Simplex Transformations and the Multiway Cut Problem

Niv Buchbinder* Roy Schwartz† Baruch Weizman‡

November 1, 2016

Abstract

We consider Multiway Cut, a basic graph partitioning problem in which the goal is to find the minimum weight collection of edges disconnecting a given set of special vertices called terminals. Multiway Cut admits a well known simplex embedding relaxation, where rounding this embedding is equivalent to partitioning the simplex. Current best known solutions to the problem are comprised of a mix of several different ingredients, resulting in intricate algorithms. Moreover, the best of these algorithms is too complex to fully analyze analytically and its approximation factor was verified using a computer. We propose a new approach to simplex partitioning and the Multiway Cut problem based on general transformations of the simplex that allow dependencies between the different variables. Our approach admits much simpler algorithms, and in addition yields an approximation guarantee for the Multiway Cut problem that (roughly) matches the current best computer verified approximation factor.

1 Introduction

The Multiway Cut problem in undirected graphs is a prime example for the success of the geometric embedding approach, a prevalent tool in the design of approximation algorithms for many NP-hard graph cut problems [3, 4, 7, 12, 20, 26, 27]. In this problem we are given an edge weighted undirected graph $G = (V, E)$, $w : E \to \mathcal{R}^+$, and a set $T = \{t_1, t_2, \ldots, t_k\} \subseteq V$ of k terminals. The goal is to find a minimum weight subset of edges $X \subseteq E$ such that all terminals are disconnected in $(V, E \setminus X)$. When $k = 2$ the Multiway Cut problem is simply the minimum $\{s, t\}$-cut problem in undirected graphs, whereas for $k = 3$ it is already known to be NP-hard [11].

The first approximation algorithm for the Multiway Cut problem was given by Dalhaus *et al.* [11]. It is a simple combinatorial heuristic: for each terminal t_i, compute a minimum cut separating it from all other terminals. Dalhaus *et al.* [11] showed that the union of the $k - 1$ cheapest cuts (of the k cuts computed) is a $2(1 - 1/k)$-approximation. Călinescu *et al.* [7] suggested a geometric relaxation for the Multiway Cut problem. In this relaxation each vertex $u \in V$ is embedded into the k-dimensional simplex $\Delta_k = \{x \in \mathcal{R}^k : x \geq 0,\ \sum_i x_i = 1\}$, while the terminals are mapped bijectively to the vertices of Δ_k. Călinescu *et al.* [7] observed that any rounding of this relaxation is in fact a partitioning of the simplex into k parts, one for each terminal (or equivalently a vertex of Δ_k). They presented a partitioning algorithm achieving an approximation guarantee of $3/2 - 1/k$. Building upon this result, Karger *et al.* [21] managed to obtain an improved guarantee of $1.3438 - \varepsilon_k$, where ε_k is a decreasing function of k that tends to 0 as k increases.

The first improvement over the work of Karger *et al.* [21] was given by Buchbinder *et al.* [5]. They presented a rounding framework that was based on two ingredients: the exponential clocks algorithm and the Călinescu *et al.* algorithm [7] (which we will denote from this point onwards as the CKR algorithm). It is worth noting that [5] proved that one can equivalently use the algorithm of Kleinberg and Tardos for Uniform Metric Labeling [25] instead the exponential clocks algorithm. [5] provided a simple $4/3$-approximation for Multiway Cut via the above approach. Additionally, they showed that within their framework of

*Statistics and Operations Research Dept., Tel Aviv University, Israel. E-mail: niv.buchbinder@gmail.com. Research is supported by ISF grant 1585/15 and BSF grant 2014414.

†Computer Science Department, Technion, Israel. E-mail: schwartz@cs.technion.ac.il. Research is supported by ISF grant 1336/16.

‡The Blavatnik School of Computer Science, Tel Aviv University, Israel. E-mail: baruch.weiz@gmail.com. Research is supported by ISF grant 1585/15 and BSF grant 2014414.

mixing the above two ingredients, a slightly better approximation of 1.32388 is possible.

Using this framework, Sharma and Vondrák [30] presented an improved algorithm that achieves an approximation of $\frac{1}{4}\left(3+\sqrt{5}\right)$. Remarkably, [30] also presented a tight lower bound on any algorithm within the Buchbinder *et al.* framework, matching their $\frac{1}{4}\left(3+\sqrt{5}\right)$-approximation. To further improve the approximation factor, Sharma and Vondrák [30] introduced a third ingredient to the mix of algorithms: the descending threshold algorithm. This resulted in an improved approximation of $\frac{1}{13}(10+4\sqrt{3}) \approx 1.30217$. Furthermore, [30] introduced yet a fourth ingredient to the mix of algorithms: the independent threshold algorithm of Karger *et al.* [21]. Unfortunately, the resulting algorithm was too complicated to fully analyze analytically, and using a computer they managed to estimate that it leads to an approximation of 1.2965.

The question this paper addresses is whether one can introduce a new approach to simplex partitioning and the `Multiway Cut` problem that is *simple*. As shown by Sharma and Vondrák [30], the introduction of a third and a fourth ingredient to the mix of algorithms improves the approximation factor. However, this addition of algorithms comes at a price since the resulting mixed algorithm is quite intricate and is too complex to fully analyze analytically.

1.1 Our Results We present a new and conceptually simple approach to simplex partitioning and the `Multiway Cut` problem. In contrast to current state of the art algorithms [30], our approach is based solely on the two original ingredients of the Buchbinder *et al.* framework: the exponential clocks and CKR algorithms. In addition to being much simpler than state of the art algorithms, we can use our new approach to provide a provable approximation of $^{297}/_{229} \approx 1.29694$. This (roughly) matches the 1.2965 approximation guarantee of [30], whose current proof is partly verified by a computer.

Using our approach we present two algorithms. The first is remarkably simple and serves as a case study: it illustrates how using only the two original ingredients of the Buchbinder *et al.* framework it is possible to achieve an approximation strictly better than the $\frac{1}{4}\left(3+\sqrt{5}\right)$ lower bound of [30]. This is summarized in the following theorem.

THEOREM 1.1. *Using only the exponential clocks and CKR algorithms, it is possible to achieve an approximation of* $^{17}/_{13}$ *for the* `Multiway Cut` *problem.*

The second algorithm builds upon the first one and achieves a provable approximation of $^{297}/_{229} \approx 1.29694$. This is summarized in the following theorem.

THEOREM 1.2. *Using only the exponential clocks and CKR algorithms, it is possible to achieve an approximation of* $^{297}/_{229} \approx 1.29694$ *for the* `Multiway Cut` *problem.*

1.2 Our Technique The original framework of [5] uses only two basic and simple ingredients: the exponential clocks and CKR algorithms. Additionally, the $\frac{1}{4}\left(3+\sqrt{5}\right)$ lower bound of [30] applies to any algorithm within this framework. One might ask how can we go below this lower bound by only using the same two ingredients?

The answer to the above question is that prior to partitioning the simplex we transform it. Transforming a solution obtained from a relaxation, prior to rounding it, is not a novel idea. Perhaps one of the most well known examples where such an approach yields a simple and elegant solution is the $^3/_4$-approximation algorithm of Goemans and Williamson for the `Max SAT` problem [19]. It is important to note that the framework of Buchbinder *et al.* [5] already uses such an approach. However, as is the case with the above mentioned solutions to both `Multiway Cut` and `Max SAT`, only a very restrictive type of transformation, that treats every variable *independently*, is used. Indeed, the $\frac{1}{4}\left(3+\sqrt{5}\right)$ lower bound of [30] applies only for this very restrictive type of transformations. We, on the other hand, employ a general transformation of the simplex that *allows dependencies* between the different variables. To the best of our knowledge, all cases in which the above transformation approach is used utilize only the restrictive type of transformations that handle each variable independently.

1.3 Related Work The geometric relaxation of Călinescu *et al.* [7] carries much importance, since Manokaran *et al.* [28] proved that its integrality gap can be translated to a hardness result for the `Multiway Cut` problem with the exact same value, assuming the unique games conjecture. Hence, this suggests that the best possible approximation guarantee for the `Multiway Cut` problem can be obtained by rounding the geometric relaxation of Călinescu *et al.* [7]. Freund and Karloff [15] showed that this relaxation has an integrality gap of at least $8/(7+1/(k-1))$. Furthermore, the latter was recently improved to $6/(5+1/(k-1))-\varepsilon$ (for any constant ε) by Angelidakis *et al.* [1]. An additional lower bound was given by Dahlhaus *et al.* [11], showing that `Multiway Cut` is APX-hard. Thus, there exists a constant $c > 1$ such that no polynomial-time algorithm can find a solution within a factor of c of the optimum, unless P=NP.

