by bringing together ideas from information theory and multicommodity flow-cut gap results on graphs. First, we provide a new outer bound on the delay-constrained network capacity by viewing the time-expanded network as a multiple-unicast problem and invoking the Generalized Network Sharing bound [6].

Next, we formulate a novel traffic pattern that we call *triangle-cast* that is of independent interest. Here there are k sources $s_1, s_2, \ldots, s_k$ and k destinations $d_1, d_2, \ldots, d_k$. The ordering of the sources and destinations is important; for each $1 \leq i \leq k$, source s_i has independent messages to be sent to each destination d_j for every $j \leq i$. From the approximation algorithms literature, it is known that the worst-case flow-cut gap

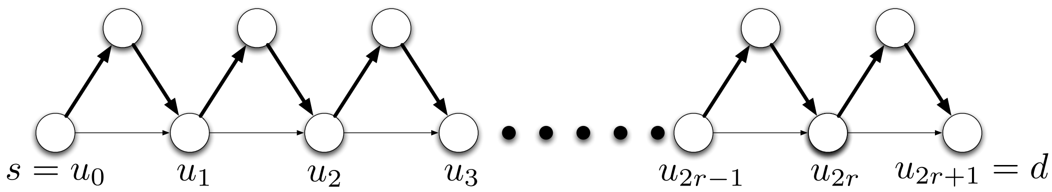

Fig. 1. A unicast network with delay-constraint $D = 3r + 1$. The bold *flank* edges have infinite capacity and the light *base* edges have unit capacity.

for multiple-unicast in directed networks is large: in particular, the gap can be polynomial in both the number of flows and the number of nodes in the graph [7]. However, as we will show, the picture improves substantially for the triangle-cast problem even when the network is directed. This will also be the correct level of abstraction from which we may view the 'regularity' in the specific multiple-unicast problem that comes from time-expansion of the delay-constrained network. By adapting with suitable modifications tools from the flow-cut gap literature in graph theory, we show a logarithmic flow-cut gap result for the triangle-cast pattern. Finally, we connect the earlier mentioned outer bound for the delay-constrained unicast problem to the triangle-cast problem to complete the converse.

Random linear coding is a canonical network coding scheme, which has been shown to be optimal and robust for multicasting in networks [3]. However, in the delay-limited context, it can be readily seen that random network coding is ineffective. Therefore the delay-limited setting can also be seen as a testing ground for understanding structured coding schemes.

The paper is organized as follows: Sec. II provides the motivating example illustrating a large gap between the known inner and outer bounds. Sec. III introduces the time-expanded graph and our new outer bound for the delay-limited problem. Sec. IV defines a new traffic pattern *triangle-cast* and establishes a new flow-cut gap result for this problem. Sec. V puts together these ideas from Sec. III and Sec. IV to establish our main result Theorem 1. Finally, we provide some concluding remarks and open questions in Sec. VI.

II. Example Network

To motivate the need for improved bounds, we consider an example showing that the gap between the known inner and outer bounds can be very large.

Consider the delay-constrained unicast network example in Fig. 1, which is a simple modification of the network example in [8, Fig. 2]. Here, r is a positive integer parameter. The source s needs to send a sequence of messages to the destination d which should be received within a delay constraint of $D = 3r + 1$ units of time.

Let us look at the following routing scheme. Consider the set of all paths from s to d that use exactly $r+1$ base edges and $2r$ flank edges (that are drawn in bold). Along each such path, send a flow of $\frac{1}{\binom{2r}{r}}$. Since each base edge lies on exactly $\binom{2r}{r}$ paths, its capacity constraint is precisely met. The total flow sent from s to d is $\frac{\binom{2r+1}{r+1}}{\binom{2r}{r}} = \frac{2r+1}{r+1}$. We can also show that this is the maximum flow by choosing each base edge to have weight $\frac{1}{r+1}$ and each flank edge to have weight 0 which yields a fractional cut, which is an upper bound on flow by LP duality.

Any cut that disconnects all paths of length at most D from the source to destination must cut at least $r + 1$ of the base edges and this suffices, so Min-Cut $= r + 1$.

This shows that Max-Flow and Min-Cut can be very far apart in the delay-limited setting. In the next section, we develop a new outer bound on the capacity for the delay-constrained unicast problem that is a significant improvement on the Min-Cut. We will show in Sec. III-A using this new outer bound that flow is optimal for this specific example, so that network coding can provide no gains.

