Optimal Output Sensitive Fault Tolerant Cuts

Niranka Banerjee
The Institute of Mathematical Sciences, HBNI, Chennai, India
nirankab@imsc.res.in

Venkatesh Raman
The Institute of Mathematical Sciences, HBNI, Chennai, India
vraman@imsc.res.in

Saket Saurabh
The Institute of Mathematical Sciences, HBNI, Chennai, India
University of Bergen, Norway
saket@imsc.res.in

Abstract

In this paper we consider two classic cut-problems, Global Min-Cut and Min k-Cut, via the lens of fault tolerant network design. In particular, given a graph G on n vertices, and a positive integer f, our objective is to compute an upper bound on the size of the sparsest subgraph H of G that preserves edge connectivity of G (denoted by $\lambda(G)$) in the case of Global Min-Cut, and $\lambda(G, k)$ (denotes the minimum number of edges whose removal would partition the graph into at least k connected components) in the case of Min k-Cut, upon failure of any f edges of G. The subgraph H corresponding to Global Min-Cut and Min k-Cut is called f-FTCS and f-FT-k-CS, respectively. We obtain the following results about the sizes of f-FTCS and f-FT-k-CS.

- There exists an f-FTCS with $(n-1)(f + \lambda(G))$ edges. We complement this upper bound with a matching lower bound, by constructing an infinite family of graphs where any f-FTCS must have at least $\frac{(n-\lambda(G)-1)(\lambda(G)+f-1)}{2} + (n - \lambda(G) - 1) + \frac{\lambda(G)(\lambda(G)+1)}{2}$ edges.
- There exists an f-FT-k-CS with $\min\{(2f + \lambda(G, k) - (k-1))(n-1), (f + \lambda(G, k))(n-k) + \ell\}$ edges. We complement this upper bound with a lower bound, by constructing an infinite family of graphs where any f-FT-k-CS must have at least $\frac{(n-\lambda(G,k)-1)(\lambda(G,k)+f-k+1)}{2}) + n - \lambda(G, k) + k - 3 + \frac{(\lambda(G,k)-k+3)(\lambda(G,k)-k+2)}{2}$ edges.

Our upper bounds exploit the structural properties of k-connectivity certificates. On the other hand, for our lower bounds we construct an infinite family of graphs, such that for any graph in the family any f-FTCS (or f-FT-k-CS) must contain all its edges. We also add that our upper bounds are constructive. That is, there exist polynomial time algorithms that construct H with the aforementioned number of edges.

2012 ACM Subject Classification Theory of computation → Data structures design and analysis; Theory of computation → Sparsification and spanners

Keywords and phrases Fault tolerant, minimum cuts, upper bound, lower bound

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2020.10

Funding *Saket Saurabh*: Received funding from European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant no. 819416), and Swarnajayanti Fellowship grant DST/SJF/MSA-01/2017-18.

1 Introduction

There is a common proverb in English – *m it is better to be safe than sorry!* Probably, it has never been more true than the Covid-19-times we are living in. Closed in our homes, computers are probably our only way of communicating with the world. Our machines are part of a larger network – it is is just a node in the network. Thus, to get past this moment in time we need our networks to be more reliable, than ever before. Unfortunately, most of the real life networks are prone to failures. A failure of a link (or a small number of links) in the network may lead to a breakdown in communication. This motivates us to build networks that are resilient to failures, leading to the field of *fault tolerant network design.*

Networks are best modelled as graphs. For example, we could imagine we have a communication network, where the nodes (or vertices) are computers, routers, or cell-towers and there is an edge between them if they can communicate. One could also imagine a transportation network, where the edges correspond to segment of a road and the junctions between the roads are vertices. Once we have abstracted these networks as graphs, there are a number of properties we could try to ask about graphs that are meaningful for the particular network they represent. As stated earlier, real life networks are prone to failures. That is, edges (or vertices) may change their status from active to failed, and vice versa. These failures may occur anytime; however it is expected that they are small in numbers. Further, we can assume that failures are not permanent as they are repaired simultaneously. The fact that we only have a small number of failures is captured by associating an integer–a fault parameter f with the network. That is, we assume that at any point of time we only have at most f-edges (or vertices) that are failed. Indeed, f is much smaller than the number of vertices in the graph. This motivates the research on designing fault tolerant structures for various graph problems in terms of fault parameter f and the input size n.

We now formally define the model of *fault tolerant network design*, with respect to a property Π, we would be interested in. A property of graphs is a function σ that assigns to each graph a value in $\{\mathsf{true}, \mathsf{false}\}$. Given a graph G, a fault parameter f, we want to find a subgraph H of G, such that for any set $F \subseteq E(G)(V(G))$ of size f, we have the following: $\sigma(G - F)$ is true if and only if $\sigma(H - F)$ is true. In general, the solution of a fault tolerant network design is measured by the size of the subgraph H. That is, our objective is to find H with as few edges as possible. Fault tolerant subgraphs have been developed for various problems like reachability [3, 4, 8], shortest path [6, 20, 37–39] and spanners [5, 7, 9, 12, 36]. A fault tolerant subgraph for single source reachability in directed graphs was shown by Baswana et al. [4] to contain $\Theta(2^f n)$ edges. Given a graph G, a source s, and an integer f, a subgraph H is an (α, β)-single source fault tolerant subgraph, if for every vertex $v \in V(G)$, for every $F \subseteq E(G)$ of size at most f, $\mathrm{dist}(s, v, H - F) \leq \alpha{\cdot}\mathrm{dist}(s, v, G - F) + \beta$. Parter and Peleg [39] gave an $(3(f + 1), (f + 1) \log n)$-single source fault tolerant subgraph with with $\mathcal{O}(fn)$ edges. For spanners with a stretch k, Dinitz et al. [12] gave an f-fault tolerant k-spanner with $\tilde{\mathcal{O}}(f^2 n^{1+\frac{2}{k+1}})$ edges. Recently, Chakraborty and Choudhary [8] showed an $\mathcal{O}(n + \min |P|\sqrt{n}, n\sqrt{|P|})$ bound on a subgraph, that is an 1-fault tolerant reachability preserver for a given vertex-pair set $P \subseteq V(G) \times V(G)$.

Our main objective of this article is to extend this study to two classic cut-problems, Global Min-Cut and Min k-Cut. Arguably, Global Min-Cut and Min k-Cut are one of the two most well-studied problems in the field of graph algorithms. In the Min k-Cut problem, input is an undirected graph G and an integer k, and the task is to partition the vertex set into k non-empty sets, say $\tilde{P}$, such that the total number of the edges with endpoints in different parts is minimized. We call such a partition as *min k-cut*, or simply a *k-cut.* For $k = 2$, rather that saying *2-cut*, we say *min-cut.* Indeed, for $k = 2$, this is the classic Global Min-Cut problem, which can be solved in polynomial time. In fact, for every fixed k, the problem is known to be polynomial time solvable [18]. However, when k is part of the input, the problem is NP-complete [18]. Both these problems have been extensively studied in the last 30 years, and the running time of algorithms for these two problems have been improved over the years [10, 15, 19, 22, 24–28, 30, 32, 35, 40, 41]. In particular, after a series of improvement, the fastest known algorithm for Global Min-Cut in unweighted graphs is given by Ghaffari et al. [17] that runs in time $\mathcal{O}(m \log n)$. On the other hand, for edge-weighted graphs the fastest known algorithm for Global Min-Cut is independently given by Gawrychowski et al [16] and Mukhopadhyay and Nanongkai [34] and (almost) runs in time $\mathcal{O}(m \log^2 n)$. Both of these algorithms are randomized. The best known deterministic algorithm for the problem on unweighted graph is given by Henzinger et al. [23] and runs in time $\mathcal{O}(m \log^2 n (\log \log n)^2)$.

The history of Min k-Cut problem is also extremely rich. The direction of polynomial time approximation algorithms is essentially settled, with factor $2(1 - \frac{1}{k})$ approximation algorithms and matching lower bounds. Recently, Gupta et al. [19] showed that for every fixed $k \geq 2$, the Karger-Stein algorithm [29] outputs any fixed k-cut with probability at least $\widehat{\mathcal{O}}(n^{-k})$, where $\widehat{\mathcal{O}}(\cdot)$ hides a $2^{\mathcal{O}(\ln \ln n)^2}$ factor. This immediately gives an extremal bound of $\widehat{\mathcal{O}}(n^k)$, on the number of minimum k-cuts in an n-vertex graph and an algorithm for Min k-Cut in similar running time. Both the extremal bound and the running time of the algorithm are essentially tight (under reasonable assumptions). Indeed the extremal bound matches known lower bounds up to $\widehat{\mathcal{O}}(1)$ factors, while any further improvement to the exact algorithm would imply an improved algorithm for Max-Weight k-Clique [1,2], which has been conjectured not to exist. One can also obtain $f(k)n^{o(k)}$ lower bound on the running time [11,13] under the Exponential Time Hypothesis (ETH). In the world of FPT-approximation, Min k-Cut is known to admit $(1+\epsilon)$ approximation algorithm running in time $(\frac{k}{\epsilon})^{\mathcal{O}(k)} n^{\mathcal{O}(1)}$ [31].

1.1 Our Results and Methods

In this paper we initiate a new research direction to the studies of Global Min-Cut and Min k-Cut. In particular we do the following.

> We focus on Global Min-Cut and Min k-Cut, via the lens of fault tolerant network design, and construct asymptotically optimal fault tolerant subgraphs for these two problems.

