Finding small balanced separators

Uriel Feige
Microsoft Research
One Microsoft Way
Redmond WA 98052
urifeige@microsoft.com

Mohammad Mahdian
Microsoft Research
One Microsoft Way
Redmond WA 98052
mahdian@microsoft.com

ABSTRACT

Let G be an n-vertex graph that has a vertex separator of size k that partitions the graph into connected components of size smaller than αn, for some fixed $2/3 \leq \alpha < 1$. Such a separator is called an α-separator. Finding an α-separator of size at most k is NP-hard. Moreover, under reasonable complexity theoretic assumptions, it is shown that this problem is not polynomially solvable even when $k = O(\log n)$. In this paper, we give a randomized algorithm that finds an α-separator of size k in the given graph, *unless* the graph contains an $(\alpha + \epsilon)$-separator of size strictly less than k, in which case our algorithm finds one such separator. For fixed ϵ, the running time of our algorithm is $n^{O(1)}2^{O(k)}$, which is polynomial for $k = O(\log n)$. For bounded degree graphs (as well as for the case of finding balanced edge separators), we present a deterministic algorithm with similar running time.

Our algorithm involves (among other things) a new concept that we call (ϵ, k)-samples. This is related to the notion of *detection sets* for network failures, introduced by Kleinberg [FOCS 2000]. Our proofs adapt and simplify techniques that were introduced by Kleinberg. As a by-product, our proof improves the known bounds on the size of detection sets. We also show applications of (ϵ, k)-samples to problems in approximation algorithms and rigorous analysis of heuristics.

Categories and Subject Descriptors

G.2.2 [**Discrete Mathematics**]: Graph theory—*graph algorithms*

General Terms

Algorithms

Keywords

Fixed parameter tractability, VC dimension

1. INTRODUCTION

An α-separator for a graph $G(V, E)$ is a set of vertices whose removal from G leaves no connected component larger than $\alpha|V|$, where $\alpha < 1$ is some constant. Finding the minimum size α-separator is NP-hard. In this paper, we shall always denote the number of vertices in the underlying graph by n, and the minimum size of an α-separator by k. The minimum size α-separator can be found in time $n^{O(1)}\binom{n}{k}$ by exhaustive search over all subsets of size k of vertices, which for small values of k behaves like $n^{k+O(1)}$. Therefore, this problem can be solved in polynomial time when k is a constant. On the other hand, there is a reduction that shows that any algorithm for finding an α-separator of size k can be used to find a k-clique in the given graph [19] (see also Section 7.1). This, together with existing results on the maximum clique problem [9] (see also [6]) implies that there is no polynomial-time algorithm for the problem of finding an α-separator of super-constant size (say, $k = O(\log n)$) unless NP has subexponential time algorithms.

To make the minimum α-separator problem more tractable, one may allow for approximate solutions, rather than exact solutions. The approximation may be in terms of *size* (allow for a separator with somewhat more than k vertices), in terms of *balance* (allow each connected component to have somewhat more than αn vertices), or as is most often the case, both. For example, this applies to the approximation algorithms for finding balanced edge cuts in [17, 2] and balanced vertex separators in [8], and they are sometimes referred to as "pseudo-approximations". (The notable exception to this is the approximation algorithm in [10] which approximates the size but is exact in terms of the balance.) In this spirit of relaxing the balance requirements, we consider the following question: given a graph with an α-separator of size $k = O(\log n)$, can one find an $(\alpha + \epsilon)$-separator of size k in polynomial time? When $\alpha \geq 2/3$, we give a positive answer to this question, and prove the following stronger (and perhaps surprising) result: there is a randomized algorithm that given a graph G containing an α-separator of size k, finds such a separator *unless* G also contains $(\alpha + \epsilon)$-separators of size *strictly* less than k, in which case the algorithm will find one such separator. For fixed ϵ, the running time of our algorithm is $n^{O(1)}2^{O(k)}$, which is polynomial for $k = O(\log n)$. Hence our result shows that (when k is small) *the only way to construct hard instances of the minimum α-separator problem is to hide such a separator among other separators with strictly smaller size, though slightly worse balance*.

For readers familiar with parameterized complexity [6], our result can be stated as a positive result on the fixed parameter tractability of finding the minimum α-separator. This problem was shown to be $W[1]$-hard by Marx [19], i.e., under some complexity theoretic assumptions, there is no exact algorithm for this problem with running time $n^{O(1)}f(k)$, where $f(k)$ is some arbitrary function of k that does not depend on n. We show that if the balance requirement of the separator is slightly relaxed, the problem becomes fixed parameter tractable.

The sketch of our algorithm is as follows: we pick a random sample of vertices of size $O(k)$, and "guess" the partition of this set with respect to the optimal separator. We add two vertices s and t and connect them to the vertices on the two sides of this partition. The optimal separator corresponds to a minimum st-cut in this graph. We then decompose this graph into "layers", and use a dynamic programming algorithm to find the most balanced minimum st-cut in this graph.

The proof of our main result has two components. The main technical tool used in the first component is a notion similar to the notion of *detection sets* introduced by Kleinberg [14]. Roughly speaking, an (ϵ, k)-detection set is a set W of vertices of a network that can "detect" k-separation events in which a removal of k vertices disconnects more than ϵn vertices from the rest of the network. This detection is achieved if two vertices of W end up in different connected components, and cannot communicate anymore through the network. We use a strengthening of this notion, which intuitively enables us to relate the balance of any separator on the entire graph to its balance on a small set of vertices which we call an (ϵ, k)-sample. We use a VC dimension argument similar to the one used by Kleinberg to show that a random set of $O(k)$ vertices is likely to be an (ϵ, k)-sample. Our argument is considerably simpler than Kleinberg's proof. Furthermore, since (ϵ, k)-sample is a strictly stronger notion than a detection set, our result answers an open of Kleinberg [14], establishing the optimal dependence of the size of the smallest detection set on k. More precisely, our result implies a bound of $O(k\epsilon^{-1}\log(1/\epsilon))$ on the size of the smallest detection set, improving a bound of $O(k\epsilon^{-1}\log k \log(1/\epsilon))$ by Fakcharoenphol [7] which in turn improves upon the bound of $O(k^3\epsilon^{-1}\log(1/\epsilon))$ by Kleinberg [14]. It is easy to see that $O(k\epsilon^{-1})$ is a lower bound on the size of a detection set, and therefore our bound has the optimal dependence on k for general graphs. For highly connected graphs, and also for weaker notions of detection sets better bounds are given by Gupta (see [7]) and Kleinberg, Sandler, and Slivkins [15].

The second component of our proof is a dynamic programming algorithm based on a new decomposition lemma. This algorithm finds a minimum st-cut that is also balanced, if exists, in time $n^{O(1)}2^{O(k)}$, where k is the size of the minimum st-cut. This is done by showing that the graph can be constructed in steps, in each step adding a number of vertices, so that in each step only a small subset of the vertices can be connected to the rest of the graph.

We also present a deterministic algorithm for finding an $(\alpha+\epsilon)$-separator of size k with running time $n^{O(1)}2^{O(k\log k)}$, and another deterministic algorithm with running time $n^{O(1)}2^{O(k)}$ on bounded degree graphs. For this purpose we introduce the notion of a Steiner t-decomposition, which essentially is a partition of the graph into connected components of size roughly t. The "Steiner" aspect of this decomposition is that in order to make some of the components connected, we may need to use vertices from other components as Steiner points. But if there are no Steiner points, and $t \ll n/k$, then it follows that every separator of size k must leave most of the components connected, a fact that can be used in finding small separators. Notions similar to Steiner t-decompositions were also considered in the context of detection sets.

The rest of this paper is organized as follows: In Section 2 we present the definitions and preliminaries. Section 3 defines the notion of an (ϵ, k)-sample and proves that a random set of $O(k)$ vertices is likely to be an (ϵ, k)-sample. This immediately gives an improved bound on the size of a detection set. We use the results of Section 3 in Section 4 to prove our main result on balanced separators. The deterministic construction is presented in Section 5. Two other applications of the notion of (ϵ, k)-samples are presented in Section 6. Negative results that complement some of our positive results are presented in Section 7.

2. PRELIMINARIES

We shall try to be consistent in using the following notation. The number of vertices in a graph will be denoted by n, and $[n]$ denotes the set of integers from 1 to n. A set of vertices that forms a minimum size separator will be denoted by S, and its size by k. The parameter α will be reserved for expressing the balance of separators (the size of largest connected component divided by n), and ϵ will quantify the error introduced by sampling. A random sample of vertices (that is subsequently used in an algorithm to find small separators) will be denoted by W. Unspecified universal constants will all be denoted by c (even though the constants involved in various statements may differ from each other).