We also note that some of the techniques, developed in the context of `Multiway Cut`, have found additional applications. Intuitively, the main idea of the algorithm of Călinescu *et al.* [7] is to iterate over the terminals in a random order and cut an ℓ_1 sphere of random radius around each terminal, thus partitioning the simplex. This idea was extended to general metrics, providing improved approximations for the `0-Extension` problem [6, 13] (whose study was originated by Karzanov [23]) and the probabilistic approximation of metrics by tree metrics [14]. For the `0-Extension` problem, Călinescu *et al.* [6] provide an approximation of $O(\log k)$, which was later improved by Fakcharoenphol *et al.* [13] to $O(\log k / \log\log k)$. Both results are obtained by rounding the *metric completion* relaxation.

In addition to the works already mentioned, several other special cases and variants of `Multiway Cut` were considered in the literature. Karger *et al.* [21] and Cunningham and Teng [10] present a tight approximation factor of $12/11$ when $k = 3$. For $k = 4, 5$, Karger *et al.* [21] provide approximation factors of 1.1539 and 1.2161, respectively. For dense unweighted graphs, Arora *et al.* [2] and Frieze and Kannan [16] provide a polynomial time approximation scheme. The `Node Multiway Cut` problem asks for the least weight subset of vertices whose removal from the graph disconnects all terminals. This variant was studied by Garg *et al.* [18] who present a $2(1 - 1/k)$-approximation algorithm for the problem. They also prove that any improvement to the latter factor would also lead to an improvement of the approximation guarantee for `Vertex Cover`, for which it is known that no approximation better than 2 can be achieved assuming the unique games conjecture (Khot and Regev [24]). The `Directed Multiway Cut` problem asks for the least weight subset of edges whose removal from the graph disconnects all directed paths connecting terminals. Clearly, `Directed Multiway Cut` generalizes `Node Multiway Cut`. For `Directed Multiway Cut`, Naor and Zosin [29] give a 2-approximation algorithm, improving upon the $O(\log k)$-approximation of Garg *et al.* [18]. The `Multicut` problem resembles `Multiway Cut`, however its goal is to separate k pairs of terminals $\{s_i, t_i\}$. The best known approximation for `Multicut` is $O(\log k)$ and is given by Garg *et al.* [17].

Another notable example that is closely related to `Multiway Cut` is the `Metric Labeling` problem. For the `Metric Labeling` problem, Kleinberg and Tardos [25] provide an approximation guarantee of $O(\log k)$ by building upon the tight probabilistic approximation of metrics by tree metrics [14]. Chekuri *et al.* [8] extend the geometric relaxation of Călinescu *et al.* [7] to the more general `Metric Labeling` problem using *earthmover metrics*, thus yielding a unified treatment of the problem for various metrics. For the geometric relaxation with the earthmover metric, Karloff *et al.* [22] prove an integrality gap of $\Omega(\log k)$ for `Metric Labeling` and $\Omega(\sqrt{\log k})$ for `0-Extension`. Both integrality gaps translate to hardness results with the same values assuming the unique games conjecture [28]. Without assuming the unique games conjecture, the corresponding hardness results are $\Omega(\sqrt{\log k})$ for `Metric Labeling` [9] and $\Omega(\log^{1/4} k)$ for `0-Extension` [22].

2 Preliminaries

The input for the `Multiway Cut` problem is an edge weighted undirected graph $G = (V, E)$, $w : E \to \mathcal{R}^+$, and a set $T = \{t_1, t_2, \ldots, t_k\} \subseteq V$ of k terminals. The goal is to find a minimum weight subset of edges $X \subseteq E$ such that all terminals are disconnected in $(V, E \setminus X)$. Equivalently, the output is an assignment $g : V \to \{1, \ldots, k\}$ that has the property that $g(t_i) = i$ for every $i = 1, \ldots, k$. An edge $(u, v) \in X$ if and only if $g(u) \neq g(v)$.

2.1 Simplex Relaxation We present the Călinescu *et al.* [7] geometric relaxation for `Multiway Cut`. Consider the k-dimensional simplex $\Delta_k = \{x \in \mathcal{R}^k : x \geq 0,\ \sum_i x_i = 1\}$. Denote by $\mathbf{e}_i$ the i^{th} vertex of Δ_k, i.e., the standard basis vector which has a 1 in the i^{th} coordinate and 0 elsewhere. The relaxation embeds every vertex $u \in V$ into Δ_k and each terminal $t_i \in T$ is embedded to $\mathbf{e}_i$. Intuitively, every point in Δ_k corresponds to a distribution over the set of terminals T. For simplicity of presentation we denote by $\mathbf{u}$ the vector $u \in V$ was embedded to. It is well known that the following program can be rewritten as a linear program.

$$\begin{aligned} \min \quad & \sum_{e=(u,v)\in E} w_e \cdot \frac{1}{2} \|\mathbf{u} - \mathbf{v}\|_1 & \\ & \mathbf{u} \in \Delta_k & \forall u \in V \\ & \mathbf{t}_i = \mathbf{e}_i & \forall i = 1, \ldots, k \end{aligned}$$

2.2 The Exponential Clocks Algorithm The first main ingredient we use is the exponential clocks algorithm [5]. It receives as input the embedding of the graph into the simplex as computed by the relaxation, *i.e.*, $\{\mathbf{u}\}_{u\in V} \subseteq \Delta_k$. In the exponential clocks algorithm, i.i.d $Z_1, \ldots, Z_k \sim exp(1)$ are chosen, one for each terminal. Every vertex $\mathbf{u} \in \Delta_k$ scales each Z_i by u_i (*i.e.*, Z_i is divided by u_i), and then performs a competition between the scaled exponentials, assigning itself to the winner. For completeness, we include a full description of the algorithm here. We require the following

Algorithm 1 Exponential Clocks $\left(\{\mathbf{u}\}_{u\in V} \subseteq \Delta_k\right)$

1: choose i.i.d random variables $Z_i \sim exp(1)$ for each $i = 1, 2..., k$.
2: $\forall u \in V$ assign u to argmin $\left\{\frac{Z_i}{u_i} : i = 1, 2, ...k\right\}$.

lemma that was proved in [5].

LEMMA 2.1. *Given $\mathbf{u}, \mathbf{v} \in \Delta_k$ that differ only in coordinates i and j where $v_i = u_i + \varepsilon$ and $v_j = u_j - \varepsilon$ for some $\varepsilon \geq 0$, the probability that $\mathbf{u}$ and $\mathbf{v}$ are assigned to different terminals by the exponential clocks algorithm is at most:* $\varepsilon\,(2 - u_i - u_j)$.

We note that [5] showed that one can equivalently use the algorithm of Kleinberg and Tardos for the `Uniform`

Metric Labeling problem [25]. The reason is that both the exponential clocks and Kleinberg and Tardos algorithms satisfy Lemma 2.1.

2.3 The CKR Algorithm The second main ingredient we use is the CKR algorithm [7]. It receives as input an embedding of the graph into the k-dimensional hypercube, *i.e.*, $\{\mathbf{u}\}_{u \in V} \subseteq [0,1]^k$. We note that the relaxation computes an embedding of the graph into Δ_k, as opposed to $[0,1]^k$. However, we will invoke the CKR algorithm *after* transforming the embedding which might result in points in $[0,1]^k$.

First, a uniform random order over the terminals and a single threshold $r \sim Unif[0,1]$ are chosen. Then, one iteratively goes over the terminals in the chosen order, assigning each unassigned vertex u to the current terminal in case the value of $\mathbf{u}$ in the coordinate that corresponds to the current terminal is at least r. For completeness, we include a full description of the algorithm here.

Algorithm 2 CKR $\left(\{\mathbf{u}\}_{u \in V} \subseteq [0,1]^k\right)$

1: choose $r \sim Unif[0,1]$ and σ a uniform random permutation over $\{1,\ldots,k\}$.
2: **for** $i = 1$ to $k-1$ **do**
3: $\quad \forall u \in V$ s.t. u is unassigned and $u_{\sigma(i)} \geq r$: assign u to $\sigma(i)$.
4: assign all unassigned vertices to $\sigma(k)$.