III. Time-Expanded Network and new outer bound

Consider the single-unicast delay-constrained problem on a given directed network. By viewing the time-expanded version of the network (Fig. 2), we observe that this is in fact, a multiple-unicast problem in disguise.

Given an edge-capacitated directed network $G = (V, E)$, with vertex set V, edge set E, and edge capacities given by a function $c(\cdot, \cdot)$ specified by the collection $\{c(u, v) : (u, v) \in E\}$. Let $s, d \in V$ be the source and destination node respectively for a unicast flow with delay constraint D. Let F and C be the max-flow and network capacity for the given network with delay constraint D. Let us define the time-expanded network.

Time-expanded Network: Let $G_k = (V_k, E_k)$ denote a k-unicast network with edge-capacity function $c_k(\cdot, \cdot)$ which corresponds to the $(k + D)$-fold time-expansion of the graph G (as illustrated in Fig. 2), where

$$V_k = \{u^{(i)} : u \in V, 1 \leq i \leq k + D\},$$
$$E_k = \{(u^{(i)}, w^{(i+1)}) : (u, w) \in E, 1 \leq i \leq k + D - 1\}$$
$$\cup \{(u^{(i)}, u^{(i+1)}) : u \in V, 1 \leq i \leq k + D - 1\},$$
$$c_k(u^{(i)}, w^{(i+1)}) = \begin{cases} c(u, w) & \text{if } (u, w) \in E, \\ \infty & \text{if } u = w. \end{cases}$$

The source-destination pairs for the k-unicast time-expanded network are given by $s_i = s^{(i)}, d_i = d^{(i+D)}$ for $1 \leq i \leq k$. Let F_k, C_k denote the maximum sum flow rate, sum capacity for the k-unicast network G_k. Then,

$$F = \limsup_k \frac{F_k}{k}, \quad C = \limsup_k \frac{C_k}{k},$$

which we will formally adopt as the definitions for F and C.

Lemma 1.

$$F \geq \frac{F_r}{r + D}, \ r = 1, 2, 3, \ldots.$$

Proof of Lemma 1. Consider the time-expanded graph G_k with $k = t(r + D) - D$ where t, r are some positive integers. This network has $t(r + D)$ layers, each layer containing a copy of the nodes of G. Consider any routing scheme that achieves a sum-rate flow F_r from sources $s^{(1)}, \ldots, s^{(r)}$ to destinations $d^{(D+1)}, \ldots, d^{(D+r)}$ respectively. This routing scheme may be duplicated to achieve another sum flow rate of F_r from $s^{(l(r+D)+1)}, \ldots, s^{(l(r+D)+r)}$ to $d^{(l(r+D)+D+1)}, \ldots, d^{((l+1)(r+D))}$ for each $l = 1, 2, 3, \ldots, t - 1$. This duplication is possible because the edges used in routing in each of these subgraphs are disjoint. This proves that $F_{t(r+D)-D} \geq tF_r$. Hence,

$$F = \limsup_k \frac{F_k}{k} \geq \limsup_t \frac{F_{t(r+D)-D}}{t(r + D) - D} \geq \frac{F_r}{r + D}.$$

$\square$

Our new outer bound is obtained by applying the Generalized Network Sharing (GNS) bound to the time-expanded graph.

Theorem 2. *[6] For an edge-capacitated multiple-unicast network $G = (V, E)$ with k flows and source-destination pairs $(s_i, d_i), i = 1, 2, \ldots, k$, a set of edges $H \subseteq E$ forms a GNS-cut if $\mathcal{G} \setminus H$ has no paths from source s_i to destination d_j whenever $j \leq i$. If H is a GNS-cut, then the sum capacity is upper bounded by $\sum_{(u,v)\in H} c(u, v)$.*

Let GNScut_k denote the smallest $\sum_{(u,v)\in H} c(u, v)$ over all GNS-cuts H in G_k. Our new outer bound may be stated as follows:

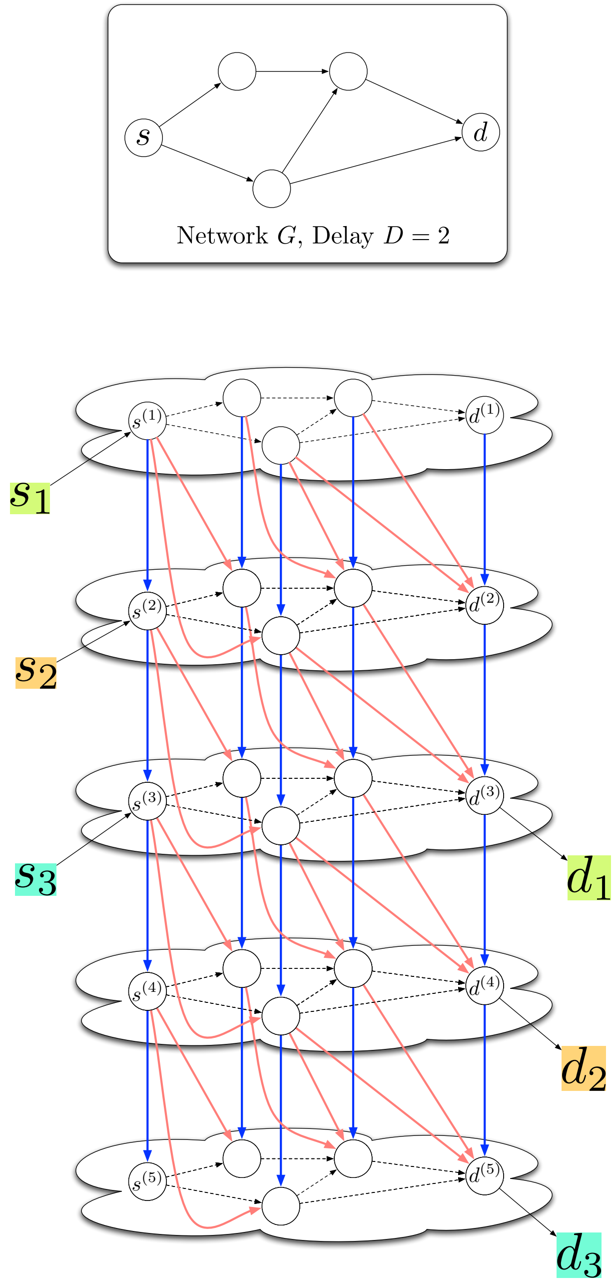

Fig. 2. An example network graph G with all edges having unit capacity. Also shown its time-expanded version G_3. Blue edges have infinite capacity, red edges have unit capacity. The dotted edges do not exist in G_3, they are drawn to illustrate the underlying skeleton of the original graph.

Theorem 3.

$$C \leq \frac{\text{GNScut}_r}{r}, \ r = 1, 2, 3, \ldots.$$

Proof of Theorem 3. Consider the time-expanded graph G_k with $k = tr$ where t, r are some positive integers. This network has $tr + D$ layers, each layer containing a copy of the nodes of G. Consider the set of edges that form a GNS-cut for the flows from $s^{(1)}, \ldots, s^{(r)}$ to destinations $d^{(D+1)}, \ldots, d^{(D+r)}$ respectively. This cut may be duplicated to similarly disconnect the flows from $s^{(lr+1)}, \ldots, s^{((l+1)r)}$ to $d^{(lr+D+1)}, \ldots, d^{((l+1)r+D)}$ for each $l = 1, 2, 3, \ldots, t-1$. Since each destination has an infinite capacity path to all future destinations, disconnecting s_i from d_i ensures disconnection from s_i to d_j for $j \le i$. Therefore, this duplication provides a GNS-cut in G_{tr}. By Theorem 2, $C_{tr} \le t(\text{GNScut}_r)$. As C_k is increasing in k, we get for $0 \le l \le r-1$, $C_{tr+l} \le C_{(t+1)r} \le (t+1)\text{GNScut}_r$. Hence,

$$C = \limsup_k \frac{C_k}{k} \le \frac{\text{GNScut}_r}{r}.$$

□

A. GNS-cut for the example network in Fig. 1

Consider the network in Fig. 1 and consider its time-expanded version G_{r+1}. Consider the set of edges $H = \{(u_j^{(j+1)}, u_{j+1}^{(j+2)}) : j = 0, 1, 2, \ldots, 2r\}$. For $i = 1, 2, \ldots, r+1$, all paths from $s_i = s^{(i)}$ to $d_i = d^{(D+i)}$ are disconnected by removal of the edges $\{(u_j^{(j+1)}, u_{j+1}^{(j+2)}) : j = i-1, i, i+1, \ldots, i+r-1\}$. As d_j has an infinite capacity path to d_i for $j \le i$, this means s_i is also disconnected from d_j for $j \le i$ by the chosen set of edges. Thus the set of edges H forms a GNS-cut and by Theorem 3, we get $C \le \frac{2r+1}{r+1}$. This shows that flow is optimal and network coding has no advantage for this example network.