2.1 Key definitions

DEFINITION 2.1. *A set S of vertices $S \subset V$ in a graph $G(V, E)$ is an (α, k)-*separator *if $|S| \le k$, and the induced subgraph $G[V \setminus S]$ that remains when S is removed from G has no connected component larger than $\alpha|V|$.*

Remark. Any set of at most k vertices will be called a k-separator, regardless of whether its removal separates the graph into more than one connected component. A k-separator is called *balanced* if it is an (α, k)-separator, where α is some constant bounded away from 1.

DEFINITION 2.2. *Given a graph $G(V, E)$ and a set of vertices $W \subseteq V$, a set S of vertices is an (α, k, W)-*separator *if $|S| \le k$, and the induced subgraph $G[V \setminus S]$ that remains when S is removed from G has no connected component with more than $\alpha|W|$ vertices from W.*

2.2 VC dimension

The notion of VC dimension (Definition 2.3) was introduced by Vapnik and Chervonenkis [22], who also showed its uses in the context of ϵ-samples (Definition 2.5). The concept of ϵ-nets (Definition 2.4) was introduced by Haussler and Welzl [13]. Lemma 2.6 (for the case of ϵ-nets) is from [4], and improves over bounds given in [13] (as well as in [1], for example) by a factor of $O(\log d)$. For additional information on VC dimension, the reader may consult [18, 5].

DEFINITION 2.3. *A set T is* shattered *by a collection $\mathcal{S}$ of subsets $S_1, S_2 \dots$ of $[n]$, if for every $T' \subset T$ there is some $S_i \in \mathcal{S}$ such that $T \cap S_i = T'$. The* VC-dimension *of $\mathcal{S}$ is the cardinality of the largest set T shattered by $\mathcal{S}$.*

DEFINITION 2.4. *A set W is an ϵ-*net *with respect to a collection $\mathcal{S}$ of subsets $S_1, S_2 \dots$ of $[n]$, if W intersects every set $S_i \in \mathcal{S}$ with $|S_i| \ge \epsilon n$.*

DEFINITION 2.5. *A set W is an ϵ-*sample *with respect to a collection $\mathcal{S}$ of subsets $S_1, S_2 \dots$ of $[n]$, if for every set $S_i \in \mathcal{S}$,*

$$(\frac{|S_i|}{n} - \epsilon)|W| \le |W \cap S_i| \le (\frac{|S_i|}{n} + \epsilon)|W|$$

LEMMA 2.6. *For some universal c, for every set system $\mathcal{S}$ over $[n]$ of VC dimension d, a random set $W \subset [n]$ of size*

$$\frac{c}{\epsilon} \quad d \log \frac{1}{\epsilon} + \log \frac{1}{\delta}$$

has probability at least $1-\delta$ of being an ϵ-net for $\mathcal{S}$. Likewise, a random set $W \subset [n]$ of size

$$\frac{c}{\epsilon^2} \quad d\log\frac{1}{\epsilon} + \log\frac{1}{\delta}$$

has probability at least $1-\delta$ of being an ϵ-sample for $\mathcal{S}$.

PROOF. This lemma for the case of ϵ-nets appears in [4] (see also [16] and [20] for improved constants). We are not aware of a reference with an explicit proof for the case of ϵ-sample, but such a proof follows by straightforward modifications to the proofs in [13, 4, 16]. The proof is sketched below for completeness.

Pick a random set W of t points (the reader may think of t as being smaller than $O(d/\epsilon^3)$). We wish to show that the event that W is not an ϵ-sample has probability at most δ. Consider the case that W is not an ϵ-sample. Then there is a set $S \in \mathcal{S}$ with a μ fraction of the points of $[n]$, but only a $(\mu-\epsilon)$ fraction of the points of W. (Alternatively, it has at least a $(\mu+\epsilon)$ fraction of the points of W, but this case is omitted from this sketch of proof.) Pick at random $T-t$ additional random points, where $T = 2t/\epsilon$. With probability at least half, at least $\mu(T-t)$ of these points are in S. Hence with probability $\delta/2$, a random set of T points contains at least $\mu(T-t)$ points from some set $S \in \mathcal{S}$, but the first t of these T points contain only at most $(\mu-\epsilon)t$ points from S. For a particular S, the probability of this happening can be estimated as follows. Picking T first and conditioning on it having at least $\mu(T-t)$ points from S, the expected number of points from S in the first t of the T points is at least $\mu(T-t)t/T = \mu t(1-t/T) \geq (\mu-\epsilon/2)t$. By bounds on large deviations (using the Chernoff bound [1, 5]), the probability of a deviation of $\epsilon t/2$ behaves like $2^{-\Omega(\epsilon^2 t/\mu)} \leq 2^{-\Omega(\epsilon^2 t)}$. The bound of d on the VC dimension implies (by Sauer's Lemma) that there are at most $d\binom{T}{d} \leq 2^{O(d\log 1/\epsilon)}$ ways of choosing S (for the inequality we used the fact that we will take $T = O(t/\epsilon) \leq O(d/\epsilon^4)$). Hence we want t to satisfy the following inequality:

$$2^{O(d\log 1/\epsilon)} 2^{-\Omega(\epsilon^2 t)} \leq \delta/2$$

which proves Lemma 2.6. $\square$

2.3 Detection sets

The notion of (ϵ,k)-detection sets was defined by Kleinberg [14] as follows. A set S of the vertices of a graph $G(V,E)$ is called an (ϵ,k)-partitioning set if $|S| \leq k$ and $G[V \setminus S]$ contains two sets of nodes A and B, each of size at least ϵn, that are separated (note that A and B need not be connected). A set W of vertices is an (ϵ,k)-detection set if for every (ϵ,k)-partitioning set S, there are nodes $u,v \in W \setminus S$ that lie in different connected components of $G[V \setminus S]$.

Kleinberg showed that a random set of size $O(k^3\epsilon^{-1}\log(1/\epsilon))$ is likely to be an (ϵ,k)-detection set. This bound was improved by Fakcharoenphol [7] to $O(k\epsilon^{-1}\log k\log(1/\epsilon))$. Gupta (as reported in [7]) defined a weaker notion of detection sets and showed that there are weak detection sets of size $O(k\epsilon^{-1})$ using a construction that uses the structure of the graph. Kleinberg et al. [15] improved the bound on the size of detection sets for graphs that are highly connected.

3. RANDOM (ϵ, K)-SAMPLES

We start by defining the concepts of (ϵ,k)-nets and (ϵ,k)-samples. We will use the notion of (ϵ,k)-samples to design our algorithm for finding balanced separators, and the notion of (ϵ,k)-nets to answer Kleinberg's question on detection sets.

DEFINITION 3.1. *A set W of vertices in a graph $G(V,E)$ is an (ϵ,k)-net if for every k-separator S, and for every set of vertices V' that forms a connected component in the induced graph $G[V \setminus S]$,*

1. *If $|V'| \geq \epsilon n$, then V' has at least one vertex from W.*
2. *If $|V'| \leq (1-\epsilon)n - |S|$, then the set $V \setminus (V' \cup S)$ has at least one vertex from W.*

Remark. Definitions for notions such as ϵ-nets (see Definition 2.4) typically have a condition such as condition 1 in Definition 3.1, but do not include a condition similar to condition 2. This last condition is added to Definition 3.1 so as to make it as strong as the notion of k-detecting sets (see Corollary 3.7).

DEFINITION 3.2. *A set W of vertices in a graph $G(V,E)$ is an (ϵ,k)-sample if for every k-separator S, and for every set of vertices V' that form a connected component in the induced graph $G[V \setminus S]$,*

$$\left(\frac{|V'|}{n} - \epsilon\right)|W| \leq |V' \cap W| \leq \left(\frac{|V'|}{n} + \epsilon\right)|W|$$

Remark. One may wonder whether every (ϵ,k)-sample is necessarily an (ϵ,k)-net. This does not simply follow from the definitions due to the following subtlety. A k-separator S may partition G into a connected component V' of size $(1-\epsilon)n-k$ and two more connected components of size $\epsilon n/2$. An (ϵ,k)-sample W might then have all its vertices but one in V', and the remaining vertex in S, and hence is not an (ϵ,k)-net (does not meet condition 2 of Definition 3.1). However, it will turn out that our constructions for (ϵ,k)-samples will also be constructions for (ϵ,k)-nets. (We could have added to Definition 3.2 a condition in the spirit of condition 2 of Definition 3.1 without changing any of the results of this paper, but chose not to do so, so as not to unnecessarily complicate definitions.)

For an (ϵ,k)-sample, we have the following key connection between separators and W-separators.

LEMMA 3.3. *Let W be an (ϵ,k)-sample in a graph $G(V,E)$. Then for every $0 < \alpha < 1$,*

1. *Every (α,k)-separator is also an $(\alpha+\epsilon,k,W)$-separator.*
2. *Every (α,k,W)-separator is also an $(\alpha+\epsilon,k)$-separator.*

PROOF. In terms of the size of separating set S, the restrictions on (α,k) and (α,k,W) separators are identical, namely, $|S| \leq k$. Hence to prove the lemma one only needs to consider the balance parameter.