Given a graph G, let $\lambda(G)$ and $\lambda(G,k)$ denote the size of min-cut and k-cut of G, respectively. We formally define the objects we consider in the paper.

▶ **Definition 1.1** (f-FTCS (f-FT-k-CS)). *An f-FTCS (f-FT-k-CS) is a subgraph H of G such that for any set of edges $F \subseteq E(G)$ of cardinality at most f, $\lambda(G - F) = \lambda(H - F)$ ($\lambda(G - F, k) = \lambda(H - F, k)$). For a graph G, we use $\Psi(G,k)$ to denote the minimum number of edges in a f-FT-k-CS of G. That is,*

$$\Psi(G,k) = \min_{H \text{ is an } f\text{-FT-}k\text{-CS of } G} |E(H)|$$

When $k = 2$, this denotes the minimum number of edges in a f-FTCS of G. In this case we simply use $\Psi(G)$, rather than $\Psi(G,2)$.

Let $\mathcal{F}$ be a family of graphs, then for all $n \in \mathbb{N}$, we define the following:

$$\begin{aligned} \textsc{Ftcs}(\mathcal{F}, n, f) &= \max_{G \in \mathcal{F}, |V(G)|=n} \Psi(G) \\ \textsc{Ft-}k\textsc{-cs}(\mathcal{F}, n, f) &= \max_{G \in \mathcal{F}, |V(G)|=n} \Psi(G,k) \end{aligned}$$

When $\mathcal{F}$ is the family of all graphs, then we simply use Ftcs(n,f) and Ft-k-cs(n,f). Our goal is to give asymptotic upper bounds on Ftcs(n,f) and Ft-k-cs(n,f). Since any graph has at most $\binom{n}{2}$ edges, we have that $\textsc{Ftcs}(n, f)$ (or $\textsc{Ft-}k\textsc{-cs}(n, f)$) is at most $\mathcal{O}(n^2)$. Let G be a clique on n vertices. First, note that $\lambda(G) = n - 1$. Next observe that any f-FTCS, H of G, even for $f = 1$, must contain *all the edges of the clique*. Indeed, if an edge $(u, v) \in E(G)$ is not present in H, then the adversary may delete an edge adjacent to u or v in the clique, that is not (u, v). In this case, $\lambda(G - F) = n - 2$, whereas $\lambda(H - F) \leq n - 3$. This simple construction shows that $\textsc{Ftcs}(n, 1)$ is at least $\Omega(n^2)$. This bound tells us that for these problems we can not improve upon the trivial upper bounds.

Our example with family of cliques seems to suggest that we have reached the end of the road. However, on the second look we observe that for a clique even $\lambda(G) = \Omega(n)$. Thus, we can also express our lower bound as $\lambda(G) \cdot n$. This motivates us to look for a *fine-grained* definition of $\textsc{Ftcs}(n, f)$ and $\textsc{Ft-}k\textsc{-cs}(n, f)$, that not only takes into account n and f, but also some parameter that captures the edge-connectivity (or the value of k-cut) of the input graph. In particular, we can come up with the following new definitions. Let $\mathcal{F}$ be a family of graphs, then for all $n, \ell \in \mathbb{N}$, we define the following:

$$\begin{aligned} \textsc{Ftcs}(\mathcal{F}, n, \ell, f) &= \max_{G \in \mathcal{F}, |V(G)| = n, \lambda(G) = \ell} \Psi(G) \\ \textsc{Ft-}k\textsc{-cs}(\mathcal{F}, n, \ell, f) &= \max_{G \in \mathcal{F}, |V(G)| = n, \lambda(G,k) = \ell} \Psi(G, k) \end{aligned}$$

With respect to our new definition, when $\mathcal{F}$ is a family of cliques, we have that $\textsc{Ftcs}(\mathcal{F}, n, \ell, 1)$ is at most $\mathcal{O}(\ell n)$. Thus, a natural question arises: Can we derive similar upper bound even when $\mathcal{F}$ denotes the family of all graphs? Indeed, we provide a matching upper and lower bound on these quantities in this paper. As before, when $\mathcal{F}$ is the family of all graphs. Then, we simply use $\textsc{Ftcs}(n, \ell, f)$ and $\textsc{Ft-}k\textsc{-cs}(n, \ell, f)$. Our first result is the following.

▶ **Theorem 1.2.** *Let n, ℓ and f be three positive integers. Then, $\textsc{Ftcs}(n, \ell, f)$ is upper bounded by $(f + \ell)(n - 1)$.*

The proof of Theorem 1.2 is inspired from the concept of *k-connectivity certificates* used in the literature [14, 35]. For a k-edge connected graph $G = (V, E)$, a subset of edges $E' \subseteq E$ is called a k-connectivity certificate of the graph G, if the subgraph $G' = (V, E')$ is k-edge connected. For a k-edge connected graph on n vertices, there always exists a $k-$connectivity certificate with at most $k(n - 1)$ edges [14]. For our proof, we modify a known construction of a k-connectivity certificate to also handle edge failures.

Our second result complements the above upper bound, by showing that this bound is tight upto constant factors. Specifically, we show the following.

▶ **Theorem 1.3.** *There exists an infinite family of triplets (n, ℓ, f) such that*

$$\textsc{Ftcs}(n, \ell, f) \geq \frac{(n - \ell - 1)(\ell + f - 1)}{2} + (n - \ell - 1) + \frac{\ell(\ell + 1)}{2}.$$

To prove Theorem 1.3, we construct an infinite family of graphs ($\mathcal{G}$), such that for any $G \in \mathcal{G}$ we have that any f-FTCS of G must contain all its edges. In particular, for any positive integers n, ℓ, f, such that $\frac{n-\ell-1}{\ell+f}$ is an *integer*, we construct a graph G on n vertices and $(n - \ell - 1)\frac{(\ell+f-1)}{2} + (n - \ell - 1) + \frac{\ell(\ell+1)}{2}$ edges with $\lambda(G) = \ell$ (note that $\ell \leq n - 1$) such that any f-FTCS of G must contain all the edges of G. The construction of the family $\mathcal{G}$, and the analysis that for any graph $G \in \mathcal{G}$, any f-FTCS of G must contain all its edges are quite technical.

Next we generalize our results on Global Min-Cut to Min k-Cut and give the following two results about Ft-k-cs(n, ℓ, f).

▶ **Theorem 1.4.** *Let n, ℓ and f be three positive integers. Then,* $\textsc{Ft-}k\textsc{-cs}(n,\ell,f)$ *is upper bounded by* $\min\{(2f+\ell-(k-1))(n-1),(f+\ell)(n-k)+\ell\}$.

Proof of Theorem 1.4 is quite involved and requires understanding the intricate relationship between edge-connectivity certificates and the Min k-Cut problem. This is one of the main technical results. In our final result, we complement Theorem 1.4 with a tight lower bound.

▶ **Theorem 1.5.** *There exists an infinite family of triplets* (n,ℓ,f) *such that*

$$\textsc{Ft-}k\textsc{-cs}(n,\ell,f) \geq \frac{(n-\ell-1)(\ell+f-k+1)}{2}) + n - \ell + k - 3 + \frac{(\ell-k+3)(\ell-k+2)}{2}.$$

While the construction is somewhat similar in spirit to the construction of the lower bound for the construction of the family of graphs for Global Min-Cut, the proof of correctness is even more involved.

Tightness of our Upper and Lower Bounds. Notwithstanding the fact that the leading terms in our upper and lower bounds appear close, there are some negative quantities in the leading terms, and in some ranges, the other terms in the bounds dominate. Still, our bounds for Global Min-Cut are asymptotically optimal. For example in the lower bound for Global Min-Cut (Theorem 1.3), when $n-\ell$ becomes $o(n)$, ℓ is $\Omega(n)$ and in this case the $\frac{\ell(\ell+1)}{2}$ bound dominates and we get a lower bound of $\Omega(n^2)$ which is asymptotially optimal given our upper bound and the range of ℓ. When ℓ is $o(n)$, our lower bound is $\Omega((f+\ell)n)$ which matches asymptotically with the upper bound.

For Min k-Cut however, there are some gaps. For example, if $\ell = n-1$ and $k = n-\log n$, the upper bound is $\mathcal{O}((f+n)\log n)$ but the lower bound is $\Omega(n)$. Such a gap exists in some ranges of f and k when $n-\ell$ and $\ell-k$ are both $o(n)$. However, when $n-\ell$ or $\ell-k$ is $\Theta(n)$, our upper and lower bounds are a constant factor away from each other.

Algorithmic Considerations. The proof of Theorem 1.2 is constructive. That is, given a graph G and an integer f, in polynomial time we can construct an f-FTCS of G with at most $(f+\lambda(G))(n-1)$ edges. For this algorithm we just need the value of $\lambda(G)$), which can be computed in $\mathcal{O}(m\log^2 n(\log\log n)^2)$ time [23]. However, the proof of Theorem 1.4 is "almost" constructive. That is, the proof can be made constructive, if for a graph G we can compute the value of $\lambda(G,k)$ in polynomial time. Indeed, for a constant value of k, we could use the polynomial time algorithm running in time $n^{\mathcal{O}(k)}$ [10, 19, 41]. However, the running time of this algorithm grows with k, and hence becomes prohibitive quite soon. Thus, as an alternative we could use an upper bound on $\lambda(G,k)$, provided by the known polynomial time factor 2 approximation algorithm [21, 42]. This leads to an upper bound of $\min\{(2f+2\lambda(G,k)-(k-1))(n-1),(f+2\lambda(G,k))(n-k)+2\lambda(G,k)\}$ on the constructed f-FT-k-CS, which is slightly worse than the upper bound provided by Theorem 1.4.