IV. Triangle-Cast traffic pattern

By viewing the time-expanded version of the network (Fig. 2), we observed in the previous section that the single-unicast delay-constrained problem is a multiple-unicast problem in disguise. But the multiple-unicast network so-formed has a very specific structure. One of the structures present in this network is that all information available at a destination d_j is also available at all destinations d_i when $j \le i$. Thus, a source may deliver the message to the desired destination at a delay precisely equal to the delay-constraint or it may deliver the message to any previous destination i.e. may deliver the message earlier if it prefers. This allows us to formulate a novel traffic pattern that we call *triangle-cast*: in an arbitrary directed graph, we have k source-destination pairs $(s_i, d_i), 1 \le i \le k$, but there are $\binom{k}{2}$ independent information flows; each source s_i has independent information to be communicated to each destination d_j for $j \le i$ (see Fig. 3). The name for this traffic pattern is derived from the upper-triangular form of the matrix of desired flows.

For such a k-triangle-cast problem in an arbitrary directed network, let $F_k^{\triangle}$ be the maximum achievable

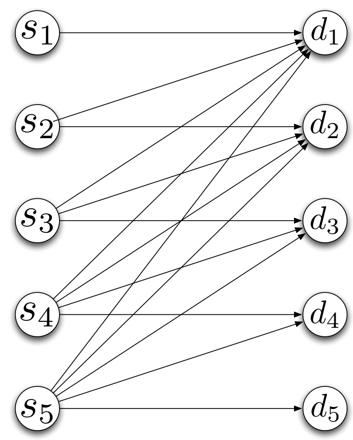

Fig. 3. Triangle-Cast: each source s_i has independent information to be communicated to each destination d_j for $j \le i$.

sum flow rate and let $\text{Cut}_k^{\triangle}$ be the smallest sum capacity of edges to be removed to disconnect s_i from d_j whenever $j \le i$. Using ideas from a 'region-growing' lemma [9] with suitable modifications, we show a strong flow-cut approximation guarantee for the triangle-cast traffic pattern. All logarithms in this paper are natural logarithms.

Theorem 4.

$$\frac{\text{Cut}_k^{\triangle}}{4\log(k+1)} \le F_k^{\triangle} \le \text{Cut}_k^{\triangle}. \tag{2}$$

The proof of this theorem is placed in the appendix. This approximation guarantee for the triangle-cast traffic pattern is in stark contrast to the flow-cut gap in the multiple-unicast traffic pattern in directed graphs which can be polynomially large in k and n [7].

Remark 1. [10] provides an algorithm that approximates the Generalized Network Sharing (GNS) bound in Theorem 2 for a k-unicast network within a factor $O(\log^2 k)$. Our proof of Theorem 4 involves randomization but can be derandomized to produce a deterministic approximation algorithm to the GNS bound for a k-unicast network within a factor $4\log(k+1)$.

V. Proof of our main result

Proof of Theorem 1. Consider the time-expanded network G_k with $k = r$ for any positive integer r. By Lemma 1 and Theorem 3, we have

$$\frac{F_r}{r+D} \le F \le C \le \frac{\text{GNScut}_r}{r}. \tag{3}$$

Viewing this network as a r-triangle-cast problem, Theorem 4 gives

$$\frac{\text{Cut}_r^{\triangle}}{4\log(r+1)} \le F_r^{\triangle} \le \text{Cut}_r^{\triangle}. \tag{4}$$

Noting that $F_r^{\triangle} = F_r$ and $\mathrm{Cut}_r^{\triangle} = \mathrm{GNScut}_r$ and using (3) and (4) gives us that for any r, we have

$$\frac{C}{\left(\frac{4(r+D)\log(r+1)}{r}\right)} \leq F \leq C \ . \tag{5}$$

Choosing $r = D$ completes the proof.