For item 1, consider an (α,k) separator. There is no connected component with more than αn vertices. Then by the right-hand side inequality in Definition 3.2, no connected component may contain more than $(\alpha+\epsilon)|W|$ vertices of W.

For item 2, consider an (α,k,W) separator. There is no connected component with more than $\alpha|W|$ vertices of W. Then by the left-hand side inequality in Definition 3.2, no connected component may contain more than $(\alpha+\epsilon)n$ vertices. $\square$

In the rest of this section we show that relatively small sets of random vertices are likely to be (ϵ,k)-samples. It is straightforward to show this if the size of the set is allowed to depend logarithmically on n (see [14]). We shall now show that the size of (ϵ,k)-nets and (ϵ,k)-samples need not depend on n. Our proof is a simplification of the proofs leading to the previously best bounds known for detection sets (recall that detection sets are (ϵ,k)-nets), and moreover, improve these bounds by a factor of $O(\log k)$. The previously best bounds for detection sets are cited in [15], based on an improvement of [7] to the bounds in [14].

Given a graph $G(V, E)$ on n vertices and a positive integer $k < n$, we define a collection $\mathcal{S}$ of subsets of V as follows. The set $S' \subset V$ belongs to $\mathcal{S}$ if there is a set $S \subset V$ of at most k vertices disjoint from S' whose removal from V disconnects G, and one of the resulting connected components has either S' or $V \setminus (S \cup S')$ as its set of vertices. (In our definition, S' is either a single connected component, or the union of all connected components but one. In contrast, Kleinberg [14] defined and used a notion of "k-segmental" sets. This notion appears to be a "red herring" and we shall not use it.)

LEMMA 3.4. *The VC-dimension of $\mathcal{S}$ defined above is at most ck, where c is some universal constant independent of k.*

To prove the above lemma we need the following result of Kleinberg [14], which is proved using a theorem of Mader (see Chapter 73 in [21], for example).

LEMMA 3.5. **([14])** *Let $G(V, E)$ be a graph and $T \subset V$. If there are no $k+1$ vertex disjoint paths in G with distinct endpoints in T, then there is a set $W \subset V$ of size at most $3k$ such that its removal from G leaves no connected component with more than one vertex from T.*

Proof of Lemma 3.4. Let T be an arbitrary set of size t. We show that for some universal constant c, if $t \geq ck$, then T cannot be shattered by $\mathcal{S}$. The constant c shall remain unspecified, since we allow ourselves some slackness in the analysis, for the sake of simplicity.

Assume first that there are $k+1$ vertex disjoint paths in G whose endpoints are in T. Then any set S' of at most k vertices leaves two endpoints of one such path in the same connected component. Hence taking T' as having exactly one endpoint from every such path, there is no shattering of T that gives T' as one of the pieces.

Hence we may assume that there are at most k vertex disjoint paths in G whose endpoints are in T. In this case, Lemma 3.5 implies that there is a set $W \subset V$ with at most $3k$ vertices whose removal from G disconnects G, and every vertex of $T \setminus W$ lies in a different connected component. Let $t' = |T \setminus W| \geq t - 3k$. Denote the components that contain vertices from T by $C_1, \ldots, C_{t'}$, and the rest of the components may remain nameless. We name the vertices in $T \setminus W$ by $v_1, v_2, \ldots, v_{t'}$, in agreement with the name of their respective components.

Consider all subsets $T' \subset (T \setminus W)$. We use a counting argument to show that at least one such T' cannot be derived as an intersection $S' \cap T$, with $S' \in \mathcal{S}$.

Let U be a connected component separated from the rest of the graph by a set S of at most k vertices. We give an upper bound of the total number of possibilities for $U \cap (T \setminus W)$. Multiplying this upper bound by 2 will also take care of sets $S' \in \mathcal{S}$ that are complements of a connected component.

There are at most $2^{|W|} \leq 2^{3k}$ ways of choosing the intersection $U \cap W$. Having chosen this intersection determines uniquely which components C_i are connected to $U \cap W$. To disconnect such a component C_i from $U \cap W$, there must be some vertex $v \in (S \cap C_i)$. As $|S| \leq k$, at most k components can be disconnected from $U \cap W$. Hence having fixed $U \cap W$, there are at most $\sum_{i=0}^{k} \binom{t'}{i} \leq (k+1)\binom{t'}{k}$ ways of choosing which vertices of $(T \setminus W)$ remain in U. It follows from the above discussion that T cannot be shattered if:

$$2^{t-3k} > 2 \cdot 2^{3k}(k+1)\binom{t}{k}$$

This inequality is satisfied when $t \geq ck$ (for some sufficiently large constant c), proving the lemma. $\square$

COROLLARY 3.6. *For some universal c, a random set $W \subset V$ of size*

$$\frac{c}{\epsilon}\left(k \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)$$

has probability at least $1 - \delta$ of being an (ϵ, k)-net. A random set $W \subset V$ of size

$$\frac{c}{\epsilon^2}\left(k \log \frac{1}{\epsilon} + \log \frac{1}{\delta}\right)$$

has probability at least $1 - \delta$ of being an (ϵ, k)-sample.

PROOF. We note that ϵ-nets and ϵ-samples (as in Definitions 2.4 and 2.5) with respect to the collection $\mathcal{S}$ used in Lemma 3.4 are (ϵ, k)-nets and (ϵ, k)-samples in the sense of Definitions 3.1 and 3.2. Now the $O(k)$ bound of the VC dimension of $\mathcal{S}$ given in Lemma 3.4 together with the bounds in Lemma 2.6 imply Corollary 3.6. $\square$

The bounds in Corollary 3.6 are best possible up to an $O(\log 1/\epsilon)$ multiplicative factor. See Section 7.3 for details.

COROLLARY 3.7. *In every graph $G(V, E)$ and for every $\epsilon > 0$ and k, there is an (ϵ, k)-detection set of size $O(k\epsilon^{-1}\log(1/\epsilon))$.*

PROOF. By Corollary 3.6 it is enough to show that every (ϵ, k)-net W is also an (ϵ, k)-detection set. This is easy to see, since if S is an (ϵ, k)-partitioning set, then no connected component of $G[V \setminus S]$ contains more than $(1-\epsilon)n - |S|$ nodes. Take one such component V_1'. By definition, W must contain a vertex $v \in V \setminus (V_1' \cup S)$. Now, let V_2' be the connected component of $G[V \setminus S]$ containing u. Again, by definition, W must contain a vertex $u \in V \setminus (V_2' \cup S)$. Therefore, u and v are two vertices of $W \setminus S$ that are separated by S. $\square$

4. FINDING SMALL BALANCED SEPARATORS

In this section, we present an application of (ϵ, k)-samples to the problem of finding small balanced separators in graphs. Marx [19] proved that under certain complexity theoretic assumptions, there is no algorithm with running time $n^{O(1)}f(k)$ that computes an (α, k)-separator in a graph that contains such a separator (For more details, see Section 7.1). In this section, we show that there exists an algorithm with running time $n^{O(1)}2^{O(k)}$ that given a graph that contains an (α, k)-separator either finds such a separator (without relaxing the balance), or computes an $(\alpha + \epsilon, k - 1)$-separator.

THEOREM 4.1. *For $\alpha \geq 2/3$ and arbitrary k, let $G(V, E)$ be an n vertex graph that has an (α, k)-separator. Then for every $\epsilon > 0$, there is a randomized algorithm with expected running time $n^{O(1)}2^{O(k\epsilon^{-2}\log(1/\epsilon))}$ that finds either an (α, k)-separator, or an $(\alpha + \epsilon, k - 1)$-separator in G. In particular, the expected running time is polynomial for every fixed $\epsilon > 0$ whenever $k = O(\log n)$.*

PROOF. Let S be a k-separator that separates G into connected components, where no connected component is larger than αn. Using the fact that $\alpha \geq 2/3$, it follows that the connected components can be arranged in two sides A and B, where no side contains more than αn vertices. (The proof of this fact is standard and omitted. This is the only place where we require $\alpha \geq 2/3$. As is well known, when $\alpha < 2/3$, still each side might need to be as large as $2n/3$ rather than αn, if S separates G into three connected components of size $n/3$.)