We require the following lemma that can be easily derived from [7]. We note that in [7] it was not stated explicitly in the form we require. For completeness we include a full proof.

LEMMA 2.2. *Given $\mathbf{u}, \mathbf{v} \in [0,1]^k$, where $u_1 \geq \ldots \geq u_k$, and $v_1 \geq \ldots \geq v_k$, the probability that $\mathbf{u}$ and $\mathbf{v}$ are assigned to different terminals by the CKR algorithm is at most: $\sum_{\ell=1}^{k} \frac{|u_\ell - v_\ell|}{\ell}$.*

Proof. We define for every $i = 1,\ldots,k$ the event A_i in which terminal t_i has cut the edge (u,v), *i.e.*, terminal t_i was the first to choose at least one vertex from $\{u,v\}$ and it chose *exactly* one vertex from $\{u,v\}$. Thus, the probability that (u,v) is cut equals $\Pr\left[\cup_{i=1}^{k} A_i\right]$. Assume without loss of generality that $u_i < v_i$, and note that:

$$\begin{aligned} &\Pr[A_i] \\ &\leq \Pr[r \in [u_i, v_i] \wedge t_i \text{ appears before } \{t_1,\ldots,t_{i-1}\} \text{ in } \sigma] \\ &= |u_i - v_i| \cdot \frac{1}{i} \end{aligned}$$

Note that the above inequality uses the fact that the coordinates of both $\mathbf{u}$ and $\mathbf{v}$ are sorted in the same non-increasing order, since it implies that terminal t_i can cut (u,v) only if it appears before terminals $t_1, t_2, \ldots, t_{i-1}$ in σ. Using the union bound, we can conclude that:

$$\Pr\left[\cup_{\ell=1}^{k} A_\ell\right] \leq \sum_{\ell=1}^{k} \left(|u_\ell - v_\ell| \cdot \frac{1}{\ell}\right)$$

2.4 Edge Structure and Cut Density For simplicity of presentation, and without loss of generality, we assume a specific structure of edges' embedding in simplex. However, unlike previous works [5, 7, 21, 30], we require a more restrictive structure that is described in the following observation.[1]

OBSERVATION 2.1. *Without loss of generality, one can assume that every edge $(u,v) \in E$ is of the form appearing in Figure 1, for some $1 \leq i < j \leq k$ and $\varepsilon \geq 0$, where (1) $u_1 \geq \ldots \geq u_k$, and (2) $u_{i-1} \geq u_i + \varepsilon$ and $u_j - \varepsilon \geq u_{j+1}$. We call such an edge an (i,j)-edge.*

Intuitively, the above observation states that given $(u,v) \in E$, there exists an order of the coordinates such that both $\mathbf{u}$ and $\mathbf{v}$ are non-increasing with respect to this order, and that u_i is increased by an additive ε and u_j is decreased by an additive ε, for some $i < j$ and $\varepsilon \geq 0$.

Given an edge $(u,v) \in E$, Observation 2.1 implies that its contribution to the relaxation's objective is $\frac{1}{2}\left|\left|\mathbf{u} - \mathbf{v}\right|\right|_1 = \varepsilon$. Thus, we would like any randomized algorithm to separate such an edge with probability of at most $\gamma\varepsilon$, for a small γ. Formally, to this end Karger *et al.* [21] introduced the notion of *cut density*, which we restate now.

DEFINITION 2.1. *A randomized algorithm is a distribution over labelings $g : \Delta_k \to \{1,\ldots,k\}$. Given any edge (u,v) that satisfies Observation 2.1 for some fixed $1 \leq i < j \leq k$, we say that (u,v) is of type (i,j). The cut density of edges of type (i,j) is at most $\gamma_{i,j}$ if for every edge (u,v) of type (i,j) satisfying Observation 2.1 the following holds:*

$$\limsup_{\varepsilon \to 0} \frac{\Pr[g(\mathbf{u}) \neq g(\mathbf{v})]}{\varepsilon} \leq \gamma_{i,j}\ .$$

As in Karger *et al.* [21], it is easy to show that any randomized algorithm for Multiway Cut achieves an approximation of $\gamma = \max_{1 \leq i < j \leq k}\{\gamma_{i,j}\}$. Hence, from this point onwards our goal will be to bound the cut density of edges of type (i,j), for any $1 \leq i < j \leq k$. We denote by $\alpha_{i,j}$ and $\beta_{i,j}$ the cut density of edges of type (i,j) in the CKR and exponential clocks algorithms, respectively.

Paper Organization: In Section 3 we give a brief overview of our new approach of transforming the simplex while allowing dependencies between the different variables. Section 4 serves as a case study, presenting a remarkably simple

[1] Though Observation 2.1 is more restrictive than assumptions on the structure of an edge made in previous work, its proof is virtually the same.

$$\begin{array}{l} \mathbf{u} = (u_1, \ldots, u_{i-1}, \quad u_i \quad, u_{i+1}, \ldots, u_{j-1}, \quad u_j \quad, u_{j+1}, \ldots, u_k) \in \Delta_k \\ \mathbf{v} = (u_1, \ldots, u_{i-1}, \quad u_i + \varepsilon \quad, u_{i+1}, \ldots, u_{j-1}, \quad u_j - \varepsilon \quad, u_{j+1}, \ldots, u_k) \in \Delta_k \end{array}$$

Figure 1: The structure of an (i, j)-edge.

algorithm that achieves an approximation guarantee of $^{17}/_{13}$. Finally, Section 5 contains an improved provable approximation of $^{297}/_{229} \approx 1.29694$.

3 Simplex Partitioning via General Transformations

3.1 Transformations with Variables Dependencies We start with some intuition on how one can construct useful simplex transformations f. Assume, for example, we are given two points $\mathbf{u}, \mathbf{v} \in \Delta_k$ where $\mathbf{v}$ is obtained from $\mathbf{u}$ by permuting its coordinates. Intuitively, since the `Multiway Cut` problem is symmetric, *i.e.*, there is no preference between the terminals, we would expect that $f(\mathbf{v})$ could be obtained from $f(\mathbf{u})$ by permuting its coordinates in exactly the same manner as obtaining $\mathbf{v}$ from $\mathbf{u}$. Thus, we use transformations f that first *sort* the coordinates, then transform each coordinate according to its position in the sorting, and finally place back the transformed coordinates in their original order. It is important to note that for such a transformation f to be well defined, it must be the case that the outcome of the transformation $f(\mathbf{u})$ does not depend on how ties in the sorting are resolved. Indeed, all the transformations we use in this work satisfy this. Formally, in order for a transformation to be useful we require it satisfies the following definition.

DEFINITION 3.1. *$f : \Delta_k \to [0,1]^k$ is* feasible *if the following two conditions hold:*

1. *$f(\mathbf{e}_i) = \mathbf{e}_i$, for every $i = 1, \ldots, k$.*
2. *For any $\mathbf{u} \in \Delta_k$, satisfying $u_{\pi(1)} \geq \ldots \geq u_{\pi(k)}$ for some permutation π on $\{1, \ldots, k\}$: $f_{\pi(1)}(\mathbf{u}) \geq \ldots \geq f_{\pi(k)}(\mathbf{u})$.*

The first property above implies that terminals (or equivalently vertices of Δ_k) are fixed points of f. It is required since each terminal t_i must be assigned to itself, and we will apply f prior to executing the CKR algorithm. The second property above states that f does not change the order of coordinates, a useful property in the analysis. For simplicity of presentation, from this point onwards we assume that given $\mathbf{u} \in \Delta_k$, its coordinates are already sorted in a non-increasing order, *i.e.*, $u_1 \geq \ldots \geq u_k$. Hence, Definition 3.1 reduces to: (1) $f(\mathbf{e}_1) = \mathbf{e}_1$, and (2) $f_1(\mathbf{u}) \geq \ldots \geq f_k(\mathbf{u})$.

3.2 The Algorithm Formally, our mixed algorithm receives the embedding of the graph into the simplex $\{\mathbf{u}\}_{u \in V} \subseteq \Delta_k$ as computed by the relaxation, a transformation $f : \Delta_k \to [0,1]^k$, and a mix probability p. Similarly to the framework of [5], our algorithm is just a simple mix of the two main ingredients, *i.e.*, with probability p we choose and exponential clocks algorithm applied to the simplex and with the remaining $1-p$ probability we choose the CKR algorithm that is executed on the *transformed* simplex. A full description of the algorithm appears in Algorithm 3.