□

VI. Conclusion

We showed a new outer bound for the capacity of delay-limited sigle-unicast in wireline networks using a connection to a novel traffic model, called triangle-cast. This outer-bound shows that routing is within a $8\log(D+1)$ factor of the capacity. We can potentially tighten the constants by optimizing our analysis (for instance, by choosing $r = 2D$ in (5), we get a $6\log(2D+1)$ factor), but to obtain a constant gap result will require new ideas.

- For the delay-limited unicast problem, the examples in [5] show that network coding can beat routing by a factor of 2. A key open question then is the following: Could this factor of 2 be tight? More generally, could the flow-capacity gap be a universal constant? It is possible that the upper-bound presented in our paper has a constant gap to the flow and our bounding technique can be potentially improved. However, demonstrating a constant gap may also require some new techniques. For example, while the triangle-cast model captures some key features of the delay limited problem, it does miss one feature: the fact that the communication graph is time invariant.
- Is the logarithmic flow-cut gap for the triangle-cast problem tight? Could there be a constant flow-cut gap for this problem? There is a related traffic model called groupcast where a group of nodes want to communicate to each other in a directed network - an independent message from each node to every other node in the group - and the sum-rate is the parameter of interest. In this context, it is known [11] that flow and cut are within a factor 2.
- *Interference Channel with Triangle-Cast*: Can the triangle-cast traffic model lend insights into the K-user Gaussian interference channel, just like the Z-channel did for the 2-user interference channel?

We believe that the delay constraint opens up fundamental questions about the structure of optimal schemes and reveals certain shortcomings in schemes not accounting for this important practical detail. Thus several questions now reinvent themselves in the presence of delay constraints.

- *Robust Coding* : Suppose we want our communication scheme to be robust to link failures. Then, random coding [3] or algebraic network coding [2] can be shown to be optimal in this setting. Is it possible to obtain robust communication schemes in the presence of delay constraints?
- *Multicasting* : How do we extend the basic multicasting results to traffic with delay constraints?
- For *multiple-unicast traffic* in undirected graphs, routing is within an $O(\log k)$ factor of the capacity region [12]. In the presence of delay constraint D on each unicast flow, does an $O(\log k \log D)$ factor gap between flow and capacity hold?

Acknowledgements

Sudeep Kamath would like to acknowledge support from the Center for Science of Information (CSoI), an NSF Science and Technology Center, under grant agreement CCF-0939370.

References

[1] R. Ahlswede, N. Cai, S.-Y.R. Li, and R.W. Yeung, "Network Information Flow", *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, July 2000.

[2] R. Koetter and M. Médard, "An algebraic approach to network coding", *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, October 2003.

[3] T. Ho, M. Médard, R. Koetter, D. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast", *Information Theory, IEEE Transactions on*, vol. 52, no. 10, pp. 4413–4430, 2006.

[4] A. El Gamal, "On information flow in relay networks", in *NTC'81; National Telecommunications Conference, Volume 2*, 1981, vol. 2, p. 4.

[5] C.-C. Wang and M. Chen, "Sending Perishable Information: Coding Improves Delay-Constrained Throughput Even for Single Unicast", in *Proc. of International Symposium on Information Theory*, Honolulu, Hawaii, USA, July 2014.

[6] S. Kamath, D.N.C. Tse, and V. Anantharam, "Generalized Network Sharing Outer Bound and the Two-Unicast Problem", in *Proc. of International Symposium on Network Coding*, Beijing, China, July 2011.

[7] J. Chuzhoy and S. Khanna, "Polynomial flow-cut gaps and hardness of directed cut problems", *Journal of the ACM (JACM)*, vol. 56, no. 2, pp. 6, 2009.

[8] G. Baier, T. Erlebach, A. Hall, E. Köhler, H. Schilling, and M. Skutella, "Length-bounded cuts and flows", in *Automata, languages and programming*, pp. 679–690. Springer, 2006.

[9] N. Garg, V. Vazirani, and M. Yannakakis, "Approximate max-flow min-(multi) cut theorems and their applications", *SIAM Journal on Computing*, vol. 25, no. 2, pp. 235–251, 1996.

[10] K. Shanmugam, M. Asteris, and A. Dimakis, "Improved Bounds for Multiple Unicast Network Information Flows", https://sites.google.com/a/utexas.edu/karthiksh/, 2014.