Pick a random set W of $O(k\epsilon^{-2}\log(1/\epsilon))$ vertices. By Corollary 3.6, W is likely to be an $(\frac{\epsilon}{2}, k)$-sample. Moreover, it is likely that W does not contain more than $|W|\cdot\frac{|A|}{n}+\sqrt{W} \le |W|(\frac{|A|}{n}+\frac{\epsilon}{2})$ vertices from A (and more than $|W|(\frac{|B|}{n}+\frac{\epsilon}{2})$ vertices from B). Consider all possible ways of partitioning W into two sets A' and B', with no set larger than $(\alpha+\frac{\epsilon}{2})|W|$. We call these partitions *balanced*. At least one balanced partition is *faithful* in the sense that $A' \subset A \cup S$ and $B' \subset B \cup S$. For every balanced partition, find a minimum vertex cut in G separating A' from B'. (Such a cut can be found in polynomial time by flow techniques, by adding a vertex s connected to all vertices in A', a vertex t connected to all vertices in B', and finding the minimum vertex cut separating s from t.) For at least one balanced partition (the faithful partition), the number of vertices in the cut is at most k. Let S' be an arbitrary vertex cut of size at most k found by the above procedure. Then S' is necessarily an $(\alpha+\epsilon, k)$-separator for G, by Lemma 3.3. Furthermore, if the size of the minimum cut separating A' from B' is strictly less than k, then S' will be an $(\alpha+\epsilon, k-1)$-separator for G. Therefore, the only thing that remains is to find an (α, k)-separator in G separating s and t, assuming that the size of the minimum cut separating s and t is precisely k. We call this problem *the balanced minimum st-cut problem*. This problem is NP-hard for general k (as shown in Section 7.2), but we show in Lemma 4.2 below that it can be solved in time $n^{O(1)}2^{O(k)}$.

The algorithm takes time $n^{O(1)}2^{|W|}2^{O(k)}$. It has small probability of failing, if it is unlucky in the choice of the random set W (for example, if W happens not to be an $(\epsilon/2, k)$ sample). In this case, the algorithm can be repeated with a new random choice of W. The expected number of times the algorithm needs to be repeated is at most 2. Hence the expected running time is $n^{O(1)}2^{O(k\epsilon^{-2}\log(1/\epsilon))}$, proving Theorem 4.1 (assuming Lemma 4.2). $\Box$

Remark. Notice that by taking $\epsilon = 1/k$ in the above theorem, we either obtain an (α, k)-separator, or a cut in which the relative loss in the balance is smaller than the relative gain in the size of the cut, and therefore has a better "cut ratio".

The only thing that remains is to solve the **Balanced minimum** st-**cut problem**. Its input is a graph $G = (V, E)$, two non-adjacent vertices s and t in G, and a constant $0 < \alpha < 1$. A desired solution is a vertex cut $S \subset V \setminus \{s, t\}$ of size k, separating s from t such that the components of $G \setminus S$ that contain s and t are of size at most $\alpha|V|$, where k is equal to the size of the smallest vertex cut separating s and t in G. Observe that we do not care here about the size of components that do not contain s or t. The reason is that in the context of Theorem 4.1, these components will not contain any vertices from the (ϵ, k) sample W, and hence their size will be at most $\epsilon|V|$.

The following lemma shows that this problem can be solved in time $n^{O(1)}2^{O(k)}$. The proof of this lemma is based on decomposing the graph into layers and using dynamic programming on these layers.

LEMMA 4.2. *There is a deterministic algorithm that solves the balanced minimum st-cut problem in time $n^{O(1)}2^{O(k)}$, where k is the size of the minimum st-cut and n is the number of vertices in the graph.*

The rest of this section is devoted to the proof of the above lemma. We start with the definition of critical and non-critical vertices. Since the minimum vertex cut between s and t is of size k, Menger's theorem implies that G contains a collection of k vertex-disjoint paths from s to t. For any such collection, every st-cut of size k must contain exactly one vertex from each path. This motivates the definition of critical and non-critical vertices.

DEFINITION 4.3. *A vertex v of G is called* critical *if every collection of k vertex-disjoint paths from s to t contains v. A vertex is* non-critical *if it is not critical. We say that two vertices u and w are* connected *if there is a path between u and w whose vertices (except possibly for u and w) are non-critical.*

We will use the following alternative definition for critical vertices later in the proof. The proof of this proposition is not difficult and is omitted here.

PROPOSITION 4.4. *A vertex v is critical if and only if there is a vertex st-cut of size k containing v.*

We now fix a collection of k vertex-disjoint paths $P_1, P_2, \ldots, P_k$ from s to t. By definition, each critical vertex must be on one of these paths. For each P_i, let $v_{i,1}, \ldots, v_{i,r_i}$ be the sequence of critical vertices of P_i in the order they appear on this path from s to t. To simplify notation, define $v_{i,0} = s$ and $v_{i,r_i+1} = t$ for every i, and think of s and t as critical (even though they are not allowed to be chosen as cut vertices). Let Ω be the set of all k-tuples $\mathbf{a} = (a_1, \ldots, a_k)$ where $0 \le a_i \le r_i$ for every i. In other words, each $\mathbf{a} \in \Omega$ corresponds to one way of selecting one vertex from each P_i (Notice that this does not have to correspond to an st-cut in G, since there are edges and vertices in G that are not on P_i's). For every k-tuple $\mathbf{a} \in \Omega$, we define an induced subgraph $G[\mathbf{a}]$ of G as follows.

DEFINITION 4.5. *The* prefix subgraph $G[\mathbf{a}]$ *defined by* $\mathbf{a} \in \Omega$ *is an induced subgraph of G with the vertex set defined as follows: a critical vertex $v_{i,j}$ is in $G[\mathbf{a}]$ if and only if $j \le a_i$; a non-critical vertex u is in $G[\mathbf{a}]$ if and only if all critical vertices that are connected to u are in $G[\mathbf{a}]$. The* last two layers *of $G[\mathbf{a}]$ is the set of critical vertices $v_{i,j}$ such that $a_i - 1 \le j \le a_i$.*

The idea behind the algorithm is to construct G by a sequence of prefix subgraphs, starting from the graph $G[\bar{\mathbf{1}}]$, and adding one critical vertex (and a number of non-critical vertices) in each step. Furthermore, maintain the invariant that at any step, every critical vertex outside the current subgraph is not connected to any critical vertex other than the ones in the last two layers of the current subgraph. In other words, the last two layers of the current prefix subgraph act as the *interface* of this subgraph to the rest of G. This will enable us to use dynamic programming to solve the balanced minimum st-cut problem, by only keeping track of the "status" of the vertices at the interface of the current prefix subgraph.

This idea is formulated in the following decomposition lemma.

LEMMA 4.6. *There is a sequence $\mathbf{a}^1, \ldots, \mathbf{a}^p \in \Omega$ such that*

(a) $\mathbf{a}^1 = (1, \ldots, 1)$ *and* $\mathbf{a}^p = (r_1, \ldots, r_k)$;

(b) for every $h = 2, \ldots, p$, $\mathbf{a}^h - \mathbf{a}^{h-1}$ is a vector with exactly one entry equal to one, and zero elsewhere; and

(c) for every $h = 1, \ldots, p$, every critical vertex not in $G[\mathbf{a}^h]$ is not connected to any critical vertex of $G[\mathbf{a}^h]$ except possibly to the vertices in the last two layers of $G[\mathbf{a}^h]$.

PROOF. We construct the sequence inductively. It is clear that $G[\mathbf{a}^1]$ satisfies the condition (c) above. Assume we have constructed the sequence up to $\mathbf{a}^{h-1}$. We show that there is some $\mathbf{a}^h$ that satisfies the conditions of the lemma. To this end, we show that there is an i, $1 \le i \le k$, such that $v_{i,\mathbf{a}_i^{h-1}-1}$ is not connected to any critical vertex outside $G[\mathbf{a}^{h-1}]$. Assume, for contradiction, that such an i does not exist. This means that for each i, there is a critical vertex $v_{j,l}$ outside $G[\mathbf{a}^{h-1}]$ (i.e., with $l > \mathbf{a}_j^{h-1}$) that is connected to $v_{i,\mathbf{a}_i^{h-1}-1}$. Now, we construct an auxiliary directed graph H with vertex set $[k]$. For each $i, j \in [k]$, there is a directed edge from i to j in H if $v_{i,\mathbf{a}_i^{h-1}-1}$ is connected to a critical vertex $v_{j,l}$ on P_j with $l > \mathbf{a}_j^{h-1}$. By our assumption, every vertex in H has outdegree at least one, and therefore H has a cycle. Consider the shortest cycle $C = i_0, \ldots, i_{f-1}$ in H. Each edge $i_b i_{(b+1) \bmod f}$ of this cycle corresponds to a path Q_b from $v_{i_b,\mathbf{a}_{i_b}^{h-1}-1}$ to $v_{i_{b+1},\ell_{b+1}}$ for some $\ell_{b+1} > \mathbf{a}_{i_{b+1}}^{h-1}$, such that all internal vertices of this path are non-critical. We show that the existence of this cycle contradicts the fact that $v_{i_0,\mathbf{a}_{i_0}^{h-1}}$ is a critical vertex. By Proposition 4.4, there exists a vertex st-cut S of size k that contains $v_{i_0,\mathbf{a}_{i_0}^{h-1}}$. The removal of S splits the graph into several connected components. We call the vertices in the component that contains s *silver* vertices, and the ones in the component that contains t *tan* vertices. Since S contains exactly one vertex from each P_i and $\ell_0 > \mathbf{a}_{i_0}^{h-1}$, the vertex v_{i_0,ℓ_0} must be a tan vertex. This vertex is connected by the path Q_{f-1} to the vertex $v_{i_{f-1},\mathbf{a}_{i_{f-1}}^{h-1}-1}$. Since all vertices of Q_{f-1} are non-critical and hence not in S, the vertex $v_{i_{f-1},\mathbf{a}_{i_{f-1}}^{h-1}-1}$ is either tan or in S. Therefore, the vertex $v_{i_{f-1},\ell_{f-1}}$ must be tan. Similarly, we can argue that the vertices $v_{i_{f-2},\ell_{f-2}}, \ldots, v_{i_1,\ell_1}$ are all tan. However, v_{i_1,ℓ_1} is connected by the path Q_0 to $v_{i_0,\mathbf{a}_{i_0-1}^{h-1}}$, and the latter vertex must be silver, since the only vertex on P_{i_0} that is in S is $v_{i_0,\mathbf{a}_{i_0}^{h-1}}$. This gives us the desired contradiction.