2 Preliminaries

Given an integer q, we use $[q]$ to denote $\{1,\ldots,q\}$. Further, for two integers, $q_1 \leq q_2$, we use $[q_1,q_2]$ to denote $\{q_1,\ldots,q_2\}$. For a graph $G=(V,E)$, we also use $V(G)$ and $E(G)$ to denote the set of vertices and the set of edges of graph G, respectively. A path P in G is a sequence of distinct vertices ($P = v_1v_2\cdots v_q$), such that two consecutive vertices have an edge between them. Let $A_1,\ldots,A_\ell$ be a partition of the vertex set $V(G)$ of a graph G. That is, $\cup_{i=1}^{\ell} A_i = V(G)$ and for all $i \neq j$, $A_i \cap A_j = \emptyset$. We use $E(A_1,\ldots,A_\ell,G)$ to denote the set of edges such that each edge in the set has one endpoint in A_i and the other endpoint in A_j, where $i \neq j$. For a graph G, and a pair of vertices $u, v \in V(G)$, we use $\lambda_G(u, v)$ to denote the minimum number of edges whose removal separates u and v (that is, u and v belong to different connected components). If the graph G is clear from the context, we omit the subscript G from $\lambda_G(u, v)$, and simply write $\lambda(u, v)$. Next, we state the classical Menger's Theorem and a simple lemma which are crucially used in our proofs,

▶ **Lemma 2.1** (Menger's Theorem, [33]). *Let G be an undirected graph and let u and v be two vertices of G. Then the maximum number of pairwise edge-disjoint u-v paths in G is equal to $\lambda(u, v)$.*

▶ **Lemma 2.2.** *Let $G = (V, E)$ be an undirected graph and let H be a subgraph of G. Let $k > 1$ be an integer. Then, $\lambda(H) \leq \lambda(G)$ and $\lambda(H, k) \leq \lambda(G, k)$.*

3 Global Min-Cut

In this section we develop upper and lower bounds on $\textsc{Ftcs}(n, \ell, f)$. In particular we prove Theorems 1.2 and 1.3.

3.1 Upper Bound

Let n, ℓ and f be three positive integers. We need to show that $\textsc{Ftcs}(n, \ell, f)$ is upper bounded by $(f + \ell)(n - 1)$. Towards this we show that given an undirected graph G, and an integer f, we can construct an f-FTCS, H, of G on at most $(f + \lambda(G))(n - 1)$ edges. Indeed, when $\lambda(G) = \ell$, the upper bound follows. Further, we assume G is connected. If G is disconnected then $\lambda(G) = 0$, and it remains so after any edge failure. Thus, in this case we can take H to be an empty graph. Our construction is presented next.

Construction of an f-FTCS of a graph G.

1. Initialize $f + \ell$ empty (no edges) forests $T_1, T_2, \ldots, T_{f+\ell}$ on the same vertex set $V(G)$.
2. for each edge $(u, v) \in E(G)$, do the following.
   - Find the smallest integer $i \in [f + \ell]$, such that u and v are in different connected components of T_i. If no such i exists, then assign i to ∞.
   - If i is not ∞ then add (u, v) to T_i.
3. Output $H = \cup_{a=1}^{f+\ell} T_a$.

We will show that H is an f-FTCS with at most $(f + \ell)(n - 1)$ edges. The bound on the number of edges on H is clear, as H is the union of at most $(f + \ell)$ forests.

▶ **Lemma 3.1.** *The subgraph H has at most $(f + \ell)(n - 1)$ edges.*

Next, we show that H is an f-FTCS. We start with the following observation.

▶ **Lemma 3.2.** ($\star$) [1] *Let $(u, v) \in E(G) \setminus E(H)$. Then there are at least $\ell + f$ edge-disjoint paths between u and v in G and H.*

[1] Results marked with $\star$ are deferred to the full version.

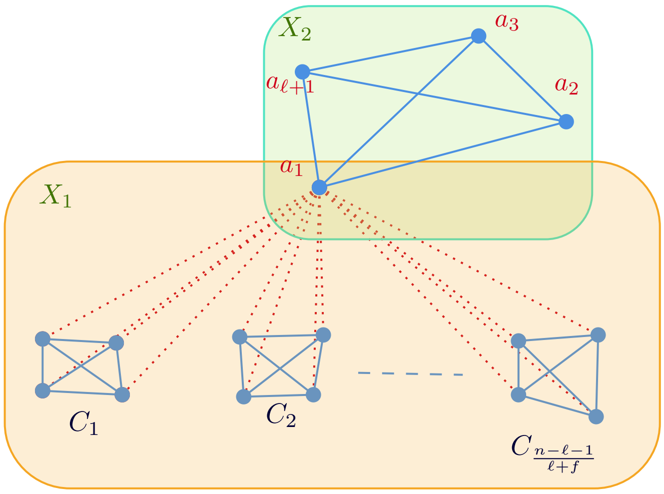

Figure 1 Vertex a_1 has degree $n-\ell-1$ within X_1. Vertices $a_1, a_2, ..., a_{\ell+1}$ in X_2 form an induced $\ell+1$-clique. $C_1, C_2, ..., C_{\frac{n-\ell-1}{\ell+f}}$ represent $\frac{n-\ell-1}{\ell+f}$ cliques each of size $\ell+f$ within X_1.

To prove that H is an f-FTCS of G, we need to show that for any set of edges $F \subseteq E(G)$ of cardinality at most f, $\lambda(H-F) = \lambda(G-F)$. As H is a subgraph of G, we know from Lemma 2.2 that $\lambda(H-F) \leq \lambda(G-F)$. Now we show that $\lambda(H-F) \geq \lambda(G-F)$.

▶ **Lemma 3.3.** *Let G be an undirected graph with $\lambda(G) = \ell$, f be a positive integer, and H be the subgraph constructed above. Then for any set F of at most f edges, $\lambda(H-F) \geq \lambda(G-F)$.*

Proof. Let A, B be a partition of $V(G)$ such that $|E(A,B,H-F)| = \lambda(H-F)$. If $E(A,B,H-F) = E(A,B,G-F)$, then we have that a min-cut in $H-F$ is also a min-cut in $G-F$ of the same size, thereby proving that $\lambda(H-F) \geq \lambda(G-F)$. Suppose not. As H is a subgraph of G, $E(A,B,H-F) \subseteq E(A,B,G-F)$. Suppose $(u,v) \in E(A,B,G-F) \setminus E(A,B,H-F)$. Then $(u,v) \in E(G) \setminus E(H)$. Then from Lemma 3.2, there are $\ell+f$ edge-disjoint paths between u and v in H, and hence there will be at least ℓ edge-disjoint paths between u and v in $H-F$. Hence, $\lambda(H-F) = |E(A,B,H-F)| \geq \ell = \lambda(G) \geq \lambda(G-F)$. ◀

Proof of Theorem 1.2 follows from Lemmas 3.1, 3.2 and 3.3.

3.2 Lower Bound

In this section we show that the upper bound shown on $\textsc{Ftcs}(n, \ell, f)$ in Section 3.1 is indeed asymptotically tight. To prove Theorem 1.3, we construct an infinite family of graphs $\mathcal{G}$, such that for any $G \in \mathcal{G}$ we have that any f-FTCS of G must contain all its edges. In particular, for any positive integers n, ℓ, f, such that $\frac{n-\ell-1}{\ell+f}$ is an *integer*, we construct a graph G on n vertices and $(n-\ell-1)\frac{(\ell+f-1)}{2} + (n-\ell-1) + \frac{\ell(\ell+1)}{2}$ edges with $\lambda(G) = \ell$, such that any f-FTCS of G must contain all the edges of G.

Let n, ℓ, f be three integers such that $\frac{n-\ell-1}{\ell+f} = q$ is an integer. We first describe the construction of a graph G on n vertices. To easily understand our construction, we would suggest to simultaneously refer to the illustration given in Figure 1.

Construction of a graph G. Here, $q = \frac{n-\ell-1}{\ell+f}$.

- The vertex set $V(G)$ is a union of X_1 and X_2, such that $|X_1 \cap X_2| = 1$.
- X_1 has q pairwise vertex disjoint cliques $C_1, \ldots, C_q$. Each clique C_i is on $(\ell + f)$ vertices. X_1 also contains a vertex a_1 as described below.(The edges of the cliques, $C_1, \ldots, C_q$, are denoted by the solid blue edges in Figure 1.)
- The set X_2 consists of $a_1, \ldots, a_{\ell+1}$ vertices that form a clique. These vertices do not belong to the cliques, $C_1, \ldots, C_q$. (The edges of the clique on $a_1, \ldots, a_{\ell+1}$ are represented by blue solid edges in Figure 1.)
- Let $a_1 \in X_2$ be a fixed vertex. Each vertex in a clique $C_i, i \in [q]$, is adjacent to the vertex a_1. There are no edges between a pair of vertices belonging to two distinct cliques, C_i and C_j. The vertex a_1 is the only common vertex between two sets X_1 and X_2. (Edges between a_1 and the vertices in the cliques, $C_1, \ldots, C_q$, are represented by the red dotted edges in Figure 1.)