Algorithm 3 $\left(\{\mathbf{u}\}_{u \in V} \subseteq \Delta_k, f, p\right)$

1: w.p. p execute Algorithm 1 with input $\{\mathbf{u}\}_{u \in V}$.
2: w.p. $1-p$ execute Algorithm 2 with input $\{f(\mathbf{u})\}_{u \in V}$.

3.3 Choosing the function f Our goal is to upper bound the cut density of Algorithm 3, which is achieved by the following lemma.

LEMMA 3.1. *If f is feasible and differentiable, then the cut density of Algorithm 3 for any edge of type (i, j) satisfying Observation 2.1 can be upper bounded as follows:*

$$\begin{aligned} & p\beta_{i,j} + (1-p)\alpha_{i,j} \\ & \leq p(2 - u_i - u_j) + (1-p) \sum_{\ell=1}^{k} \frac{1}{\ell} \left| \frac{\partial f_\ell(\mathbf{u})}{\partial u_i} - \frac{\partial f_\ell(\mathbf{u})}{\partial u_j} \right| . \end{aligned}$$

Proof. Lemma 2.1 implies that:

$$\beta_{i,j} \leq 2 - u_i - u_j .$$

Since we assume all edges of type (i, j) satisfy Observation 2.1 and that f is feasible and differentiable, all conditions of Lemma 2.2 are satisfied and we can conclude:

$$\alpha_{i,j} \leq \sum_{\ell=1}^{k} \frac{1}{\ell} \left| \frac{\partial f_\ell(\mathbf{u})}{\partial u_i} - \frac{\partial f_\ell(\mathbf{u})}{\partial u_j} \right| .$$

The definition of Algorithm 3 concludes the proof.

We strive for a constant cut density, therefore it is reasonable to choose each f_ℓ to be a multivariate polynomial in the variables $u_1, \ldots, u_k$ of degree 2. Hence, when operating in this manner, the linear contributions of the variables $u_1, \ldots, u_k$ in the overall cut density might be balanced. The main challenge with this approach is to choose such an f while ensuring the f is feasible and differentiable.

4 Beyond the $\frac{3+\sqrt{5}}{4}$ Lower Bound via a Simplex Transformation

In this section we present a remarkably simple algorithm for `Multiway Cut` that achieves an approximation of $^{17}/_{13}$. As mentioned in Section 3, our transformation $f : \Delta_k \to [0,1]^k$, given $\mathbf{u} \in \Delta_k$, first sorts the coordinates of $\mathbf{u}$ in a non-increasing order and than applies a transformation to each of the coordinates that depends on its position in the sorting. Without loss of generality, rename the coordinates of $\mathbf{u}$ according to this order: $u_1 \geq \ldots \geq u_k$. The transformation we use is the following:

$$f_\ell(\mathbf{u}) = \left(1 + \frac{\ell - 1}{3}\right) u_\ell^2 + \frac{1}{3} u_\ell \sum_{s=\ell+1}^{k} u_s \quad \forall \ell = 1, \ldots, k.$$

First, it is important to note that $f(\mathbf{u})$ does not depend on how ties in the sorting are resolved. Hence, f is well defined. Second, it is easy to verify that f satisfies Definition 3.1 and that f is also differentiable. We are now ready to prove Theorem 1.1.

Proof. [Proof (of Theorem 1.1)] Consider an edge $(u,v) \in E$ of type (i,j) satisfying Observation 2.1. One can easily verify that the following hold:

$$\begin{aligned}
\frac{\partial f_\ell(\mathbf{u})}{\partial u_i} - \frac{\partial f_\ell(\mathbf{u})}{\partial u_j} &= 0 \quad \forall \ell = 1, \ldots, i-1 \\
\frac{\partial f_i(\mathbf{u})}{\partial u_i} - \frac{\partial f_i(\mathbf{u})}{\partial u_j} &= \left[u_i \left(1 + \frac{2i}{3}\right) + \frac{1}{3} \sum_{\ell=i+1}^{k} u_\ell\right] \\
\frac{\partial f_\ell(\mathbf{u})}{\partial u_i} - \frac{\partial f_\ell(\mathbf{u})}{\partial u_j} &= -\frac{1}{3} u_\ell \quad \forall \ell = i+1, \ldots, j-1 \\
\frac{\partial f_j(\mathbf{u})}{\partial u_i} - \frac{\partial f_j(\mathbf{u})}{\partial u_j} &= -\left[u_j \left(\frac{4}{3} + \frac{2j}{3}\right) + \frac{1}{3} \sum_{\ell=j+1}^{k} u_\ell\right] \\
\frac{\partial f_\ell(\mathbf{u})}{\partial u_i} - \frac{\partial f_\ell(\mathbf{u})}{\partial u_j} &= 0 \quad \forall \ell = j+1, \ldots, k
\end{aligned}$$

Note that:

$$\begin{aligned}
&\sum_{\ell=1}^{k} \frac{1}{\ell} \left| \frac{\partial f_\ell(\mathbf{u})}{\partial u_i} - \frac{\partial f_\ell(\mathbf{u})}{\partial u_j} \right| \\
&\quad = \left[u_i \left(\frac{2}{3} + \frac{1}{i}\right) + \sum_{\ell=i+1}^{j-1} u_\ell \left(\frac{1}{3i} + \frac{1}{3\ell}\right) \right. \\
&\quad \left. + \, u_j \left(\frac{2}{3} + \frac{1}{3i} + \frac{4}{3j}\right) + \sum_{\ell=j+1}^{k} u_\ell \left(\frac{1}{3i} + \frac{1}{3j}\right) \right].
\end{aligned}$$

Since f is feasible and differentiable, we can apply Lemma 3.1 with $p = {}^7/_{13}$ and conclude:

$$\begin{aligned}
&(1-p)\alpha_{i,j} + p\beta_{i,j} \\
&\leq \frac{14}{13} + u_i \left(-\frac{3}{13} + \frac{6}{13i}\right) + \sum_{\ell=i+1}^{j-1} u_\ell \frac{2}{13} \left(\frac{1}{i} + \frac{1}{\ell}\right) \\
&\quad + u_j \left(-\frac{3}{13} + \frac{2}{13i} + \frac{8}{13j}\right) + \sum_{\ell=j+1}^{k} u_\ell \frac{2}{13} \left(\frac{1}{i} + \frac{1}{j}\right) \\
&\overset{\text{(i)}}{\leq} \frac{14}{13} + \frac{3}{13} (u_i + \ldots + u_k) \overset{\text{(ii)}}{\leq} \frac{17}{13}
\end{aligned}$$

Inequality (i) follows from the choice of the worst possible i and j (*i.e.*, $i = 1$ and $j = 2$), whereas inequality (ii) follows from the fact that $\mathbf{u} \in \Delta_k$.

5 Choosing a Simplex Transformation

In this section we present a general framework for obtaining a transformation f that is feasible, *i.e.*, satisfies Definition 3.1, and differentiable. We then design a $^{297}/_{229} \approx 1.2969$-approximation for `Multiway Cut` proving our main theorem. As before, for any vector $\mathbf{u} \in \mathcal{R}^k$, we assume without loss of generality that $\mathbf{u}$'s coordinates are sorted in a non-increasing order, *i.e.*, $u_1 \geq u_2 \geq \ldots \geq u_k$. For such a vector $\mathbf{u}$, we denote by $\mathbf{u}^\ell \in \mathcal{R}^k$ the vector obtained from $\mathbf{u}$ by replacing the first $\ell - 1$ coordinates with the value u_ℓ. Thus,

$$\mathbf{u}^\ell = (\underbrace{u_\ell, \ldots, u_\ell}_{\ell \text{ times}}, u_{\ell+1}, u_{\ell+2}, \ldots, u_k).$$

We choose the transformation f to be a homogenous multivariate polynomial in $u_1, \ldots, u_k$ of degree 2, *i.e.*, a quadratic form. For every $\ell = 1, \ldots, k$ we define $f_\ell : \Delta_k \to [0,1]$ as follows:

$$f_\ell(\mathbf{u}) \triangleq \left(\mathbf{u}^\ell\right)^T A \mathbf{u}^\ell, \tag{5.1}$$

where $A \in \mathcal{R}^{k \times k}$ is a symmetric matrix. The transformation f is defined as in Section 4: $f(\mathbf{u}) \triangleq (f_1(\mathbf{u}), f_2(\mathbf{u}), \ldots, f_k(\mathbf{u}))$.