The above argument shows that there is a vertex $v_{i,\mathbf{a}_i^{h-1}-1}$ that is not connected to any critical vertex outside $G[\mathbf{a}^{h-1}]$. Such a vertex can be found efficiently by trying all possibilities. Now, we simply let $\mathbf{a}_i^h = \mathbf{a}_i^{h-1} + 1$ and $\mathbf{a}_j^h = \mathbf{a}_j^{h-1}$ for every $j \neq i$. It is easy to see that this choice of $\mathbf{a}^h$ satisfies the conditions of the lemma. $\square$

Equipped with the above lemma, we can complete the proof of Lemma 4.2 using dynamic programming. We start by defining the notion of a *valid coloring* for a prefix subgraph.

DEFINITION 4.7. *A* valid coloring *of a prefix subgraph $G[\mathbf{a}]$ is a partial coloring of the vertices of $G[\mathbf{a}]$ with colors silver, tan, and black such that*

- *for each i, there is at most one vertex of $G[\mathbf{a}]$ on P_i that is colored black; furthermore, s and t cannot be colored black;*
- *for each i and $j \leq \mathbf{a}_i$, if there is no $j' \leq \mathbf{a}_i$ such that $v_{i,j'}$ is colored black, then $v_{i,j}$ must be colored silver; if there is such a j', then $v_{i,j}$ must be colored silver if $j < j'$ and tan if $j > j'$;*
- *there are no two connected critical vertices that are colored silver and tan, respectively; and*
- *every non-critical vertex that is connected to at least one silver (tan, resp.) critical vertex is colored silver (tan, resp.); every non-critical vertex that is not connected to any silver or tan critical vertex remains uncolored.*

Notice that the third condition in the above definition guarantees that no non-critical vertex is connected both to a silver and a tan critical vertex, and therefore the fourth condition specifies a well-defined coloring for non-critical vertices.

Proof of Lemma 4.2. Note that a valid coloring of the entire graph G that colors t tan corresponds to a minimum st-cut in G. Conversely, any minimum st-cut in G gives rise to a valid coloring of G that colors t tan. Therefore, the problem is to decide whether there is a valid coloring of G that colors t tan and contains at most αn silver and αn tan vertices. This can be solved using dynamic programming, as sketched below.

We use Lemma 4.6 to construct the sequence $\mathbf{a}^1, \ldots, \mathbf{a}^p$. Based on this sequence, we define a binary $p \times 3^{2k} \times n \times n$ table A as follows. The entry $A(h, y, x_s, x_t)$ is indexed by integers h, x_s, and x_t, and a string $y \in \{s, t, b\}^{2k}$. This entry will be 1 if and only if there is a valid coloring of $G[\mathbf{a}^h]$ with x_s silver and x_t tan vertices that colors the vertices in the last two layers of $G[\mathbf{a}^h]$ according to y.

The entries $A(1, ., ., .)$ can be computed by inspection. We now show an algorithm that computes $A(h, y, x_s, x_t)$, based on the entries $A(h-1, ., ., .)$. The last two layers of $G[\mathbf{a}^h]$ differ from the last two layers of $G[\mathbf{a}^{h-1}]$ in that one vertex (say v) was added and one vertex (say u) was removed. (Technically, if $u = s$ then s may still belong to the last two layers of $G[\mathbf{a}^h]$, but the treatment of this case is only simpler than the case $u \neq s$, and is omitted.) We try all three colors for u. For each color, we first check if in combination with y it violates any of the first three conditions of Definition 4.7. If it does not, we compute the color of all vertices that are in $G[\mathbf{a}^h]$ but not in $G[\mathbf{a}^{h-1}]$ using the fourth condition in Definition 4.7. By condition (c) of Lemma 4.6, the color of any such vertex can be uniquely specified. Given the number of such vertices that are colored silver and tan, we can compute the number of silver and tan vertices that we need in $G[\mathbf{a}^{h-1}]$ to make the total number of silver and tan vertices in $G[\mathbf{a}^h]$ add up to x_s and x_t, respectively. If for at least one guess for the color of u the corresponding entry in $A(h-1, ., ., .)$ indicates that there is a valid coloring of $G[\mathbf{a}^{h-1}]$ with the required number of silver and tan vertices, then we set $A(h, y, x_s, x_t)$ to one. Otherwise, it is set to zero.

Given the table A, one can easily check the existence of a balanced minimum st-cut by checking the entries $A(p, y, x_s, x_t)$ for all strings y that color t tan, and all values $x_s, x_t \leq \alpha n$. It is easy to see that the running time of this algorithm is $n^{O(1)} 2^{O(k)}$. $\square$

5. DETERMINISTIC ALGORITHMS

We showed that a random set of vertices (that is sufficiently large) is likely to be an (ϵ, k)-sample. The fact that (ϵ, k) samples can be chosen in a manner oblivious to the structure of the underlying graph may be useful in some applications (for example, in finding detection sets in large unknown graphs). However, for some other applications (such as the α-separator problem in Section 4) obliviousness is not required, and it might be preferable to have deterministic algorithms.

We do not know of a deterministic way of selecting an (ϵ, k) sample of size $O(k/\epsilon^{O(1)})$ is graph, but suspect that this may be possible. In Section 5.2 we show that this is indeed the case if one considers edge separators rather than vertex separators. In Section 5.3 we show that for vertex separators, this holds for a large family of graphs. In Section 5.4 we no longer explicitly consider (ϵ, k)-samples, and describe a deterministic version of Theorem 4.1, with a somewhat worse dependency on k. All these results are based on the notion of Steiner t-decomposition that is described in Section 5.1. We remark that arguments similar to the ones that we use here were also used to some extent in the context of detection sets. See the part attributed to Gupta in [7].

Remark. For simplicity of the presentation, when dealing with finding (α, k) separators in this section, we shall be content with finding $(\alpha + \epsilon, k)$ separators. These results can be extended to finding either an (α, k) separator or an $(\alpha + \epsilon, k - 1)$ separator using the techniques of Section 4.

5.1 Steiner t-decompositions

We start by defining the concept of a Steiner t-decomposition.

DEFINITION 5.1. *Given $t \geq 2$, a* Steiner t-decomposition *for a graph $G(V,E)$ is a partition of its vertex set into disjoint sets $V_0, V_1, \ldots, V_q$, and a partition of its edge set into disjoint sets $E_0, E_1, \ldots, E_q, E'$, with the following properties:*

1. *$|V_i| < 2t$ for every $0 \leq i \leq q$.*
2. *$|V_i| \geq t$ for every $1 \leq i \leq q$. ($|V_0|$ may be smaller than t, and may also be empty.)*
3. *For every $0 \leq i \leq q$, the subgraph $G[E_i]$ of G induced by the edges E_i forms a tree that contains all vertices of V_i. It may contain also some vertices not in V_i, which are then called* Steiner vertices.

The load of a vertex v in a Steiner decomposition is the number of sets V_i for which v is a Steiner vertex. The load *of a Steiner t-decomposition is the maximum load of any vertex.*

Remark. Definition 5.1 is formulated in a way that makes it easy to use both in the context of edge separators and vertex separators. For edge separators, we want the collection of Steiner trees to be edge-disjoint (a requirement that is not used in our treatment of vertex separators). For vertex separators we want the load to be small (and the notion of load is irrelevant to our treatment of edge separators). It will be the case that in our construction of Steiner t-decompositions, every subgraph $G[E_i]$ contains at most one Steiner point, a fact that will be used in the proof of Theorem 5.7.