In the upcoming lemmas we show certain properties of our construction. Here, Lemma 3.5 is used to prove Lemma 3.6.

▶ **Lemma 3.4.** ($\star$) *The number of edges in G is $(n - \ell - 1)\frac{(\ell+f+1)}{2} + \frac{\ell(\ell+1)}{2}$.*

▶ **Lemma 3.5.** ($\star$) *For any two vertices $u_1, u_2 \in X_1$, $\lambda(u_1, u_2) \geq \ell + f$.*

▶ **Lemma 3.6.** ($\star$) *Let G be a graph and $f \geq 1$ be a positive integer. Then $\lambda(G) = \ell$. Further, for any $F \subseteq E(G[X_1])$ of size at most f, we have that $\lambda(G - F) = \ell$.*

We now prove the final property of an f-FTCS.

▶ **Lemma 3.7.** *Any f-FTCS of G must contain all the edges of G.*

Proof. Let H be an f-FTCS of G. We will show that H must contain all the edges of G. Towards this, we partition the edges of G into three parts, and show that all these edges are required in H. In particular, we show that if H does not include an edge of G, then there is a strategy for the adversary to choose a subset F of edges (of size at most f) to delete from G such that $\lambda(G - F)$ and $\lambda(H - F)$ are not the same. Let u_i, $i \in [\ell + f]$, be the set of vertices of a fixed clique C_j.

(i) Let us first show that the edges in the cliques $C_i, i \in [q]$, have to be present in H (the solid blue edges in X_1 in Figure 1). Each u_i has $\ell + f - 1$ edges to vertices in C_j apart from an edge to a_1. Suppose an edge (u_y, u_z), $y, z \in [\ell + f]$, $y \neq z$ is not present in H. Let F consist of any f edges adjacent to u_z in C_j other than (u_y, u_z). We know that f edges exist as $\ell \geq 1$ (by construction G is connected). Now by Lemma 3.6 we know that $\lambda(G - F) = \ell$. But the degree of u_z in $H - F$ becomes $\ell - 1$ as $(u_y, u_z) \notin H$. Thus, $\lambda(H - F) \leq \ell - 1$. This contradicts H being an f-FTCS of G. Therefore, all edges of the cliques C_i must be present in H.

(ii) Next, we show that edges $E(\{a_1\}, C_i, G), i \in [q]$, must be present in H (the red dotted edges in X_1 in Figure 1). Suppose $(u_z, a_1), z \in [\ell + f]$ is not present in H. Let F consist of any f edges adjacent to u_z in C_j other than (u_z, a_1). Now by Lemma 3.6 we know that $\lambda(G - f) = \ell$. However, the degree of u_z in $H - F$ is $\ell - 1$. Thus, $\lambda(H - F) \leq \ell - 1$. This contradicts H being an f-FTCS of G. Therefore, for all $i \in [q]$, all the edges in $E(\{a_1\}, C_i, G)$ must be present in H.

(iii) Lastly, we show that all the edges of the $(\ell + 1)$-clique in X_2 formed by $a_i, i \in [\ell + 1]$ must be present in H (the solid blue edges in X_2 in Figure 1). Suppose an edge

(a_i, a_j) , $i, j \in [\ell+1]$, $i \neq j$ is not present in H. Let F consist of any f edges of the form $(u_i, a_1), i \in [f]$. All these edges exist in $G - F$ as $\ell + f \geq f + 1$ (Since by construction G is connected and $\ell \geq 1$). Observe that $F \subseteq E(G[X_1])$ of size at most f, and hence by Lemma 3.6 we have that $\lambda(G - F) = \ell$. However, $\lambda(H - F) = \ell - 1$, as a_i and a_j have degree $\ell - 1$ inside X_2 in $H - F$. This contradicts H being an f-FTCS of G. Therefore, all the edges of the $\ell + 1$- clique in X_2 must be present in H.

The three cases together show that if H is an f-FTCS of G then all edges of the graph G must be present in H. Thus, the total number of edges present in H is $(n - \ell - 1)\frac{(\ell+f-1)}{2} + (n - \ell - 1) + \frac{\ell(\ell+1)}{2}$. Our proof follows. ◀

Proof of Theorem 1.3 follows from Lemmas 3.4, 3.6 and 3.7.

4 Min k-Cut

In this section we develop upper and lower bounds on FT-k-CS(n, ℓ, f). In particular we prove Theorems 1.4 and 1.5.

4.1 Upper Bound

Let n, ℓ and f be three positive integers. We need to show that FT-k-CS(n, ℓ, f) is upper bounded by $\min\{(2f + \ell - (k-1))(n-1), (f + \ell)(n - k) + \ell\}$. Towards this we show that given an undirected graph G, and an integer $f \geq 1$, we can construct an f-FT-k-CS, H of G on at most $\min\{(2f + \lambda(G,k) - (k-1))(n-1), ((f + \lambda(G,k))(n-k) + \lambda(G,k))\}$ edges. Indeed, when $\lambda(G,k) = \ell$, the upper bound follows. Our construction is presented next. It is similar to the construction of in Section 3.1 except for the choice of t. G is assumed to be connected in the algorithm. The complementary case will be handled later.

K-WAY-FAULT-TOLERANT-CONSTRUCTION

Construction of an f-FT-k-CS of a graph G.

1. Let $t = \min\{2f + \ell + 1 - k, f + \ell\}$.
2. Initialize t empty (no edges) forests $T_1, T_2, \ldots, T_t$ on the same vertex set $V(G)$.
3. for each edge $(u, v) \in E(G)$, do the following.
   - Find the smallest integer $i \in [t]$, such that u and v are in different connected components of T_i. If no such i exists, then assign i to ∞.
   - If i is not ∞ then add (u, v) to T_i.
4. Output $H = \cup_{a=1}^{t} T_a$.

Next, we show that H is an f-FT-k-CS for both the values the variable t can take. We start with the following observation which we use in both the cases.

▶ **Lemma 4.1.** ($\star$) *Let $(u, v) \in E(G) \setminus E(H)$. Then there are at least t edge-disjoint paths between u and v in G and H.*

Note that the $t(n-1)$ upper bound of Section 3.1 for the number of edges in H applies here too with the same proof. However, we show stronger bounds for certain values of t.

4.1.1 Case of $t = f + \ell$

▶ **Lemma 4.2.** *The subgraph H has at most $(f + \ell)(n - k) + \ell$ edges.*

Proof. Let $A_1, \ldots, A_k$ be a partition of $V(G)$ such that $|E(A_1, \ldots, A_k, G)| = \ell$. Let $X = E(A_1, \ldots, A_k, G)$. We will show that every forest $T_i - X$, $i \in [f + \ell]$, has at least k connected components. Note that, once we can show this claim, we can get the upper bound on the number of edges in H. Indeed, each $T_i - X$ has at most $n - k$ edges (since, it has at least k components) and hence $|E(H)| \le \sum_{i=1}^{f+\ell} |E(T_i - X)| + |X| \le (f + \ell)(n - k) + \ell$. Next we prove our claim. Observe that every edge going out of the connected components A_j, $j \in [k]$, is contained inside X. Thus, in particular, every edge going out of the vertices in A_j in T_i is also contained inside X. Hence, the vertices of A_j at least form one connected component in $T_i - X$. This concludes the proof that every forest $T_i - X$, $i \in [f + \ell]$, has at least k connected components. ◀

Next, we show that H is an f-FT-k-CS.

▶ **Lemma 4.3.** *Let G be a graph with $\lambda(G, k) = \ell$, f be a positive integer, and H be the subgraph constructed above. Then, for any set F of at most f edges, $\lambda(H - F, k) \ge \lambda(G - F, k)$.*

Proof. Let $A_1, \ldots, A_k$ be a partition of $V(G)$ such that $|E(A_1, \ldots, A_k, H - F)| = \lambda(H - F, k)$. If $E(A_1, \ldots, A_k, H - F) = E(A_1, \ldots, A_k, G - F)$, then we have that a k-cut in $H - F$ is also a k-cut in $G - F$ of the same size, thereby proving that $\lambda(H - F, k) \ge \lambda(G - F, k)$. Suppose not. As H is a subgraph of G, $E(A_1, \ldots, A_k, H - F) \subseteq E(A_1, \ldots, A_k, G - F)$. Suppose $(u, v) \in E(A_1, \ldots, A_k, G - F) \setminus E(A_1, \ldots, A_k, H - F)$. Then $(u, v) \in E(G) \setminus E(H)$. Then from Lemma 4.1, there are $\ell + f$ edge-disjoint paths between u and v in H, and hence there will be at least ℓ edge-disjoint paths between u and v in $H - F$. Hence, $\lambda(H - F, k) = |E(A_1, \ldots, A_k, H - F)| \ge \ell = \lambda(G, k) \ge \lambda(G - F, k)$. This concludes the proof. ◀

4.1.2 Case of $t = 2f + \ell + 1 - k$

We will show that H is an f-FT-k-CS with at most $(2f + \ell + 1 - k)(n - 1)$ edges.

The bound on the number of edges on H is clear, as H is the union of at most $(2f+\ell+1-k)$ forests.

▶ **Lemma 4.4.** *The subgraph H has at most $(2f + \ell + 1 - k)(n - 1)$ edges.*

We could have obtained a bound similar to Lemma 4.2, but in this case, it does not give us asymptotically better bound than that of $(2f + \ell + 1 - k)(n - 1)$. Next, we show that H is an f-FT-k-CS. We start with the following lemma which is a folklore and we give the proof here for completeness.