First, it is important to note that f is well defined since $f(\mathbf{u})$ does *not* depend on how ties in the sorting are resolved, and that f is also differentiable. Second, we provide sufficient conditions on the matrix A such that f is feasible, *i.e.*, satisfies all the conditions of Definition 3.1. The following lemma summarizes these conditions.

LEMMA 5.1. *Let $A \in \mathcal{R}^{k \times k}$ be a symmetric matrix and $f : \Delta_k \to [0,1]^k$ where $f_\ell = \left(\mathbf{u}^\ell\right)^T A \mathbf{u}^\ell$ for every $\ell = 1, \ldots, k$. If A satisfies the following conditions:*

1. *$a_{11} = 1$ and $a_{ij} \leq 1$ for every $i, j = 1, \ldots, k$.*
2. *$\sum_{i=1}^{\ell} \sum_{j=1}^{\ell} a_{ij} \geq 0$ for every $\ell = 1, \ldots, k$.*

3. $\sum_{i=1}^{\ell}\sum_{j=\ell+1}^{r} a_{ij} \geq 0$ *for every* $\ell = 1,\ldots,k$ *and* $r = \ell+1,\ldots,k$.

then f is feasible.

Proof. It is easy to verify that if $a_{11} = 1$ then $f(\mathbf{e}_1) = \mathbf{e}_1$. Next, we prove that for each $\ell = 1,\ldots,k-1$: $f_\ell(\mathbf{u}) \geq f_{\ell+1}(\mathbf{u})$. Note that:

$$f_\ell(\mathbf{u}) - f_{\ell+1}(\mathbf{u}) = \left(\mathbf{u}^\ell\right)^T A\mathbf{u}^\ell - \left(\mathbf{u}^{\ell+1}\right)^T A\mathbf{u}^{\ell+1}$$
$$= \left(u_\ell^2 - u_{\ell+1}^2\right)\sum_{i=1}^{\ell}\sum_{j=1}^{\ell} a_{ij} + 2\left(u_\ell - u_{\ell+1}\right)\sum_{i=1}^{\ell}\sum_{j=\ell+1}^{k} a_{ij}u_j$$
$$= \left(u_\ell^2 - u_{\ell+1}^2\right)\left(\sum_{i=1}^{\ell}\sum_{j=1}^{\ell} a_{ij}\right)$$
$$+ 2\left(u_\ell - u_{\ell+1}\right)\sum_{r=\ell+1}^{k}\left(u_r - u_{r+1}\right)\left(\sum_{i=1}^{\ell}\sum_{j=\ell+1}^{r} a_{ij}\right) \overset{(*)}{\geq} 0$$

Inequality $(*)$ follows since the coordinates of $\mathbf{u}$ are sorted, *i.e.*, $u_\ell - u_{\ell+1} \geq 0$, and by properties (2) and (3). By the same arguments $f_k(\mathbf{u}) = u_k^2\left(\sum_{i=1}^{k}\sum_{j=1}^{k} a_{ij}\right) \geq 0$ and hence $f_\ell(\mathbf{u}) \geq 0$ for every $\ell = 1,\ldots,k$. Finally, by property (1) we get that:

$$f_1(\mathbf{u}) = \left(\mathbf{u}^1\right)^T A\mathbf{u}^1 = \sum_{i=1}^{k}\sum_{j=1}^{k} a_{ij}u_iu_j \leq \sum_{i=1}^{k}\sum_{j=1}^{k} u_iu_j = 1$$

Since $f_\ell(\mathbf{u}) \geq f_{\ell+1}(\mathbf{u})$ for every $\ell = 1,\ldots,k-1$, we get that $f_\ell(\mathbf{u}) \leq 1$ for evry $\ell = 1,\ldots,k$.

Denote by $\mathbf{a}_i$ the i^{th} column (or equivalently row) of A. Given a matrix A that satisfies the requirements of Lemma 5.1, the following lemma upper bounds the cut density $\alpha_{i,j}$ of Algorithm 2 when executed *after* the application of the transformation f.

LEMMA 5.2. *If $A \in \mathcal{R}^{k\times k}$ satisfies the conditions of Lemma 5.1, then:*

$$\alpha_{i,j} \leq \sum_{\ell=1}^{i-1}\frac{2}{\ell}\left|(\mathbf{a}_i - \mathbf{a}_j)\cdot\mathbf{u}^\ell\right| + \frac{2}{i}\left|\left(\sum_{\ell=1}^{i}\mathbf{a}_\ell - \mathbf{a}_j\right)\cdot\mathbf{u}^i\right|$$
$$+ \sum_{\ell=i+1}^{j-1}\frac{2}{\ell}\left|\mathbf{a}_j\cdot\mathbf{u}^\ell\right| + \frac{2}{j}\left|\left(\sum_{\ell=1}^{j}\mathbf{a}_\ell\right)\cdot\mathbf{u}^j\right| \quad (5.2)$$

Proof. Since A satisfies the conditions of Lemma 5.1, we can derive that f is differentiable and feasible. In particular, $f_1(\mathbf{u}) \geq \ldots \geq f_k(\mathbf{u})$. Hence, for all small enough ε the edge (u,v) satisfies Observation 2.1 *after* applying the transformation f. Therefore, all the requirements of Lemma 2.2 are satisfied and we can conclude that:

$$\alpha_{i,j} \overset{\text{(i)}}{\leq} \sum_{\ell=1}^{k}\frac{1}{\ell}\left|\frac{\partial f_\ell(\mathbf{u})}{\partial u_i} - \frac{\partial f_\ell(\mathbf{u})}{\partial u_j}\right|$$
$$\overset{\text{(ii)}}{=} \sum_{\ell=1}^{i-1}\frac{2}{\ell}\left|(\mathbf{a}_i - \mathbf{a}_j)\cdot\mathbf{u}^\ell\right| + \frac{2}{i}\left|\left(\sum_{\ell=1}^{i}\mathbf{a}_\ell - \mathbf{a}_j\right)\cdot\mathbf{u}^i\right|$$
$$+ \sum_{\ell=i+1}^{j-1}\frac{2}{\ell}\left|\mathbf{a}_j\cdot\mathbf{u}^\ell\right| + \frac{2}{j}\left|\left(\sum_{\ell=1}^{j}\mathbf{a}_\ell\right)\cdot\mathbf{u}^j\right|$$

Inequality (i) follows from Lemma 2.2 and the definition of the cut density $\alpha_{i,j}$. Equality (ii) follows from the definition of $\mathbf{u}_\ell$ and the observation that for any vector $\mathbf{x}$: $\frac{\partial\left(\mathbf{x}^TA\mathbf{x}\right)}{\partial x_i} = 2\left(\mathbf{a}_i\cdot\mathbf{x}\right)$ (as A is symmetric).

Analyzing the bound given by Lemma 5.2 is not straightforward due to the absolute values. Hence, to be able to remove the absolute values in (5.2) we introduce the following partial order over vectors.

DEFINITION 5.1. *For $\mathbf{a},\mathbf{b} \in \mathcal{R}^k$, $\mathbf{a} \succeq \mathbf{b}$ if for all $\ell = 1,2,\ldots,k$: $\sum_{j=1}^{\ell} a_j \geq \sum_{j=1}^{\ell} b_j$.*

The following lemma allows us to remove the absolute values in (5.2), and it plays an important role in the proof of Lemma 5.4.

LEMMA 5.3. *Let $\mathbf{a} \in \mathcal{R}^k$, the following two are equivalent:*

1. $\mathbf{a}\cdot\mathbf{u} \geq 0$ *for every* $\mathbf{u} \in \mathcal{R}_+^k$ *s.t.* $u_1 \geq \ldots \geq u_k \geq 0$.

2. $\mathbf{a} \succeq 0$.