LEMMA 5.2. *Let $G(V,E)$ be an arbitrary connected graph and let t be an integer satisfying $2 \leq t < |V|$. Then G has a Steiner t-decomposition in the sense of Definition 5.1. Moreover, a Steiner t-decomposition can be found in polynomial time.*

PROOF. Let T be an arbitrary spanning tree of G. The edges not in this spanning tree will be placed in E'. Choose an arbitrary vertex $r \in V$ as its root, and direct all edges of T away from the root. With every edge $e = (u,v)$ in the spanning tree associate the set of vertices V_e that includes all vertices that are cut off from r in T by removing e (and in particular, includes v but not u), and a set of edges E_e that includes all tree edges induced by V_e. Note that the subgraph $G[E_e]$ induced on E_e is connected and includes all vertices of V_e. Moreover, the spanning tree T restricted to the vertices $(V \setminus V_e)$ is a spanning tree of $G[V \setminus V_e]$, and this is an invariant that we shall maintain throughout the proof.

If for some edge $e = (u,v)$, $t \leq |V_e| < 2t$, then let V_1 and E_1 of the Steiner t-decomposition be V_e and E_e, remove from T all vertices of V_e and their incident edges, and continue inductively with the subgraph that remains (to find the sets $V_2, \ldots, V_q$). If this subgraph contains at most t vertices, take its vertices as V_0 and the remaining tree edges as E_0, and terminate.

If there is no edge e with $t \leq |V_e| < 2t$, then there must be some vertex u (where possibly $u = r$) and edges $e_1 = (u, v_1), \ldots, e_p = (u, v_p)$ (for some $p > 1$) such that for every $1 \leq i \leq p$, $|V_{e_i}| < t$, and moreover $\sum_i^p |V_{e_i}| \geq t$. In this case there must be some $p' < p$ such that

$$t \leq \sum_{i=1}^{p'} |V_{e_i}| < 2t.$$

Let V_1 from the statement of the lemma be $\bigcup_{i=1}^{p'} V_{e_i}$ and let E_1 be all tree edges incident with V_1. The vertex u serves as a Steiner vertex. Remove from T all vertices of V_1, and continue inductively with the subgraph that remains. Again, if this subgraph contains at most t vertices, take its vertices as V_0 and the remaining tree edges as E_0, and terminate.

It is clear that all sets V_i constructed above are of size between t and $2t$ (except for V_0), and are disjoint. The subgraph $G[E_0]$ contains exactly the vertices of V_0, and for $i \geq 1$, every subgraph $G[E_i]$ is connected, and contains the vertices of V_i and at most one more Steiner vertex (denoted by u in the above description). $\square$

5.2 Edge separators

In this section we discuss edge separators, unlike all other sections that deal with vertex separators. As a rule of thumb (which can be supported by formal arguments), problems on edge separators are easier than the corresponding problems on vertex separators. Hence all the results derived in this paper apply not only to vertex separators, but also to edge separators. Moreover, some of the results can be strengthened, as we shall show here.

DEFINITION 5.3. *A set S of edges $S \subset E$ in a graph $G(V,E)$ is an (α, k)* edge separator *if $|S| \leq k$, and the graph $G(V, E \setminus S)$ that remains when S is removed from G has no connected component larger than $\alpha|V|$.*

Other definitions in this paper (such as the notions of (ϵ, k) nets and samples) generalize in a straightforward way to the case of edge separator, and will not be repeated.

THEOREM 5.4. *For every connected graph $G(V,E)$, for every $1 \leq k \leq |V|$ and $\epsilon > 0$, the following holds with respect to edge separators.*

1. *There is a deterministic polynomial time algorithm for choosing an (ϵ, k)-net of size $O(k/\epsilon)$.*
2. *There is a deterministic polynomial time algorithm for choosing an (ϵ, k)-sample of size $O(k/\epsilon^2)$.*
3. *If the graph has an (α, k) edge separator (with $\alpha \geq 2/3$) then there is a deterministic algorithm with running time $n^{O(1)} 2^{O(k/\epsilon)}$ that finds an $(\alpha + \epsilon, k)$ edge separator in G.*

PROOF. Let $t = \epsilon n / 4k$ (rounded to the nearest integer). Consider sets $V_0, \ldots, V_q$ with $t \leq |V_i| < 2t$ for $1 \leq i \leq q$, as implied by Lemma 5.2. Note that necessarily, $q \geq n/2t = 2k/\epsilon$. Consider an arbitrary edge separator in G of size k. The k edges of the cut may be in at most k of the subgraphs $G[E_i]$, because these subgraphs are edge-disjoint. (If fact, some edges may be in E' rather than in one of these subgraphs, a fact that may become useful if we ever come to care about the constants in the proof.) At most $2tk = \epsilon n/2$ vertices are in these subgraphs. At least $q - k$ of the sets V_i are not disconnected by edges of the cut, and will be called *good*. Every good set must reside entirely in one connected component.

To prove item 1 of the Lemma ((ϵ, k)-nets), pick one vertex from every set V_i. Every connected component larger than ϵn must contain at least one good set, and hence at least one vertex from the net. Likewise, every connected component smaller than $(1 - \epsilon)n - k$ does not contain at least one good set, and hence cannot have all the net points.

To prove item 2 ((ϵ, k)-samples), consider first the case that all sets V_i have the same size t (or sizes between t and $(1 + \epsilon/2)t$). Then the (ϵ, k)-net described above is also an (ϵ, k)-sample. If the sizes of sets vary more radically (they may differ by a factor of 2) pick $|V_i|/t\epsilon$ vertices from each set V_i (rounded to the nearest integer). Every connected component contains its fair fraction of the sample points, up to an additive error of $O(\epsilon)$. The source of error is twofold: rounding effects, and the fact that an ϵ fraction of sets might not be good.

To prove item 3 (fixed parameter tractability), let S be an optimal k edge separator partitioning G into sides A and B of size at most αn. Guess which sets V_i contain separator edges (at most k sets), and for the rest of the sets (the good sets), guess which of them are on side A of the cut and which are on side B. Now find a minimum edge cut in G between the vertices already placed in side A and those already placed in side B. Its size is at most k, and at most $\epsilon n/2$ vertices change side compared to the optimal separator. The running time of the above algorithm is at most $n^{O(1)}3^q$. $\square$

5.3 Bounded degree spanning trees

For a connected graph G, let $\Delta_T(G)$ denote the maximum degree in a spanning tree whose maximum degree is smallest. For example, $\Delta_T(G) = 2$ if and only if the graph has a hamiltonian path. In any graph G, a spanning tree of maximum degree $\Delta_T(G) + 1$ can be found in polynomial time [12].

LEMMA 5.5. *There is a polynomial time algorithm that in every graph G finds a Steiner t-decomposition of load at most $\Delta_T(G)/2$.*

PROOF. In the proof of Lemma 5.2, take T to be a tree of maximum degree $\Delta_T(G) + 1$. For every Steiner vertex v, at least one of its edges connect v to the set V_i containing v, and every set V_j for which v is a Steiner vertex uses two of v's edges. As all the sets E_j are disjoint, the proof follows. $\square$

COROLLARY 5.6. *For every connected graph $G(V, E)$, for every $1 \le k \le |V|$ and $\epsilon > 0$, the following holds with respect to vertex separators.*

1. *There is a deterministic polynomial time algorithm for choosing an (ϵ, k)-net of size $O(k\Delta_T(G)/\epsilon)$.*
2. *There is a deterministic polynomial time algorithm for choosing an (ϵ, k)-sample of size $O(k\Delta_T(G)/\epsilon^2)$.*
3. *If the graph has an (α, k) separator (with $\alpha \ge 2/3$) then there is a deterministic algorithm that finds an $(\alpha + \epsilon, k)$ separator in time $n^{O(1)}2^{O(k\Delta_T(G)/\epsilon)}$.*

PROOF. The proof is similar to the proof of Theorem 5.4, with the following changes. Use the Steiner t-decomposition promised in Lemma 5.5. Pick $t = \epsilon n/2k\Delta_T(G)$. Use the fact that every separator vertex v separates at most $1 + \Delta_T/2$ sets V_i. Details omitted. $\square$

Remark. If G is Hamiltonian, and furthermore, a Hamiltonian path is given, then by cutting the path into segments of size t one gets a Steiner t-decomposition of load 0. The fact that all sets of the decomposition (except for at most one) has size exactly t leads to a deterministic construction of (ϵ, k)-samples of size $O(k/\epsilon)$, in this special case.

5.4 A deterministic algorithm for k-separators

Here we use the notion of Steiner t-decomposition to present a deterministic version of Theorem 4.1, with a slightly worse dependence of k (and a better dependence on ϵ).