▶ **Lemma 4.5.** ($\star$) *Let G be a connected graph and let $u_1, \ldots, u_p \in V(G)$. Further, let $E_{[p]}$ be an inclusion-wise minimal subset of edges, such that $u_1, \ldots, u_p$ get pairwise separated in $G - E_{[p]}$, then $G - E_{[p]}$ has exactly p connected components, one containing each u_i, $i \in [p]$.*

▶ **Lemma 4.6.** *Let G be a connected graph, then for all $p \le k$, we have that $\lambda(G, k) \ge \lambda(G, p) + (k - p)$.*

Proof. Let $A_1, \ldots A_k$ be a k partition of $V(G)$ such that $|E(A_1, \ldots, A_k, G)| = \lambda(G, k)$. Let $u_i \in A_i$, $i \in [p]$, be a vertex. Clearly, $E(A_1, \ldots, A_k, G)$ separates any pair of vertices in $\{u_1, \ldots, u_p\}$, and thus there exists an inclusion-wise minimal subset $E_{[p]} \subseteq E(A_1, \ldots, A_k, G)$, such that any pair of vertices in $\{u_1, \ldots, u_p\}$ gets separated in $G - E_{[p]}$. Now by Lemma 4.5, we have that $G - E_{[p]}$ has exactly p connected components, one containing each u_i, $i \in [p]$. This implies that E_p is a p-cut in G (may not be of the minimum size) and thus, $|E_{[p]}| \geq \lambda(G, p)$.

However, $G - E(A_1, \ldots, A_k, G)$ has k connected components, and deleting an edge can only increase the number of connected components by 1. This implies that $|E(A_1, \ldots, A_k, G) \setminus E_{[p]}| \geq (k - p)$. Putting together this with the fact that $|E_{[p]}| \geq \lambda(G, p)$, we get that

$$\lambda(G, k) \geq |E_{[p]}| + (k - p) \geq \lambda(G, p) + (k - p).$$

This concludes the proof. ◀

To prove that H is an f-FT-k-CS of G, we need to show that for any set of edges $F \subseteq E(G)$ of cardinality at most f, $\lambda(H - F, k) = \lambda(G - F, k)$. As H is a subgraph of G, we know from Lemma 2.2 that $\lambda(H - F, k) \leq \lambda(G - F, k)$. Now we show that $\lambda(H - F, k) \geq \lambda(G - F, k)$. In fact, we will prove something stronger, which we call *robustness*. That is, for all $k^\star \leq k$, we have that $\lambda(H - F, k^\star) \geq \lambda(G - F, k^\star)$.

▶ **Lemma 4.7** (Robustness). *Let G be a connected graph with $\lambda(G, k) = \ell$, f be a positive integer, and H be the subgraph constructed above. Then, for any set F of at most f edges, and for $k^\star \leq k$, $\lambda(H - F, k^\star) \geq \lambda(G - F, k^\star)$.*

Proof. Let $A_1, \ldots A_{k^\star}$ be a partition into $k^\star$ sets of $V(G)$ such that $|E(A_1, \ldots, A_{k^\star}, H - F)| = \lambda(H - F, k^\star)$. If $E(A_1, \ldots, A_{k^\star}, H - F) = E(A_1, \ldots, A_{k^\star}, G - F)$, then we have that a min $k^\star$-cut in $H - F$ is also a min $k^\star$-cut in $G - F$ of the same size, thereby proving that $\lambda(H - F, k^\star) \geq \lambda(G - F, k^\star)$. Suppose not. As H is a subgraph of G, $E(A_1, \ldots, A_{k^\star}, H - F) \subseteq E(A_1, \ldots, A_{k^\star}, G - F)$. Suppose $(u, v) \in E(A_i, A_j, G - F) \setminus E(A_i, A_j, H - F)$, $i, j \in [k^\star]$, and $i \neq j$. Then $(u, v) \in E(G) \setminus E(H)$. From Lemma 4.1, there are $2f + \ell + 1 - k$ edge-disjoint paths between u and v in H, and hence there will be at least $f + \ell + 1 - k$ edge-disjoint paths between u and v in $H - F$.

Observe that, since G is connected, H is also connected by our construction (T_1 is definitely a spanning tree). However, $H - F$ may not be *connected*. On the other hand, since $\lambda_H(u, v) \geq 2f + \ell + 1 - k$, we get that $\lambda_{H-F}(u, v) \geq f + \ell + 1 - k$. Note that since, H is connected, any k-cut has size at least $k - 1$, and thus, $\ell + 1 \geq k$ (recall that, $\lambda(G, k) = \ell$). Since, $\lambda_{H-F}(u, v) \geq f + \ell + 1 - k \geq f \geq 1$, we have that u and v are in the same connected component of $H - F$. Further, they get separated after we delete $E(A_1, \ldots, A_{k^\star}, H - F)$ from $H - F$. This implies that the number of connected components in $H - F$ is at most $k^\star - 1$. Next observe that since H is connected, deleting F from H can only result in at most $|F| + 1$ connected components in $H - F$. Thus, the number of connected components in $H - F$, say d, is upper bounded by the minimum of $\{k^\star - 1, f + 1\}$.

Let the connected component containing u, v in $H - F$ be denoted by C_{uv}. Observe that $E(A_1, \ldots, A_{k^\star}, H - F)$ separates u from v in $H - F - E(A_1, \ldots, A_{k^\star}, H - F)$, and thus there exists an inclusion-wise minimal subset $E_{uv} \subseteq E(A_1, \ldots, A_{k^\star}, H - F)$, such that u and v get separated in $(H - F) - E_{uv}$. Further, note that the minimality of E_{uv} implies that $E_{uv} \subseteq E(C_{uv})$, and it is an an inclusion-wise minimal separator for u and v in C_{uv}. Applying, Lemma 4.5 on C_{uv}, we get that $C_{uv} - E_{uv}$ has exactly two connected components, C_u and C_v, containing u v, respectively. This implies that $|E_{uv}| \geq \lambda_{H-F}(u, v) = \lambda_{C_{uv}}(u, v) \geq f + \ell + 1 - k$. Recall that, $H - F$ has d components, and thus $H - F - E_{uv}$ has $d + 1$ components. However, $H - F - E(A_1, \ldots, A_{k^\star}, H - F)$ has $k^\star$ connected components. This implies that

$|E(A_1, \ldots, A_{k^\star}, H - F) \setminus E_{uv}| \geq (k^\star - d)$. Hence,

$$
\begin{aligned}
\lambda(H - F, k^\star) &= |E(A_1, \ldots, A_{k^\star}, H - F)| \\
&\geq f + \ell + 1 - k + (k^\star - d) \\
&\geq \ell + (f+1) - (f+1) - (k - k^\star) && \text{(Using } d \leq f+1\text{)} \\
&= \ell - (k - k^\star) \\
&= \lambda(G, k) - (k - k^\star) && \text{(since, } \lambda(G,k) = \ell\text{)} \\
&\geq \lambda(G - F, k^\star) + (k - k^\star) - (k - k^\star) && \text{(Lemma 4.6)} \\
&= \lambda(G - F, k^\star).
\end{aligned}
$$

This concludes the proof. ◀

Now we deal with the case when G is not connected.

▶ **Lemma 4.8.** *Let G be a disconnected graph with $d > 1$ connected components with $\lambda(G, k) = \ell$ and let f be a positive integer. Then there exists a subgraph H of G with at most $(n-d)(2f + \ell + 2 - k + d)$ edges such that for any set F of at most f edges, $\lambda(H - F, k) \geq \lambda(G - F, k)$.*

Proof. If $d \geq k$, then we return H as an empty (edgeless) graph on the vertices of G. So let us assume that $d < k$. Suppose, G has connected components $G_1, \ldots, G_d$. We apply K-WAY-FAULT-TOLERANT-CONSTRUCTION with $G_i, i \in [d]$ and $k' = (k - d + 1)$ and get H_i. Let $H = \cup_{i=1}^{d} H_i$. That is, we apply our upper bound construction on *each of the connected components* with k' and get the desired H. Lemma 4.7 implies the following.

▷ Claim 4.9. For all $i \in [d]$, $k^\star \leq k'$, H_i is a f-FT-$k^\star$-CS of G_i.

Next we show that for any set F of at most f edges, $\lambda(H - F, k) \geq \lambda(G - F, k)$. Let $A_1, \ldots, A_k$ be a k partition of $V(G)$ such that $|E(A_1, \ldots, A_k, H - F)| = \lambda(H - F, k)$. Observe that, since G_i is connected we have that H_i is connected. Let $A_1, \ldots, A_k$ be a k partition of $V(G)$ such that $|E(A_1, \ldots, A_k, H - F)| = \lambda(H - F, k)$. Recall, that $d < k$, thus, $\lambda(H - F, k) > 0$. Further, from the minimality of $E(A_1, \ldots, A_k, H - F)$, and the fact that $|E(A_1, \ldots, A_k, H - F)| \geq 1$, we have that $H[A_i]$ is connected, and completely contained inside one of G_j, $j \in [d]$. That is, $E(A_1, \ldots, A_k, H - F)$ further partitions some of the connected components, and each connected component of $H - E(A_1, \ldots, A_k, H - F)$ is a part in the partition $(A_1, \ldots, A_k)$ of $V(G)$.