[11] J. Naor and L. Zosin, "A 2-approximation algorithm for the directed multiway cut problem", *SIAM Journal on Computing*, vol. 31, no. 2, pp. 477–482, 2001.

[12] T. Leighton and S. Rao, "Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms", *Journal of the ACM (JACM)*, vol. 46, no. 6, pp. 787–832, 1999.

Appendix

A. Proof of Theorem 4

Proof. Consider the triangle-cast problem on an arbitrary directed graph with k sources $s_1, s_2, \ldots, s_k$ and k destinations $d_1, d_2, \ldots, d_k$, where source s_i has a message for each destination d_j so that $i \geq j$. Let c_e denote the capacity of edge e. Let $\mathcal{P}_{i,j}$ be the set of all paths from s_i to d_j. $\mathrm{Cut}_k^{\triangle}$ can be computed by the following integer program:

$$\begin{gathered}\text{Minimize } \sum_e c_e x_e \\ \text{subject to } \sum_{e \in p} x_e \geq 1, \quad \forall\, p \in \mathcal{P}_{i,j}, 1 \leq j \leq i \leq k, \\ x_e \in \{0,1\}.\end{gathered}$$

Let V^* be the optimal value of the corresponding linear program, where the integrality constraint for x_e is relaxed to $0 \leq x_e \leq 1$. It is a well known fact (and readily verified) that the dual of this linear program is the flow linear program, and by linear programming duality, V^* is the max-flow $F_k^{\triangle}$ which can be computed in polynomial time. Now, the key question is: how much is the integrality gap? We will show that the gap is at most $4\log(k+1)$ which will complete the proof.

Proof Sketch: We will start with an optimal solution x to the linear program above. If the optimal solution itself is integral, then clearly $V^* = \mathrm{Cut}_k^{\triangle}$. However, this need not be the case in general, and we need to "round" the x obtained from the linear program in order to get an integral solution. However, the integral x (after rounding) may give rise to a higher objective function than the linear program. If we can bound the increase in the objective function, then we get a handle on the integrality gap. First we will use x_e to define a distance $d(u,v)$ on the set of all nodes, as the minimum over all paths from u to v in the graph using lengths x_e on edge e.

We will now give an overview of our *randomized* rounding scheme. This scheme is an adaptation of the scheme proposed for rounding multi-cuts in undirected graphs [9]. Let us start with the source s_i. Placing this source at the origin, embed all the nodes on to the real line, with the node v placed at distance $d(s_i,v)$ and choose a random cut at a distance r chosen randomly between 0 to $\frac{1}{2}$, that is, the cut separates the nodes $\{v : d(s_i,v) \leq r\}$ from $\{v : d(s_i,v) > r\}$. We do this successively for the various sources $s_k, ..., s_1$, and define the union of all these edges as the cut. It can be shown that this cut is a legitimate edge-cut (because the distance between any s_i to d_j for $j \leq i$ is 1 which is greater than 1/2). The expected objective function obtained from this randomized cut can be bounded by a logarithmic factor in the number of source-destination pairs times the objective function for the linear program.

Detailed Algorithm and Analysis:

Define $V_x(s_i,r)$ for an optimal solution x as follows:

$$\begin{aligned} V_x(s_i,r) := \frac{V^*}{k} + &\sum_{\substack{e=(u,v)\\ u,v \in B_x(s_i,r)}} c_e x_e \\ &+ \sum_{\substack{e=(u,v)\\ u \in B_x(s_i,r)\\ v \notin B_x(s_i,r)}} c_e (r - d_x(s_i,u)), \end{aligned} \tag{6}$$

where $B_x(s_i,r) = \{u \in V : d_x(s_i,u) \leq r\}$, and $d_x(\cdot,\cdot)$ is the minimum distance defined according to lengths x above.

Let $\delta(B_x(s_i,r))$ denote the vertex bipartition cut $(B_x(s_i,r), B_x(s_i,r)^c)$ and let $c(\delta(B_x(s_i,r)))$ denote the sum of edge-capacities of all the edges crossing from $B_x(s_i,r)$ to $B_x(s_i,r)^c$.