THEOREM 5.7. *For $\alpha \ge 2/3$ and arbitrary k, let $G(V, E)$ be an n vertex graph that has an (α, k)-separator. Then for every $\epsilon > 0$, there is a deterministic algorithm that finds an $(\alpha + \epsilon, k)$-separator in time $n^{O(1)}2^{O(k(\log k+1/\epsilon))}$. In particular, the running time is polynomial for every fixed $\epsilon > 0$ whenever $k = O(\log n/\log\log n)$.*

PROOF. Let S be a k-separator that separates G into connected components, where no connected component is larger than αn. Using the fact that $\alpha \ge 2/3$, it follows that the connected components can be arranged in two sides A and B, where no side contains more than αn vertices.

Find a Steiner t-decomposition for G with $t = \epsilon n/2k$, and hence $q \le 2k/\epsilon$. Observe that if none of the Steiner vertices are in S, the proof of item 3 of Theorem 5.4 can serve also as a proof for the current theorem. Hence it remains to deal with the case that some Steiner vertices are in S. There are at most q Steiner vertices (because our constructions of Steiner t-decompositions have the property that every induced subgraph $G[E_i]$ has at most one Steiner vertex). Let the computation now branch into $q + 1$ possibilities, depending on which is the first Steiner vertex that is in S, if any. In branch 0 (corresponding to no Steiner vertex in S), proceed as in the proof of item 3 of Theorem 5.4. In every other branch, remove the corresponding Steiner vertex from the graph, and repeat the above algorithm (of selecting a Steiner t-decomposition and branching) to search for a $k-1$ size separator in the new graph. (In fact, this needs to be done only on the largest component of the new graph, and only if this component has at least $(\alpha + \epsilon)n$ vertices.) After at most k iterations of this process, no more Steiner vertices can be in S, and the process ends.

Altogether, the branching process results in a tree with q^k nodes. The amount of computation per node of the tree is at most $n^{O(1)}3^q$, as in the proof of item 3 of Theorem 5.4. The fact that $q = O(k/\epsilon)$ completes the proof of Theorem 5.7. $\square$

6. OTHER APPLICATIONS

In this paper, we showed the application of our results on (ϵ, k)-samples to the problem of finding balanced separators (Section 4) and improving the best known bound on the size of a detection set. However, our original motivation for considering (ϵ, k)-samples was somewhat different. It relates to improving the approximation ratio for balanced separators. See Section 6.1. The possible use of (ϵ, k)-samples in local search heuristics is discussed in Section 6.2.

6.1 Approximation algorithms

The following theorem was proved by Arora, Rao and Vazirani [2] for edge separators, and by Feige, Hajiaghayi and Lee [8] for vertex separators, which is the version we discuss here. (The reader may wish to recall Definition 2.2.)

THEOREM 6.1. *For every constants $0 < \alpha < 1$ and $\epsilon > 0$, there is a randomized polynomial time algorithm that for every graph $G(V, E)$ and set $W \subset V$ finds an $(\alpha + \epsilon, k\sqrt{\log|W|}, W)$-separator if G has an (α, k, W)-separator.*

As a special case, when $W = V$ Theorem 6.1 offers a (pseudo) approximation ratio of $O(\sqrt{\log n})$ for (α, k)-separators. We shall improve upon this ratio when k is small.

THEOREM 6.2. *For every constants $0 < \alpha < 1$ and $\epsilon > 0$, there is a randomized polynomial time algorithm that for every graph $G(V, E)$ with an (α, k)-separator finds an $(\alpha+\epsilon, k)$-separator of size $O(k\sqrt{\log k})$.*

PROOF. Pick a set W of $O(k/\epsilon^3) = O(k)$ (because ϵ is constant) vertices in G. By Corollary 3.6, it has high probability of being an (ϵ, k)-sample. Then by Lemma 3.3, G has an $(\alpha+\epsilon, k, W)$-separator. By Theorem 6.1, an $(\alpha+2\epsilon, O(k\sqrt{\log k}), W)$ separator can be found in polynomial time. By Lemma 3.3, this is also an $(\alpha + 3\epsilon, O(k\sqrt{\log k}))$-separator in G. Scaling ϵ by a factor of 3 proves Theorem 6.2. $\square$

We remark that a somewhat different rounding technique for the semidefinite relaxations given in [2, 8] can be used to directly prove Theorem 6.2, without using (ϵ, k)-samples. This rounding technique is based on a deterministic choice of an ϵ-net with respect to the geometry of the solution to the semidefinite program. See [8] for details.

6.2 Rigorous analysis of a local search heuristic

We show how the notion of (ϵ, k)-samples can be used in combination with some local search based heuristics. For simplicity of the presentation, it refers to edge separators rather than vertex separators. The edge separator of size k is assumed to partition the graph into two parts, named A and B. Hence it will be referred to as a *2cut*.

A 2cut (A, B) of size k is *t-optimal* if for every 2cut (A', B') of size less than k it must hold that $|A \oplus A'| > t$ (where $A \oplus A'$ denotes the set of vertices that need to change sides so as to make A equal to A'). The range of parameters of interest for t-optimality requires $t < \min[|A|, |B|]$. Checking t-optimality can be done in time proportional to $\binom{n}{t}$ by exhaustive search. This time is polynomial in n only when t is constant. When t is not a small constant, the following randomized algorithm may sometimes be useful in testing for t-optimality.

1. Select random sets $S \subset A$ and $R \subset B$ with $|S|/|A| \simeq |R|/|B| \ll 1/t$.
2. Unify S into one vertex s, unify R into one vertex r, and find a minimum (s, r)-cut.
   (a) If the cut found is of size less than k, and it differs from the cut (A, B) in the location of at most t vertices, conclude that (A, B) is not t-optimal.
   (b) If the cut found is of size at least k, then conclude that (A, B) is probably t-optimal.
   (c) If the cut found is of size less than k, and it differs from the cut (A, B) in the location of more than t vertices, then abort.

The output of the algorithm in step 2(a) is certainly correct. As for step 2(b), here the rational is that if there is a cut (A', B') smaller than k that differs from (A, B) in the location of at most t vertices, neither S nor R is likely to contain any of these vertices. In this case, the (s, r)-cut found in step 2 will be at most as large as the cut (A, B). Thus not finding a cut smaller than k is (probabilistic) evidence that (A, B) is t-optimal.

The problematic part of the above algorithm is that if the algorithm reaches step 2(c), then its output is not informative regarding the t-optimality of the cut (A, B). This can be remedied in some special cases, using the notion of (ϵ, k)-samples. Assume that the cut (A, B) is balanced, and furthermore, that the cut size k is much smaller than n/t. In this case $|S|, |R| \gg k$, and Corollary 3.6 implies that $S \cup R$ is an (ϵ, k)-net. (In fact, since we are considering edge separators here, we can use instead item 2 in Theorem 5.4, but rather than picking vertices arbitrarily within each set V_i, pick them at random.) Now if step 2(c) is reached, the cut produced will be balanced. Hence with high probability, either the algorithm correctly declares the cut (A, B) to be t-optimal, or it finds another balanced cut of smaller size.

The above discussion relates to a notion of *stability* of inputs that is investigated by Bilu and Linial [3]. Under one plausible definition of stability, a bisection (A, B) (with $|A| = |B|$) of size k is called *stable* if for every set S of vertices the 2cut (A', B') with $A' \oplus A = S$ is of size at least $k + |S|$, provided that $|A'|, |B'| \geq n/4$. (As a convention we assume that $|A' \oplus A| \leq |A' \oplus B|$).

THEOREM 6.3. *Let $G(V, E)$ be a graph with a minimum bisection (A, B) of size $k \ll \sqrt{n}/(\log n)^{1/4}$, and moreover, assume that this bisection is stable in the sense defined above. Then this bisection can be found in polynomial time.*

PROOF. We sketch the proof. Using [2], one can find in polynomial time a balanced 2-cut (A', B') of size at most $O(k\sqrt{\log n}) \ll \sqrt{n}(\log n)^{1/4}$. Stability then implies that $|A' \oplus A| \ll \sqrt{n}(\log n)^{1/4}$. Now we take an (ϵ, k)-sample W of size $O(k) \leq O(\sqrt{n}/(\log n)^{1/4})$ (here we take ϵ to be a fixed small constant, hidden in the O notation). By the union bound it is likely that $W \cap A = W \cap A'$. Hence the 2cut found by the algorithm for t-optimality sketched above will be of size at most k, and furthermore, by the fact that W is an (ϵ, k)-sample the sides of the cut will be similar to those of A' and B' (up to ϵn vertices). By stability, the only cut with these properties is the minimum bisection (A, B). $\square$

7. SOME NEGATIVE RESULTS

7.1 Hardness of finding balanced separators

A variation on the following theorem appears in [19]. We sketch its proof for completeness.