Let $E_i = E(H_i) \cap E(A_1, \ldots, A_k, H - F)$, $i \in [d]$ and $\ell_i = |E_i|$. Further, let $\mathcal{A}_i$, $i \in [d]$, be the set of parts among $A_1, \ldots, A_k$, which are completely contained inside G_i. Note that for all $i, j \in [d]$, $i \neq j$, $\mathcal{A}_i$ and $\mathcal{A}_j$ are pairwise disjoint, and for all $q \in [k]$, there exists an integer $j \in [d]$, such that $A_q \in \mathcal{A}_j$. We summarize this in the following claim.

▷ Claim 4.10. Let $k_i = |\mathcal{A}_i|$, then $\sum_{i=1}^{d} k_i = k$.

Next we have the following claim.

▷ Claim 4.11. For all $i \in [d]$, $k_i \leq k'$, $\lambda(H_i - F, k_i) = \ell_i$. Further, $\sum_{i=1} \ell_i = \ell$.

Proof. No k_i can be more than k', otherwise we get strictly greater than k components, contradicting that $H - F - E(A_1, \ldots, A_k, H - F)$ has exactly k components. Further, for some i, let $\lambda(H_i, k_i) = \ell_i' < \ell_i$. In this case, we can delete ℓ_i' edges inside G_i, and delete E_j, $j \neq i$, to get k components in $H - F$, by deleting strictly less than ℓ edges. This contradicts the definition of $E(A_1, \ldots, A_k, H - F)$. By definition $\sum_{i=1} \ell_i = \ell$. ◁

$$\begin{aligned}
\lambda(H-F,k) &= \sum_{i=1}^{d} \lambda(H_i - F, k_i) && \text{(Claim 4.11)} \\
&\geq \sum_{i=1}^{d} \lambda(G_i - F, k_i) && \text{(Claim 4.9)} \\
&= \lambda(G-F,k).
\end{aligned}$$

To see the last inequality observe the following. Let W_i be a subset of edges in $E(G_i)$ such that $|W_i| = \lambda(G_i - F, k_i)$. Then, clearly by deleting $W = \cup_{i=1}^{d} W_i$, we get $\sum_{i=1}^{d} k_i = k$ components in $G-F$. Here, we used Claim 4.10 to conclude that $\sum_{i=1}^{d} k_i = k$. This implies that $\sum_{i=1}^{d} \lambda(G_i - F, k_i) = \sum_{i=1}^{d} |W_i|$ is a k-cut of $G-F$ (may not be of the minimum size). Thus, a min k-cut in $G-F$ can only be *smaller*. This concludes the correctness proof.

All that remains to show is the upper bound on the number of edges. Let the number of vertices in each component be $n_i, i \in [d]$. Then, the total number of edges in H is upper bounded as follows.

$$|E(H)| \leq \sum_{i=1}^{d} (n_i - 1)(2f + \ell + 1 - k') = (n-d)(2f + \ell + 1 - k') = (n-d)(2f + \ell + 2 - k + d).$$

This concludes the proof. ◀

Proof of Theorem 1.4 follows from Lemmas 4.2, 4.3, 4.4, 4.7 and 4.8.

4.2 Lower Bound

In this section we show that the upper bound shown on $\textsc{Ft}\text{-}k\text{-}\textsc{cs}(n, \ell, f)$ in Section 4.1 is indeed asymptotically tight. To prove Theorem 1.5, we construct an infinite family of graphs $\mathcal{G}$, such that for any $G \in \mathcal{G}$ we have that any f-FT-k-CS of G must contain all its edges. In particular, for any positive integers n, ℓ, f, such that $\frac{n-\ell-1}{\ell+f-(k-2)}$ is an *integer*, we construct a graph G on n vertices and $\frac{(n-\ell-1)(\ell+f-k+1)}{2}) + n - \ell + k - 3 + \frac{(\ell-k+3)(\ell-k+2)}{2}$ edges with $\lambda(G, k) = \ell$, such that any f-FT-k-CS of G must contain all the edges of G.

The graph G is a modification of the graph used to show the lower bound for global minimum cut in Section 3.2.

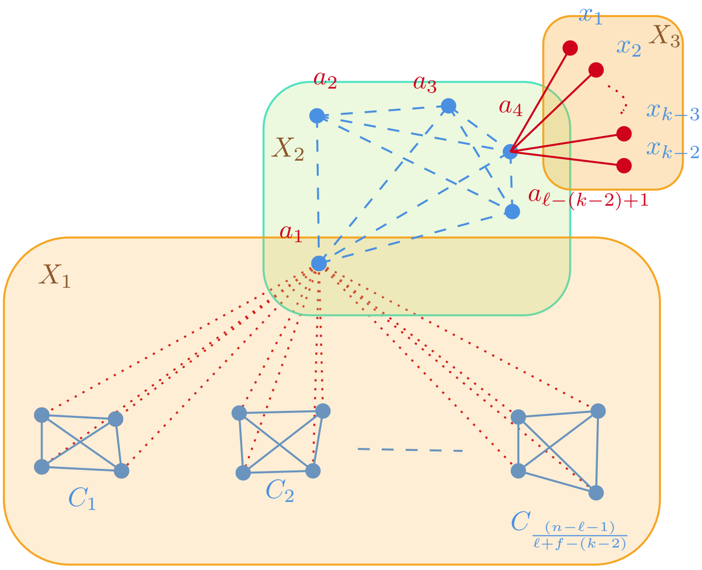

Figure 2 Vertex a_1 has degree $n-\ell-1$ within X_1. In X_2 the vertices $a_1, a_2, ..., a_{\ell-(k-2)+1}$ induce a clique. The vertex a_4 has $k-2$ edges in X_3 each going to a separate vertex x_i. There are no edges between $x_i's$. $C_1, C_2, ..., C_{\frac{(n-\ell-1)}{\ell+f-(k-2)}}$ represent $\frac{(n-\ell-1)}{\ell+f-(k-2)}$ cliques each of size $\ell+f-(k-2)$ within X_1.

Construction of a graph G. Here, $q = \frac{n-\ell-1}{\ell+f-(k-2)}$.

- The vertex set $V(G)$ is a union of X_1, X_2 and X_3, such that $|X_1 \cap X_2| = 1$.
- X_1 has q pairwise vertex disjoint cliques $C_1, \ldots, C_q$. Each clique C_i is on $(\ell+f-(k-2))$ vertices. (The edges of the cliques, $C_1, \ldots, C_q$ are denoted by the solid blue edges in Figure 2.)
- The set X_2 consists of $a_1, \ldots, a_{\ell-(k-2)+1}$ vertices that form a clique. These vertices do not belong to the cliques, $C_1, \ldots, C_q$. (The edges of the clique on $a_1, \ldots, a_{\ell-(k-2)+1}$ are represented by blue solid edges in X_2 in Figure 2.)
- Let $a_1 \in X_2$ be a fixed vertex. All the vertices in a clique $C_i, i \in [q]$, is adjacent to the vertex a_1. There are no edges between a pair of vertices belonging to two distinct cliques, C_i and C_j. The vertex a_1 is the only common vertex between two sets X_1 and X_2. (Edges between a_1 and the vertices in the cliques, $C_1, \ldots, C_q$, are represented by the red dotted edges in Figure 2.)
- X_3 consists of $k-2$ vertices $x_i, i \in [k-2]$. Let $a_4 \in X_2$ be a fixed vertex. Edges in X_3 are of the form $(a_4, x_i), i \in [k-2]$. There are no edges between $x_i's$. (Edges between a_4 and the vertices $x_i \in X_3$, are represented by the red solid edges in Figure 2.)

In the upcoming lemmas we show certain properties of our construction.

▶ **Lemma 4.12.** *The number of edges in G is* $\frac{(n-\ell-1)(\ell+f-k+1)}{2})+n-\ell+k-3+\frac{(\ell-k+3)(\ell-k+2)}{2}$.

Proof. Each clique C_i is of size $(\ell+f-(k-2))$ and contributes $(\ell+f-(k-2))(\frac{\ell+f-k+1}{2})$ edges. There are q cliques and thus the total number of edges contributed by all cliques $C_i, i \in [q]$ is $(n-\ell-1)(\frac{\ell+f-k+1}{2})$. The vertex a_1 is adjacent to all vertices of all $C_i's$. Hence, a_1 has degree $n-\ell-1$ inside X_1. The $(\ell-(k-2)+1)-$clique in X_2 contributes $\frac{(\ell-k+3)(\ell-k+2)}{2}$ edges. The vertex a_4 contributes $k-2$ edges in X_3. Therefore, the total number of edges of G is $\frac{(n-\ell-1)(\ell+f-k+1)}{2}) + n - \ell + k - 3 + \frac{(\ell-k+3)(\ell-k+2)}{2}$. ◀

▶ **Lemma 4.13.** *For any two vertices $u_1, u_2 \in X_1$, $\lambda(u_1, u_2) \geq \ell + f - k + 2$.*

Proof. The pair $\{u_1, u_2\}$ is of one of the three types described below. We prove the claim for each of the three types.