Proof. Note that $\mathbf{a}\cdot\mathbf{u} = \sum_{j=1}^{k}\left(u_j - u_{j+1}\right)\left(\sum_{i=1}^{j} a_i\right)$, where $u_{k+1} = 0$. If for all $j = 1,\ldots,k$ we have that $\sum_{i=1}^{j} a_i \geq 0$, *i.e.*, $\mathbf{a} \succeq 0$, then $\mathbf{a}\cdot\mathbf{u}$ is certainly non-negative. Otherwise, if for some j, $\sum_{i=1}^{j} a_i < 0$, then $\mathbf{a}\cdot\mathbf{u}$ is negative for a vector $\mathbf{u}$ whose first j coordinates are $1/j$ and the rest of the coordinates are zero.

5.1 A $297/229$-Approximation for the Multiway Cut Problem In this section we prove our main theorem and present an algorithm that has a provable approximation guarantee of $297/229 \approx 1.2969$ for `Multiway Cut`. Our algorithm is simply Algorithm 3 with $p = 121/229$ and a transformation f as defined by (5.1) with the following matrix A:

$$
(5.3) \qquad A = \begin{pmatrix}
1 & \frac{5}{27} & \frac{5}{27} & \frac{5}{27} & \frac{5}{27} & \dots & \frac{5}{27} \\
\frac{5}{27} & \frac{1}{9} & \frac{5}{108} & \frac{5}{108} & \frac{5}{108} & \dots & \frac{5}{108} \\
\frac{5}{27} & \frac{5}{108} & \frac{1}{9} & \frac{5}{108} & \frac{5}{108} & \dots & \frac{5}{108} \\
\frac{5}{27} & \frac{5}{108} & \frac{5}{108} & -\frac{5}{18} & 0 & \dots & 0 \\
\frac{5}{27} & \frac{5}{108} & \frac{5}{108} & 0 & 0 & \dots & 0 \\
 & \vdots & & & & & \\
\frac{5}{27} & \frac{5}{108} & \frac{5}{108} & 0 & 0 & \dots & 0
\end{pmatrix}
$$

Note that the (symmetric) matrix A has non-zero entries only in the first 3 rows (and columns) and in coordinate a_{44}. In the following we summarize useful properties of the matrix A that we use later on in the proof.

OBSERVATION 5.1. *A satisfies the conditions in Lemma 5.1 and:*

1. $\mathbf{a}_j = \mathbf{a}_5$ *for all* $j \geq 5$.
2. $\mathbf{a}_1 \succeq \mathbf{a}_2 \succeq \mathbf{a}_3 \succeq \mathbf{a}_5 \succeq \mathbf{a}_4 \succeq 0$. *(Note that* $\mathbf{a}_5 \succeq \mathbf{a}_4$*).*

It is easy to verify that the above A satisfies all the conditions of Lemma 5.1. The following is our main technical lemma bounding the cut density of Algorithm 2 when executed after applying the transformation defined by the above matrix A.

LEMMA 5.4. *The cut density of Algorithm 2 executed after applying the transformation defined by the matrix* (5.3) *satisfies:*

$$
\alpha_{i,j} \leq \frac{44}{27}(u_i + u_j) + \frac{55}{108}\sum_{\ell \neq i,j} u_\ell
$$

Using Lemma 5.4 the proof of Theorem 1.2 follows easily.

Proof. [Proof (of Theorem 1.2)] We execute Algorithm 3 with $p = {}^{121}/_{229}$ and a transformation defined by the matrix A as in (5.3). Lemmas 2.1 and 5.4 imply that for edges of type (i,j):

$$
\begin{aligned}
&(1-p)\alpha_{i,j} + p\beta_{i,j} \\
\leq & 2p + (1-p)\frac{55}{108}\sum_{\ell \neq i,j} u_\ell + (u_i + u_j)\left((1-p)\frac{44}{27} - p\right) \\
= & \frac{242}{229} + \frac{55}{229}\sum_{\ell=1}^{k} u_\ell = \frac{297}{229}
\end{aligned}
$$

The last equality follows since $\mathbf{u} \in \Delta_k$.

We conclude by proving Lemma 5.4.

Proof. [Proof of Lemma 5.4] By Lemma 5.2 the cut density of Algorithm 2 with transformation defined as in Equation (5.1) using the matrix A is,

$$
\begin{aligned}
\alpha_{i,j} \leq & \sum_{\ell=1}^{i-1}\frac{2}{\ell}\left|(\mathbf{a}_i - \mathbf{a}_j)\cdot \mathbf{u}^\ell\right| + \frac{2}{i}\left|\left(\sum_{\ell=1}^{i}\mathbf{a}_\ell - \mathbf{a}_j\right)\cdot \mathbf{u}^i\right| \\
& + \sum_{\ell=i+1}^{j-1}\frac{2}{\ell}\left|\mathbf{a}_j \cdot \mathbf{u}^\ell\right| + \frac{2}{j}\left|\left(\sum_{\ell=1}^{j}\mathbf{a}_\ell\right)\cdot \mathbf{u}^j\right|
\end{aligned}
$$

By the Observation 5.1 and Lemma 5.3 we can remove the absolute values in the following direction:

$$
\begin{aligned}
\alpha_{i,j} \leq & \sum_{\ell=1}^{i-1}\frac{2}{\ell}(\mathbf{a}_i - \mathbf{a}_j)\cdot \mathbf{u}^\ell + \frac{2}{i}\left(\sum_{\ell=1}^{i}\mathbf{a}_\ell - \mathbf{a}_j\right)\cdot \mathbf{u}^i \\
& + \sum_{\ell=i+1}^{j-1}\frac{2}{\ell}\mathbf{a}_j \cdot \mathbf{u}^\ell + \frac{2}{j}\left(\sum_{\ell=1}^{j}\mathbf{a}_\ell\right)\cdot \mathbf{u}^j \quad i \neq 4 \\
\alpha_{4,j} \leq & \sum_{\ell=1}^{i-1}\frac{2}{\ell}(\mathbf{a}_5 - \mathbf{a}_4)\mathbf{u}^\ell + \frac{2}{4}\left(\sum_{\ell=1}^{4}\mathbf{a}_\ell - \mathbf{a}_5\right)\cdot \mathbf{u}^4 \\
& + \sum_{\ell=5}^{j-1}\frac{2}{\ell}\mathbf{a}_5 \cdot \mathbf{u}^\ell + \frac{2}{j}\left(\sum_{\ell=1}^{j}\mathbf{a}_\ell\right)\cdot \mathbf{u}^j \quad i = 4
\end{aligned}
$$

For simplicity of presentation, we introduce some useful notations. Let H_i be the i^{th} harmonic number. For all $j \geq 5$, let $T_5 = \sum_{i=1}^{3} a_{i5} = \sum_{i=1}^{3} a_{ij} = {}^5/_{18}$. Additionally, $S_\ell = \sum_{i=1}^{\ell}\sum_{j=1}^{\ell} a_{ij}$. In particular,

$$
S_1 = 1, \quad S_2 = \tfrac{40}{27}, \quad S_3 = \tfrac{37}{18}, \quad S_4 = \tfrac{7}{3},
$$

$S_\ell = S_4 + 2(\ell - 4)T_5 \quad \forall \ell \geq 4$. Therefore, for all $\ell \geq 4$

$$
(5.4) \qquad \frac{1}{\ell}S_\ell = \frac{1}{\ell}(S_4 - 8T_5) + 2T_5 = \frac{1}{9\ell} + \frac{5}{9} \leq \frac{7}{12}
$$

We start by analyzing the case $i \neq 4$. Rearranging the bound on $\alpha_{i,j}$ and using the above notation we get the following equivalent form:

(5.5)
$$
\alpha_{i,j} \leq 2\sum_{r=1}^{i-1} u_r\left[(a_{ir} - a_{jr})H_{r-1} + \frac{2}{r}\sum_{s=1}^{r}(a_{is} - a_{js})\right]
$$

(5.6)
$$
+ 2u_i\left[(a_{ii} - a_{ji})H_{i-1} + \frac{1}{i}\left(S_i - \sum_{s=1}^{i} a_{js}\right)\right]
$$

(5.7)
$$
\begin{aligned}
& + 2\sum_{r=i+1}^{j-1} u_r\left[(a_{ir} - a_{jr})H_{i-1} + \frac{1}{i}\left(-a_{jr} + \sum_{\ell=1}^{i} a_{\ell r}\right)\right. \\
& \left. + a_{jr}\sum_{s=i+1}^{r-1}\frac{1}{s} + \frac{1}{r}\sum_{s=1}^{r} a_{js}\right]
\end{aligned}
$$

(5.8)

$$+ 2u_j \left[(a_{ij} - a_{jj})\, H_{i-1} + \frac{1}{i} \left(-a_{jj} + \sum_{\ell=1}^{i} \mathbf{a}_{\ell j} \right) + a_{jj} \sum_{s=i+1}^{j-1} \frac{1}{s} + \frac{1}{j} S_j \right]$$

(5.9)

$$+ 2 \sum_{r=j+1}^{k} u_r \left[(a_{ir} - a_{jr})\, H_{i-1} + \frac{1}{i} \left(-a_{jr} + \sum_{\ell=1}^{i} a_{\ell r} \right) + a_{jr} \sum_{s=i+1}^{j-1} \frac{1}{s} + \frac{1}{j} \sum_{\ell=1}^{j} a_{\ell r} \right]$$

Our proof follows by a case analysis of the coefficients of the $u_1, \ldots, u_k$ variables in the above sum (for every i, j) proving that they all satisfy the guarantees of Lemma 5.4. In the course of the analysis we will use inequality (5.4).