THEOREM 7.1. *For every k and $\alpha \geq 1/2$, there is a polynomial time reduction from the problem of finding a clique of size k in a graph, to the problem of finding an (α, k)-separator.*

PROOF. Let $G(V, E)$ be a graph in which one seeks to find a clique of size k. Without loss of generality, assume that vertex 1 is known to belong to a clique of size k. (There are at most n vertices to choose from, and one may simply try all of them.) Now construct the following graph H. For every vertex $i \in V$ introduce a vertex v_i in H. For every edge $(i, j) \in E$ introduce a vertex v_{ij} in H. In addition, introduce a set W of vertices, where $|W|$ is chosen so that

$$|W| + \binom{k}{2} + k = (1 - \alpha)(|V| + |E| + |W|) \qquad (1)$$

For every $1 \leq i < j \leq n$, put in H the edges (v_i, v_j). (Namely, the set V forms a clique in H.) For every $(i, j) \in E$, put in H the edges (v_{ij}, v_i) and (v_{ij}, v_j). In addition, connect all vertices of W to v_1. This completes the description of H.

It is not hard to see that S, the most balanced k-separator in H must include v_1 and $k - 1$ other vertices from V. To have only $\alpha(|V| + |E| + |W|)$ vertices remain in the largest connected component, it must be that the vertices of S form a k-clique in G, by Equation (1). $\square$

Theorem 7.1 shows the difficulty in designing algorithms running in time $n^{O(1)}f(k)$ for finding (α, k)-separators, for some function k. (Theorem 4.1 gives an algorithm of running time $n^{O(1)}2^{O(k)}$, but this algorithm might fail to find (α, k)-separators in graphs that also have $(\alpha + \epsilon, k - 1)$-separators.) Such algorithms are not known for finding k-cliques, and moreover, having such algorithms for k-cliques would have far reaching consequences in computational complexity. See [6] for more details.

7.2 NP-hardness of the balanced min st-cut problem

In this section we prove that the balanced min st-cut problem is NP-hard, even for the case of edge cuts. The proof for the case of vertex cuts is similar (and in fact simpler). For directed graphs, Feige and Yahalom [11] proved that the problem of finding an (α, k)-separator is NP-hard, even when $k = 0$, and therefore the problem is not even fixed parameter tractable.

The proof is by a reduction from CLIQUE. Let G be a graph, and k be an integer. In order to find a k-clique in G, we construct an instance of the balanced min st-edge-cut problem. For simplicity, we allow the vertices to have integer weights and allow parallel edges. It is possible to remove these assumptions by replacing parallel edges by parallel paths and adjusting the weights, and replacing a vertex of weight w by a vertex attached to a clique of size $w-1$. The set of vertices in our instance is $\{s,t\} \cup V(G) \cup E(G)$. For every $v \in V(G)$ and $e \in E(G)$, there are $\deg_G(v)$ parallel edges from s to v, two edges from e to t, and an edge between v and e if v is an endpoint of e. The weight of all vertices in $V(G)$ are W (some integer larger than k^2), and the weight of other vertices are one. Furthermore, we set α such that α times the total weight of the graph is equal to $Wk + \binom{k}{2} + 1$. It is not difficult to see that the existence of a minimum edge-cut in this graph that contains an α fraction of the vertices on the s-side is equivalent to the existence of a k-clique in G. This is based on the fact that the graph is a disjoint union of $2|E(G)|$ length-3 paths between s and t. Details of the proof are left to the full version of the paper.

7.3 A lower bound for (ϵ, k)-samples

We show that Corollary 3.6 is optimal up to a factor of $O(\log 1/\epsilon)$.

THEOREM 7.2. *For every k and ϵ, there are infinitely many graphs for which a set W of vertices chosen uniformly at random is likely not to be an (ϵ, k)-sample, unless it contains $\Omega(k/\epsilon^2)$ vertices.*

PROOF. For n a multiple of k, let $P(n/k)$ denote the graph that is composed of k vertex disjoint paths, each with n/k vertices, and in which the leftmost points of every two paths are connected by an edge, and the rightmost points of every two paths are connected by an edge. (In other words, there are two k-cliques, and k equal-length vertex disjoint paths connecting them.)

Pick at random a set W of kt^2 vertices. In expectation, every path contains t^2 vertices from W, with standard deviation $\Omega(t)$. In each path, exactly one vertex will be included in the k-separator S. If the path has less than t^2 vertices from W, the left-most vertex of the path is placed in S. If the path has more than t^2 vertices from W, the right-most vertex of the path is placed in S. Standard probabilistic analysis shows that for the connected component C to the right of S the following holds with high probability:

$$|W \cap C| \leq |W|(\frac{|C|}{n} - \Omega(1/t))$$

Hence W is not an (ϵ, k)-sample unless $t = \Omega(1/\epsilon)$, proving Theorem 7.2. $\square$

The observant reader may have noticed that the proof without change applies also to edge separators. Recall however that Theorem 5.4 shows that edge separators of size k can be found in time $n^{O(1)}2^{O(k/\epsilon)}$. Hence the lower bound in Theorem 7.2 on the size of (ϵ, k)-samples is perhaps not very informative regarding the prospects of improving the dependence on ϵ in Theorem 4.1.

Acknowledgements

The first author is on leave from the Weizmann Institute where his work is supported in part by grants from the Israeli Science Foundation (ISF) and the German Israeli Foundation (GIF). Special thanks to Anupam Gupta, who directed us to Kleinberg's work on detection sets. Shimon Kogan sent us very useful comments on a preliminary version of the paper, and independently derived Theorem 5.7. We would also like to thank MohammadTaghi Hajiaghayi, James Lee, Laci Lovasz, Amin Saberi and Kunal Talwar for useful discussions.

8. REFERENCES

[1] N. Alon and J. Spencer. *The Probabilistic Method.* Wiley-Interscience, 2000.

[2] S. Arora, S. Rao, U. Vazirani. "Expander flows, geometric embeddings and graph partitioning." *STOC 2004*, 222–231.

[3] Y. Bilu, N. Linial. "Are stable instances easy?". *Manuscript*, 2004.

[4] A. Blumer, A. Ehrenfeucht, D. Haussler, M. Warmuth. "Learnability and the Vapnik-Chervonenkis dimension." *J. ACM 36(4)*, 929–965 (1989).

[5] B. Chazelle. *The Discrepency Method.* Cambridge University Press, 2000.

[6] R. Downey, M. Fellows. *Parameterized Complexity*. Springer, 1999.

[7] J. Fakcharoenphol. MSc thesis, Berkeley.

[8] U. Feige, M. Hajiaghayi, J. Lee. "Improved approximation algoritms for minimum-weight vertex separators". *Proceedings of 37th STOC*, 2005, 563–572.

[9] U. Feige, and J. Kilian. "On Limited versus Polynomial Nondeterminism." Chicago J. Theor. Comput. Sci., 1997.

[10] U. Feige, R. Krauthgamer. "A Polylogarithmic Approximation of the Minimum Bisection." *SIAM J. Comput. 31(4)*, 1090–1118 (2002).

[11] U. Feige, O. Yahalom. "On the complexity of finding balanced oneway cuts". *Inf. Process. Lett. 87(1)*, 1–5 (2003)

[12] M. Furer, B. Raghavachari. "Approximating the minimum degree spanning tree to within one from the optimal degree." *In Proceedings of the 3rd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '92)*, pages 317–324, 1992.

[13] D. Haussler, E. Welzl. "Epsilon-Nets and Simplex Range Queries." *Discrete and Computational Geometry 2*, 127–151 (1987).

[14] J. Kleinberg. "Detecting a Network Failure." Internet Mathematics **1**, No. 1, 37–56, 2003. Conference version appeared in *FOCS 2000*.

[15] J. Kleinberg, M. Sandler, A. Slivkins. "Network failure detection and graph connectivity." *SODA 2004*, 76–85.

[16] J. Komlos, J. Pach, G. Woeginger. "Almost Tight Bounds for epsilon-Nets." *Discrete and Computational Geometry 7*, 163–173 (1992).

[17] F. T. Leighton, S. Rao. "Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms." *J. ACM 46(6)* 787–832 (1999).

[18] J. Matousek. *Geometric Discrepency.* Springer, 1999.

[19] D. Marx. "Parameterized graph separation problems." *International Workshop on Parameterized and Exact Computation (IWPEC)*, 2004 (Bergen), 71–82, Lecture Notes in Comput. Sci., 3162, Springer, Berlin, 2004.

[20] J. Pach, P. Agarwal. *Combinatorial Geometry*. Wiley, 1995.

[21] A. Schrijver. *Combinarorial Optimization (Volume C).* Springer, 2003.

[22] V. Vapnik, A. Chervonenkis. "On the uniform convergence of relative frequencies of events to their probabilities." *Theory of Probability and its Applications* 16 (1971), 264–280.