- Both u_1 and u_2 are part of the same clique C_i in X_1. We know that the size of C_i is $\ell + f - k + 2$. Let the other vertices in C_i be $u_j, j \in [3, \ell + f - k + 2]$. Then $u_1 u_j u_2, j \in [3, \ell + f - k + 2], u_1, u_2$ and $u_1 a_1 u_2$ are $\ell + f - k + 2$ edge-disjoint paths between u_1 and u_2. By Theorem 2.1 $\lambda(u_1, u_2) \geq \ell + f - k + 2$.
- Vertices $u_1 \in C_i$ and $u_2 \in C_j$, and $i \neq j$. Let $v_j, j \in [\ell + f - k + 1]$ denote the vertices in C_j other than u_2. Let w_1 be a vertex in C_i other than u_1. Then $u_1 a_1 v_j u_2, j \in [\ell + f - k + 1]$ and $u_1 w_1 a_1 u_2$ are $\ell + f - k + 2$ edge-disjoint paths between u_1 and u_2. By Theorem 2.1 $\lambda(u_1, u_2) \geq \ell + f - k + 2$.
- Let u_1 be a part of clique C_i and $u_2 = a_1$. Let $v_j, j \in [\ell + f - k + 1]$ denote the vertices in C_i other than u_1. Then $u_1 v_j a_1, j \in [\ell + f - k + 1]$ and $u_1 a_1$ are $\ell + f - k + 2$ edge-disjoint paths between u_1 and a_1. By Theorem 2.1 that $\lambda(u_1, u_2) \geq \ell + f - k + 2$.

This concludes the proof. ◀

▶ **Lemma 4.14.** *Let G be a graph and $f \geq 1$ be a positive integer. Then $\lambda(G, k) = \ell$. Further, for any $F \subseteq E(G[X_1])$ of size at most f, we have that $\lambda(G - F, k) = \ell$.*

Proof. Vertices $x_i, i \in [k - 2]$ in X_3 have degree 1 with all of them adjacent to a_4. The edges $E(x, X_1 \cup X_2, G)$ for all $x = x_i, i \in [k - 2]$ partition the graph into $k - 1$ components using $k - 2$ edges. As a minimum of $k - 2$ edges are required to partition a connected graph into $k - 1$ components all these edges will be part of $\lambda(G, k)$. We need one more partition of the graph to get k components.

The cut $E(\{a\}, X_1 \cup X_2 \setminus \{a\}, G)$, where $a = a_i$ for $i \in [\ell - k + 3]$ is of size $\ell - k + 2$. Together, with the edges $E(x, X_1 \cup X_2, G)$ for all $x = x_i, i \in [k - 2]$ we get a $k-$cut of G of size ℓ.

Now we show that any other cut if of size at least ℓ. From Lemma 4.13 we know that for any two vertices $u_1, u_2 \in X_1$, $\lambda(u_1, u_2) \geq \ell + f - k + 2$. This implies that for any 2 partitions A, B of $V(G)$ such that $|X_1 \cap A| \geq 1$ and $|X_1 \cap B| \geq 1$ we have that $|E(A, B, G)| \geq \ell + f - k + 2 \geq \ell - k + 3$. In this case, $\lambda(G - F, k) \geq \ell - k + 3 + (k - 2) = \ell + 1$. Thus, any min-$k$-cut should keep all of X_1 in one side of the partition. It can be easily checked that $|E(X_1 \cup Y, X_2 \setminus Y, G)| \geq \ell - k + 2$ for any $Y \subseteq X_2$, with the minimum being achieved when Y is a singleton set. These edges along with $k - 2$ edges from X_3 shows that $\lambda(G, k) = \ell$. This concludes the first part of the proof.

Let $F \subseteq E(G[X_1])$ of size at most f and $A_i, i \in [k]$ be a partitioning of $V(G)$. We will show that $|E(A_1, ..., A_k, G - F)| \geq \ell$. Indeed, if $|X_1 \cap A_i| \geq 1$ and $|X_1 \cap A_j| \geq 1$ for $i \neq j$, we have that $|E(A_i, A_j, G - F)| \geq \ell - k + 2$ (since, $|E(A_i, A_j, G)| \geq \ell + f - k + 2$). These edges alongwith the $k - 2$ edges $(a_4, x_i), i \in [k - 2]$ give $|E(A_1, ..., A_k, G - F)| \geq \ell$. Thus, let us assume that all of X_1 in one side of the partition. Again in this case, we can easily check that $|E(X_1 \cup Y, X_2 \setminus Y, G - F)| \geq \ell - k + 2$ for any $Y \subseteq X_2$, with the minimum being achieved when Y is a singleton set. Alongwith the edges $(a_4, x_i), i \in [k - 2]$, we have that $|E(A_1, ..., A_k, G - F)| \geq \ell$. Thus, $\lambda(G - F, k) = \ell$. This concludes the proof. ◀

We now prove the final property of an f-FT-k-CS.

▶ **Lemma 4.15.** *Any f-FT-k-CS of G must contain all the edges of G.*

Proof. Let H be an f-FT-k-CS of G. We will show that H must contain all edges of G. Towards this, we partition the edges of G into four parts, and show that all these edges are required in H. In particular, we show that if H does not include an edge of G, then there is a strategy for the adversary to choose a subset F of edges (of size at most f) to delete from G such that $\lambda(G-F,k)$ and $\lambda(H-F,k)$ are not the same. Let u_i, $i \in [\ell+f-k+2)]$, be the set of vertices of a fixed clique C_j.

(i) We first show that the edges in the cliques $C_i, i \in [q]$ in X_1 are present in H(the solid blue edges in X_1 in Figure 2). Each u_i has degree $\ell+f-k+1$ within C_j apart from an edge to a_i. Suppose an edge $(u_y,u_z), y,z \in [\ell+f-k+2], y \neq z$ is not present in H. Let F consist of any f edges adjacent to u_z in C_j other than (u_y,u_z). We know that f edges exist as $\ell \geq k-1$ (by construction G is connected).
Now by Lemma 4.14 we know that $\lambda(G-F,k)=\ell$. But the degree of u_z in $H-F$ becomes $\ell-k+1$ as $(u_y,u_z) \notin H-F$. In $H-F$, we will choose all the remaining adjacent edges of u_z and the $k-2$ edges in X_3 as our cut edges. Thus, $\lambda(H-F,k)=\ell-1$. This contradicts H being an f-FT-k-CS of G. Therefore, all edges of the cliques C_i must be present in H.

(ii) Next, we show that the edges $E(\{a_1\},C_i,G)$ are present in H(the red dotted edges in X_1 in Figure 2). Suppose $(u_z,a_1), z \in [\ell+f-k+2]$ is not present in H. Let F consist of any f edges adjacent to u_z in C_j other than (u_z,a_1). By Lemma 4.14, we know that $\lambda(G-F)=\ell$ but $\lambda(H-F)=\ell-1$. A similar argument to case (i), shows that all such edges $E(\{a_1\},C_i,G)$ must be present in H.

(iii) Next, let us show that the edges in the $\ell-k+3$-clique in X_2 formed by $a_i, i \in [\ell-k+3]$ are present in H(the dashed blue edges in X_2 in Figure 2). Suppose edge $(a_i,a_j), i,j \in [\ell-k+3], i \neq j$ is not present in H. Let F consist of any f edges of the form $(u_i,a_1), i \in [f]$. All these edges exist in $G-F$ as $\ell+f-k+2 \geq f+1$ (Since G is connected and $\ell \geq k-1$).
By Lemma 4.14 we have that $\lambda(G,k)=\ell$. However, as a_i and a_j both have degree $\ell-k+1$ inside X_2 in H so $E(\{a_i\},X_1\cup X_2\setminus\{a_i\},H-F)$ or $E(\{a_j\},X_1\cup X_2\setminus\{a_j\},H-F)$ alongwith $k-2$ edges in X_3 give $\lambda(H-F,k)=\ell-1$. This contradicts H being an f-FT-k-CS of G. Therefore, all edges of the $\ell-k+3$-clique in X_2 must be present in H.

(iv) Lastly, we show that all the $k-2$ edges $E(x,X_1\cup X_2,G)$ for all $x=x_i, i \in [k-2]$ are present in H(the solid red edges in X_3 in Figure 2). Suppose an edge $(a_4,x_z), z \in [k-2]$ is not present in H. Let F consist of any f edges of the form $(u_i,a_1), i \in [f]$.
Again by Lemma 4.14 we have that $\lambda(G-F,k)=\ell$. However, as edge $(a_4,x_z) \notin H$, we have that $\lambda(H-F,k)=\ell-1$. This contradicts H being an f-FT-k-CS of G. Therefore, all $k-2$ edges of X_3 must be present in H.

All the cases together show that all edges of the graph G must be present in H if H is an f-FT-k-CS of G. Thus, the total number of edges present in H is $\frac{(n-\ell-1)(\ell+f-k+1)}{2})+n-\ell+k-3+\frac{(\ell-k+3)(\ell-k+2)}{2}$. Our proof follows. ◀

Proof of Theorem 1.5 follows from Lemmas 4.12, 4.14 and 4.15.

References

1 Amir Abboud, Virginia Vassilevska Williams, and Oren Weimann. Consequences of faster alignment of sequences. In Javier Esparza, Pierre Fraigniaud, Thore Husfeldt, and Elias Koutsoupias, editors, *Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I*, volume 8572 of *Lecture Notes in Computer Science*, pages 39–51. Springer, 2014. doi:10.1007/978-3-662-43948-7_4.

2 Arturs Backurs and Christos Tzamos. Improving viterbi is hard: Better runtimes imply faster clique algorithms. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11*

August 2017, volume 70 of *Proceedings of Machine Learning Research*, pages 311–321. PMLR, 2017. URL: http://proceedings.mlr.press/v70/backurs17a.html.