Lemma 2. *(Region Growing Lemma) Given any feasible solution x. For any $i, 1 \leq i \leq k$, we can find a radius $r \leq \frac{1}{2}$ so that*

$$c(\delta(B_x(s_i,r))) \leq [2\log(k+1)] V_x(s_i,r). \tag{7}$$

Proof of Lemma 2. Note that if $V_x(s_i,r)$ is differentiable at any r, then

$$\frac{d}{dr} V_x(s_i,r) = c(\delta(B_x(s_i,r))). \tag{8}$$

Sort and label the vertices in $B_x(s_i,\frac{1}{2})$ according to their distance from s_i. Let $r_j = d_x(s_i,v_j)$ so that

$$0 = r_0 \leq r_1 \leq r_2 \leq \ldots \leq r_{l-1} \leq \frac{1}{2} =: r_l, \tag{9}$$

with vertices $s_i = v_0, v_1, v_2, \ldots, v_{l-1}$.

Let r_j^- be a value infinitesimally smaller than r_j.

If r is drawn uniformly over $[0,\frac{1}{2}]$, the expected value of $\frac{c(\delta(B_x(s_i,r)))}{V_x(s_i,r)}$ is

$$\frac{1}{1/2}\sum_{j=0}^{l-1}\int_{r_j}^{r_{j+1}^-} \frac{c(\delta(B_x(s_i,r)))}{V_x(s_i,r)}\,dr \tag{10}$$

$$=2\sum_{j=0}^{l-1}\int_{r_j}^{r_{j+1}^-} \frac{1}{V_x(s_i,r)}\frac{d}{dr}V_x(s_i,r)\,dr \tag{11}$$

$$=2\sum_{j=1}^{l-1} \log V_x(s_i,r)\big|_{r_j}^{r_{j+1}^-} \tag{12}$$

$$=2\sum_{j=1}^{l-1} \log V_x(s_i,r_{j+1}^-) - \log V_x(s_i,r_j) \tag{13}$$

$$\leq 2\sum_{j=1}^{l-1} \log V_x(s_i,r_{j+1}) - \log V_x(s_i,r_j) \tag{14}$$

$$=2\log\frac{V_x(s_i,r_l)}{V_x(s_i,r_0)} \tag{15}$$

$$=2\log\frac{V_x(s_i,\frac{1}{2})}{V_x(s_i,0)}. \tag{16}$$

Now, $V_x(s_i, \frac{1}{2}) \leq \frac{V^*}{k} + V^*$ and $V_x(s_i, 0) \geq \frac{V^*}{k}$. Hence, the expected value is at most $2\log(k+1)$. Therefore, there exists some $r \leq \frac{1}{2}$ so that

$$c(\delta(B_x(s_i, r))) \leq [2\log(k+1)]V_x(s_i, r). \tag{17}$$

□

Now, we will find a cut of value at most $[4\log(k+1)]V^*$ thus showing that $\mathrm{Cut}_k^{\triangle} \leq [4\log(k+1)]F_k^{\triangle}$.

Perform the following sequence of operations:

- Initialize $H = \emptyset$ and the graph to be the given graph.
- For $i = k, k-1, \ldots, 2, 1$,
  - If there is a path from s_i to any of $d_i, d_{i-1}, \ldots, d_1$ in the present graph,
    - $*$ choose $r \leq \frac{1}{2}$ according to Lemma 2.
    - $*$ $H \leftarrow H \cup \delta(B_x(s_i, r))$,
    - $*$ remove all vertices in $B_x(s_i, r)$ and all edges incident to vertices in $B_x(s_i, r)$ from the graph.
- Return H.

For any (i, j) with $i \geq j$, the nodes s_i and d_j are never simultaneously removed by a ball $B_x(s_l, r)$ for some $l \geq i$, since $B_x(s_l, r)$ does not contain d_j for any j satisfying $l \geq i \geq j$. Thus, in the graph $G \setminus H$, there exist no paths from s_i to d_j whenever $i \geq j$.

Let V_i be the sum capacity of all edges that have one or both endpoints in $B_x(s_i, r)$. Then, by definition, $V_x(s_i, r) \leq \frac{V^*}{k} + V_i$.

$$\sum_{e \in H} c_e \leq 2\log(k+1) \sum_{i=1}^{k} \left(\frac{V^*}{k} + V_i \right) \tag{18}$$

$$\leq 4\log(k+1)V^* \quad \text{since } \sum_{i=1}^{k} V_i \leq V^*. \tag{19}$$

This proves $F_k^{\triangle} \leq \mathrm{Cut}_k^{\triangle} \leq [4\log(k+1)]F_k^{\triangle}$ and we are done.

□