Analysis of (5.5)**:** This sum is always zero for all i, j. since for any $r < i$, $a_{ir} = a_{jr}$.

Analysis of (5.6)**:** We analyze the required cases of the value of i.

$\mathbf{i = 1}$: For this case the term equals :

$$2(a_{11} - a_{1j}) = 2(1 - \frac{5}{27}) = \frac{44}{27}$$

$\mathbf{i = 2}$: For this case the term equals :

$$2 \left[(a_{22} - a_{2j}) + \frac{1}{2} (S_2 - a_{21} - a_{22}) \right] = \frac{71}{54} < \frac{44}{27}$$

$\mathbf{i = 3}$: For this case the term equals :

$$2 \left[\frac{3}{2} (a_{33} - a_{3j}) + \frac{1}{3} (S_3 - a_{31} - a_{32} - a_{33}) \right] = \frac{149}{108} < \frac{44}{27}$$

Finally, for $i \geq 5$ the term equals :

$$\frac{2}{i} [S_i - T_5] \leq 2\frac{S_i}{i} \leq \frac{7}{6} < \frac{44}{27}$$

Analysis of (5.7)**:** (note that $j > r > i$) We analyze the required cases of the value of i.

$\mathbf{i = 1}$: For this case there are several subcases. When $r = 2$ the term equals:

$$2 \left[(-a_{j2} + a_{12}) + \frac{1}{2} (a_{j1} + a_{j2}) \right] = \frac{55}{108}$$

When $r = 3$ the term equals:

$$2 \left[(-a_{j3} + a_{13}) + \frac{1}{2} a_{j3} + \frac{1}{3} (a_{j1} + a_{j2} + a_{j3}) \right] = \frac{55}{108}$$

Finally, when $r \geq 4$ the term equals:

$$2 \left[a_{14} + \frac{1}{r} (a_{j1} + a_{j2} + a_{j3}) \right] = \frac{10}{27} + \frac{5}{9r} \leq \frac{55}{108}$$

$\mathbf{i = 2}$: Again, for this case there are several subcases to check.
When $r = 3$ the term equals

$$2 \left[(a_{23} - a_{j3}) + \frac{1}{2} (-a_{j3} + a_{13} + a_{23}) + \frac{1}{3} \sum_{s=1}^{3} a_{js} \right] = \frac{10}{27} < \frac{55}{108}$$

For $r \geq 4$ the term equals:

$$2 \left[a_{24} + \frac{1}{2} (a_{14} + a_{24}) + \frac{1}{r} \sum_{s=1}^{3} a_{js} \right] \leq \frac{25}{54} < \frac{55}{108}$$

$\mathbf{i = 3}$: In this case $r \geq 4$ and the term equals:

$$2 \left[\frac{3}{2} a_{34} + \frac{1}{3} (a_{14} + a_{24} + a_{34}) + \frac{1}{r} \sum_{s=1}^{3} a_{js} \right] \leq \frac{50}{108} < \frac{55}{108}$$

Finally, when $i \geq 5$ (and $r > i$) the term equals:

$$2 \left[\frac{1}{i} (a_{1r} + a_{2r} + a_{3r}) + \frac{1}{r} \sum_{s=1}^{3} a_{js} \right] \leq \frac{22}{108} < \frac{55}{108}$$

Analysis of (5.8)**:** We analyze the required cases of the value of i.

$\mathbf{i = 1}$: For this the general term equals:

$$2 \left[a_{1j} + a_{jj} \left(-1 + \sum_{s=2}^{j-1} \frac{1}{s} \right) + \frac{1}{j} S_j \right]$$

We analyze several subcases of values of j. When $j = 2$ the term equals:

$$2 \left[a_{12} - a_{22} + \frac{1}{2} S_2 \right] = \frac{44}{27}$$

When $j = 3$ the term equals:

$$2 \left[a_{13} - \frac{1}{2} a_{33} + \frac{1}{3} S_3 \right] = \frac{44}{27}$$

When $j = 4$ the term equals:

$$2\left[a_{14} - \frac{1}{6}a_{44} + \frac{1}{4}S_4\right] = \frac{44}{27}$$

Finally, when $j \geq 5$ the term equals:

$$2\left[a_{1j} + \frac{1}{j}S_j\right] \leq 2\left[\frac{5}{27} + \frac{7}{12}\right] = \frac{83}{54} < \frac{44}{27}$$

$\mathbf{i = 2}$: For this case the general term equals:

$$2\left[a_{2j} - a_{jj} + \frac{1}{2}\left(-a_{jj} + a_{1j} + a_{2j}\right) + a_{jj}\sum_{s=3}^{j-1}\frac{1}{s} + \frac{1}{j}S_j\right]$$

Again, we analyze several subcases of values of $j > i$. When $j = 3$ the term equals:

$$2\left[a_{23} - a_{33} + \frac{1}{2}\left(-a_{33} + a_{13} + a_{23}\right) + \frac{1}{3}S_3\right] = \frac{49}{36} < \frac{44}{27}$$

When $j = 4$ the term equals:

$$2\left[a_{24} - a_{44} + \frac{1}{2}\left(-a_{44} + a_{14} + a_{24}\right) + \frac{1}{3}a_{44} + \frac{1}{4}S_4\right] = \frac{77}{36} = \frac{44}{27} + \frac{55}{108}$$

Finally, when $j \geq 5$ the term equals:

$$2\left[a_{2j} + \frac{1}{2}\left(a_{1j} + a_{2j}\right) + \frac{1}{j}S_j\right] = \frac{35}{108} + 2\frac{S_j}{j} \leq \frac{35}{108} + \frac{7}{6} < \frac{44}{27}$$

$\mathbf{i = 3}$: For this case the term equals:

$$2\left[\left(a_{3j} - a_{jj}\right)\frac{3}{2} + \frac{1}{3}\left(-a_{jj} + a_{1j} + a_{2j} + a_{3j}\right) + a_{jj}\sum_{s=4}^{j-1}\frac{1}{s} + \frac{1}{j}S_j\right]$$

Again, we analyze several subcases of values of $j > i$. When $j = 4$ the term equals:

$$2\left[\left(a_{34} - a_{44}\right)\frac{3}{2} + \frac{1}{3}\left(-a_{44} + a_{14} + a_{24} + a_{34}\right) + \frac{1}{4}S_4\right] = \frac{271}{108} = \frac{44}{27} + \frac{95}{108}$$

When $j \geq 5$ the term equals:

$$2\left[\frac{3}{2}a_{3j} + \frac{1}{3}\left(a_{1j} + a_{2j} + a_{3j}\right) + \frac{1}{j}S_j\right] \leq \frac{161}{108} < \frac{44}{27}$$

Finally, when $i \geq 5$ (and $j > i$) the term equals:

$$2\left[\frac{1}{i}\left(a_{1j} + a_{2j} + a_{3j}\right) + \frac{1}{j}S_j\right] \leq 2\left(\frac{1}{5}T_5 + \frac{7}{12}\right) = \frac{23}{18} < \frac{44}{27}$$

Analysis of (5.9)**:** This case is completely symmetric with (5.7) with r and j switching roles. Note that in all cases apart from the cases $\{i = 2, j = 4\}$ and $\{i = 3, j = 4\}$,

$$\alpha_{i,j} \leq \frac{44}{27}(u_i + u_j) + \sum_{\ell \neq i,j}\frac{55}{108}u_\ell$$

as desired. For these cases we get:

$\{\mathbf{i = 2}, \mathbf{j = 4}\}$:

$$\begin{aligned}\alpha_{2,4} \leq & 0 \cdot u_1 + \frac{44}{27}u_2 + \frac{55}{108}u_3 + \left(\frac{44}{27} + \frac{55}{108}\right)u_4 + \sum_{\ell > 4}\frac{55}{108}u_\ell \\ \leq & \frac{55}{108}u_1 + \frac{44}{27}u_2 + \frac{55}{108}u_3 + \frac{44}{27}u_4 + \sum_{\ell > 4}\frac{55}{108}u_\ell\end{aligned}$$

$\{\mathbf{i = 3}, \mathbf{j = 4}\}$:

$$\begin{aligned}\alpha_{3,4} \leq & 0 \cdot u_1 + 0 \cdot u_2 + \frac{44}{27}u_3 + \left(\frac{44}{27} + \frac{110}{108}\right)u_4 + \sum_{\ell > 4}\frac{55}{108}u_\ell \\ \leq & \frac{55}{108}u_1 + \frac{55}{108}u_2 + \frac{44}{27}u_3 + \frac{44}{27}u_4 + \sum_{\ell > 4}\frac{55}{108}u_\ell\end{aligned}$$

In the above we use that $u_1 \geq u_2 \geq u_3 \geq u_4$.

Finally, we analyze the case $i = 4$ (and $j \geq 5$). Rearranging the the bound $\alpha_{4,j}$ we get:

$$\begin{aligned}\alpha_{4,j} \leq & 2\left[\left(-a_{44}\right)H_3 + \frac{1}{4}\left(S_4 - T_5\right)\right]u_4 \\ & + \sum_{r=5}^{j-1} 2\left[\frac{1}{4}T_5 + \frac{1}{j}T_5\right]u_r + 2\left[\frac{1}{4}T_5 + \frac{1}{j}S_j\right]u_j \\ & + \sum_{r=j+1}^{k} 2\left[\frac{1}{4}T_5 + \frac{1}{r}T_5\right]u_r \\ \leq & \frac{221}{108}u_4 + \sum_{r=5}^{j}\frac{1}{4}u_r + \frac{47}{36}u_j + \sum_{r=j+1}^{k}\frac{1}{4}u_r \\ \leq & \frac{44}{27}\left(u_4 + u_j\right) + \frac{55}{108}\sum_{r \neq 4,j}u_r\end{aligned}$$

The second inequality follows by our bounds on $\frac{S_j}{j}$. The final inequality follows since $\frac{221}{108}u_4$ is at most $\frac{55}{108}(u_1+u_2+u_3)+\frac{44}{27}u_4$, and that $\frac{1}{4}<\frac{55}{108}$, $\frac{47}{36}<\frac{44}{27}$.

Acknowledgments

The authors would like to thank Uri Zwick for his insightful remarks, and the anonymous referees for their helpful comments.

References

[1] Haris Angelidakis, Yury Makarychev, and Pasin Manurangsi. An improved integrality gap for the călinescu-karloff-rabani relaxation for multiway cut. *personal communication*, 2016.

[2] Sanjeev Arora, David Karger, and Marek Karpinski. Polynomial time approximation schemes for dense instances of np-hard problems. STOC '95, pages 284–293, 1995.

[3] Sanjeev Arora, Satish Rao, and Umesh Vazirani. Expander flows, geometric embeddings and graph partitioning. *J. ACM*, 56(2):5:1–5:37, 2009.

[4] Yonatan Aumann and Yuval Rabani. An $O(\log k)$ approximate min-cut max-flow theorem and approximation algorithm. *SIAM J. Comput.*, 27(1):291–301, 1998.

[5] Niv Buchbinder, Joseph (Seffi) Naor, and Roy Schwartz. Simplex partitioning via exponential clocks and the multiway cut problem. In *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 535–544, 2013.

[6] Gruia Călinescu, Howard Karloff, and Yuval Rabani. Approximation algorithms for the 0-extension problem. SODA '01, pages 8–16, 2001.

[7] Gruia Călinescu, Howard J. Karloff, and Yuval Rabani. An improved approximation algorithm for multiway cut. *J. Comput. Syst. Sci.*, 60(3):564–574, 2000.

[8] Chandra Chekuri, Sanjeev Khanna, Joseph Naor, and Leonid Zosin. A linear programming formulation and approximation algorithms for the metric labeling problem. *SIAM J. Discrete Math.*, 18(3):608–625, 2004.

[9] Julia Chuzhoy and Joseph Naor. The hardness of metric labeling. *SIAM J. Comput.*, 36(5):1376–1386, 2007.

[10] William H. Cunningham and Lawrence Tang. Optimal 3-terminal cuts and linear programming. IPCO '09, pages 114–125, 1999.

[11] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis. The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23:864–894, 1994.

[12] Guy Even, Joseph Seffi Naor, Satish Rao, and Baruch Schieber. Divide-and-conquer approximation algorithms via spreading metrics. *J. ACM*, 47(4):585–616, 2000.

[13] Jittat Fakcharoenphol, Chris Harrelson, Satish Rao, and Kunal Talwar. An improved approximation algorithm for the 0-extension problem. SODA '03, pages 257–265, 2003.

[14] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *J. Comput. Syst. Sci.*, 69(3):485–497, 2004.

[15] Ari Freund and Howard J. Karloff. A lower bound of 8/(7+(1/k-1)) on the integrality ratio of the Calinescu-Karloff-Rabani relaxation for multiway cut. *Inf. Process. Lett.*, 75(1-2):43–50, 2000.

[16] Alan M. Frieze and Ravi Kannan. The regularity lemma and approximation schemes for dense problems. FOCS '96, pages 12–20, 1996.

[17] Naveen Garg, Vijay V. Vazirani, and Mihalis Yannakakis. Approximate max-flow min-(multi)cut theorems and their applications. *SIAM Journal on Computing*, 25:698–707, 1993.

[18] Naveen Garg, Vijay V. Vazirani, and Mihalis Yannakakis. Multiway cuts in node weighted graphs. *J. Algorithms*, 50(1):49–61, 2004.

[19] Michel X. Goemans and David P. Williamson. New 3/4-approximation algorithms for the maximum satisfability problem. *SIAM J. Disc. Math.*, 7(4):656–666, 1994.

[20] Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, November 1995.

[21] David R. Karger, Philip N. Klein, Clifford Stein, Mikkel Thorup, and Neal E. Young. Rounding algorithms for a geometric embedding of minimum multiway cut. *Math. Oper. Res.*, 29(3):436–461, 2004.

[22] Howard J. Karloff, Subhash Khot, Aranyak Mehta, and Yuval Rabani. On earthmover distance, metric labeling, and 0-extension. *SIAM J. Comput.*, 39(2):371–387, 2009.

[23] Alexander V. Karzanov. Minimum 0-extensions of graph metrics. *Eur. J. Comb.*, 19(1):71–101, 1998.

[24] Subhash Khot and Oded Regev. Vertex cover might be hard to approximate to within 2-ϵ. *J. Comput. Syst. Sci.*, 74(3):335–349, May 2008.

[25] Jon Kleinberg and Éva Tardos. Approximation algorithms for classification problems with pairwise relationships: metric labeling and markov random fields. *J. ACM*, 49(5):616–639, 2002.

[26] Tom Leighton and Satish Rao. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *J. ACM*, 46(6):787–832, 1999.

[27] Nathan Linial, Eran London, and Yuri Rabinovich. The geometry of graphs and some of its algorithmic applications. *Combinatorica*, 15(2):215–245, 1995.

[28] Rajsekar Manokaran, Joseph (Seffi) Naor, Prasad Raghavendra, and Roy Schwartz. SDP gaps and UGC hardness for multiway cut, 0-extension, and metric labeling. STOC '08, pages 11–20, 2008.

[29] Joseph Naor and Leonid Zosin. A 2-approximation algorithm for the directed multiway cut problem. *SIAM J. Comput.*, 31(2):477–482, 2001.

[30] Ankit Sharma and Jan Vondrák. Multiway cut, pairwise realizable distributions, and descending thresholds. In *Symposium on Theory of Computing, STOC 2014*, pages 724–733, 2014.