3 Surender Baswana, Shreejit Ray Chaudhury, Keerti Choudhary, and Shahbaz Khan. Dynamic DFS in undirected graphs: breaking the O(m) barrier. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, pages 730–739, 2016.

4 Surender Baswana, Keerti Choudhary, and Liam Roditty. Fault tolerant reachability for directed graphs. In *Distributed Computing - 29th International Symposium, DISC 2015, Tokyo, Japan, October 7-9, 2015, Proceedings*, pages 528–543, 2015.

5 Davide Bilò, Fabrizio Grandoni, Luciano Gualà, Stefano Leucci, and Guido Proietti. Improved purely additive fault-tolerant spanners. In *Algorithms - ESA 2015 - 23rd Annual European Symposium, Patras, Greece, September 14-16, 2015, Proceedings*, pages 167–178, 2015.

6 Davide Bilò, Luciano Gualà, Stefano Leucci, and Guido Proietti. Multiple-edge-fault-tolerant approximate shortest-path trees. In *33rd Symposium on Theoretical Aspects of Computer Science, STACS 2016, February 17-20, 2016, Orléans, France*, pages 18:1–18:14, 2016.

7 Gilad Braunschvig, Shiri Chechik, David Peleg, and Adam Sealfon. Fault tolerant additive and (μ, α)-spanners. *Theor. Comput. Sci.*, 580:94–100, 2015.

8 Diptarka Chakraborty and Keerti Choudhary. New extremal bounds for reachability and strong-connectivity preservers under failures. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 25:1–25:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ICALP.2020.25.

9 Shiri Chechik, Michael Langberg, David Peleg, and Liam Roditty. Fault tolerant spanners for general graphs. *SIAM J. Comput.*, 39(7):3403–3423, 2010.

10 Chandra Chekuri, Kent Quanrud, and Chao Xu. LP relaxation and tree packing for minimum k-cuts. In *2nd Symposium on Simplicity in Algorithms, SOSA@SODA 2019, January 8-9, 2019 - San Diego, CA, USA*, pages 7:1–7:18, 2019. doi:10.4230/OASIcs.SOSA.2019.7.

11 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.

12 Michael Dinitz and Robert Krauthgamer. Fault-tolerant spanners: better and simpler. In *Proceedings of the 30th Annual ACM Symposium on Principles of Distributed Computing, PODC 2011, San Jose, CA, USA, June 6-8, 2011*, pages 169–178, 2011.

13 Rodney G. Downey, Vladimir Estivill-Castro, Michael R. Fellows, Elena Prieto-Rodriguez, and Frances A. Rosamond. Cutting up is hard to do: the parameterized complexity of k-cut and related problems. *Electron. Notes Theor. Comput. Sci.*, 78:209–222, 2003. doi:10.1016/S1571-0661(04)81014-4.

14 Shimon Even. An algorithm for determining whether the connectivity of a graph is at least k. *SIAM J. Comput.*, 4(3):393–396, 1975. doi:10.1137/0204034.

15 Harold N. Gabow. A matroid approach to finding edge connectivity and packing arborescences. In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing, May 5-8, 1991, New Orleans, Louisiana, USA*, pages 112–122, 1991. doi:10.1145/103418.103436.

16 Pawel Gawrychowski, Shay Mozes, and Oren Weimann. Minimum cut in $O(m \log^2 n)$ time. In Artur Czumaj, Anuj Dawar, and Emanuela Merelli, editors, *47th International Colloquium on Automata, Languages, and Programming, ICALP 2020, July 8-11, 2020, Saarbrücken, Germany (Virtual Conference)*, volume 168 of *LIPIcs*, pages 57:1–57:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ICALP.2020.57.

17 Mohsen Ghaffari, Krzysztof Nowicki, and Mikkel Thorup. Faster algorithms for edge connectivity via random 2-out contractions. In Shuchi Chawla, editor, *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms, SODA 2020, Salt Lake City, UT, USA, January 5-8, 2020*, pages 1260–1279. SIAM, 2020. doi:10.1137/1.9781611975994.77.

18 Olivier Goldschmidt and Dorit S. Hochbaum. A polynomial algorithm for the k-cut problem for fixed k. *Math. Oper. Res.*, 19(1):24–37, 1994. doi:10.1287/moor.19.1.24.

19 Anupam Gupta, Euiwoong Lee, and Jason Li. The karger-stein algorithm is optimal for k-cut. In *Proccedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 473–484, 2020. doi:10.1145/3357713.3384285.

20 Manoj Gupta and Shahbaz Khan. Multiple source dual fault tolerant BFS trees. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017, July 10-14, 2017, Warsaw, Poland*, volume 80 of *LIPIcs*, pages 127:1–127:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPIcs.ICALP.2017.127.

21 Nili Guttmann-Beck and Refael Hassin. Approximation algorithms for minimum K-cut. *Algorithmica*, 27(2):198–207, 2000. doi:10.1007/s004530010013.

22 Monika Henzinger, Satish Rao, and Di Wang. Local flow partitioning for faster edge connectivity. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 1919–1938, 2017. doi:10.1137/1.9781611974782.125.

23 Monika Henzinger, Satish Rao, and Di Wang. Local flow partitioning for faster edge connectivity. *SIAM J. Comput.*, 49(1):1–36, 2020. doi:10.1137/18M1180335.

24 David R. Karger. Global min-cuts in rnc, and other ramifications of a simple min-cut algorithm. In *Proceedings of the Fourth Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms, 25-27 January 1993, Austin, Texas, USA*, pages 21–30, 1993. URL: http://dl.acm.org/citation.cfm?id=313559.313605.

25 David R. Karger. Using randomized sparsification to approximate minimum cuts. In Daniel Dominic Sleator, editor, *Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms. 23-25 January 1994, Arlington, Virginia, USA*, pages 424–432. ACM/SIAM, 1994. URL: http://dl.acm.org/citation.cfm?id=314464.314582.

26 David R. Karger. Using randomized sparsification to approximate minimum cuts. In *Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms. 23-25 January 1994, Arlington, Virginia, USA*, pages 424–432, 1994.

27 David R. Karger. Minimum cuts in near-linear time. *J. ACM*, 47(1):46–76, 2000. doi:10.1145/331605.331608.

28 David R. Karger and Clifford Stein. An o~(n^2) algorithm for minimum cuts. In *Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing, May 16-18, 1993, San Diego, CA, USA*, pages 757–765, 1993. doi:10.1145/167088.167281.

29 David R. Karger and Clifford Stein. A new approach to the minimum cut problem. *J. ACM*, 43(4):601–640, 1996. doi:10.1145/234533.234534.

30 Ken-ichi Kawarabayashi and Mikkel Thorup. Deterministic global minimum cut of a simple graph in near-linear time. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 665–674, 2015. doi:10.1145/2746539.2746588.

31 Daniel Lokshtanov, Saket Saurabh, and Vaishali Surianarayanan. A parameterized approximation scheme for min k-cut. *CoRR, to appear in FOCS 2020*, abs/2005.00134, 2020. arXiv:2005.00134.

32 David W. Matula. A linear time 2+epsilon approximation algorithm for edge connectivity. In *Proceedings of the Fourth Annual ACM/SIGACT-SIAM Symposium on Discrete Algorithms, 25-27 January 1993, Austin, Texas, USA*, pages 500–504, 1993. URL: http://dl.acm.org/citation.cfm?id=313559.313872.

33 Karl Menger. Zur allgemeinen kurventheorie. *Fund. Math. 10:*, pages 96–115, 1927.

34 Sagnik Mukhopadhyay and Danupon Nanongkai. Weighted min-cut: sequential, cut-query, and streaming algorithms. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *Proccedings of the 52nd Annual ACM SIGACT*

Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020, pages 496–509. ACM, 2020. doi:10.1145/3357713.3384334.

35 Hiroshi Nagamochi and Toshihide Ibaraki. A linear-time algorithm for finding a sparse k-connected spanning subgraph of a k-connected graph. *Algorithmica*, 7(5&6):583–596, 1992. doi:10.1007/BF01758778.

36 Merav Parter. Vertex fault tolerant additive spanners. In *Distributed Computing - 28th International Symposium, DISC 2014, Austin, TX, USA, October 12-15, 2014. Proceedings*, pages 167–181, 2014.

37 Merav Parter. Dual failure resilient BFS structure. In *Proceedings of the 2015 ACM Symposium on Principles of Distributed Computing, PODC 2015, Donostia-San Sebastián, Spain, July 21 - 23, 2015*, pages 481–490, 2015.

38 Merav Parter and David Peleg. Sparse fault-tolerant BFS trees. In *Algorithms - ESA 2013 - 21st Annual European Symposium, Sophia Antipolis, France, September 2-4, 2013. Proceedings*, pages 779–790, 2013.

39 Merav Parter and David Peleg. Fault tolerant approximate BFS structures. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 1073–1092, 2014.

40 Mechthild Stoer and Frank Wagner. A simple min-cut algorithm. *J. ACM*, 44(4):585–591, 1997. doi:10.1145/263867.263872.

41 Mikkel Thorup. Minimum k-way cuts via deterministic greedy tree packing. In Cynthia Dwork, editor, *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 159–166. ACM, 2008. doi:10.1145/1374376.1374402.

42 Vijay Vazirani. Approximation algorithms. *Berlin: Springer, ISBN 978-3-540-65367-7*, 2003.