From Graph to Hypergraph Multiway Partition: Is the Single Threshold the Only Route?

Alina Ene[*] Huy L. Nguyễn[†]

August 1, 2014

Abstract

We consider the Hypergraph Multiway Partition problem (Hyper-MP). The input consists of an edge-weighted hypergraph $\mathcal{G} = (V, \mathcal{E})$ and k vertices $s_1, \ldots, s_k$ called terminals. A multiway partition of the hypergraph is a partition (or labeling) of the vertices of $\mathcal{G}$ into k sets $A_1, \ldots, A_k$ such that $s_i \in A_i$ for each $i \in [k]$. The cost of a multiway partition $(A_1, \ldots, A_k)$ is $\sum_{i=1}^{k} w(\delta(A_i))$, where $w(\delta(\cdot))$ is the hypergraph cut function. The Hyper-MP problem asks for a multiway partition of minimum cost.

Our main result is a $4/3$ approximation for the Hyper-MP problem on 3-uniform hypergraphs, which is the first improvement over the $(1.5-1/k)$ approximation of [5]. The algorithm combines the single-threshold rounding strategy of Calinescu *et al.* [3] with the rounding strategy of Kleinberg and Tardos [8], and it parallels the recent algorithm of Buchbinder *et al.* [2] for the Graph Multiway Cut problem, which is a special case.

On the negative side, we show that the KT rounding scheme [8] and the exponential clocks rounding scheme [2] cannot break the $(1.5 - 1/k)$ barrier for arbitrary hypergraphs. We give a family of instances for which both rounding schemes have an approximation ratio bounded from below by $\Omega(\sqrt{k})$, and thus the Graph Multiway Cut rounding schemes may not be sufficient for the Hyper-MP problem when the maximum hyperedge size is large. We remark that these instances have $k = \Theta(\log n)$.

1 Introduction

In this paper, we consider the Hypergraph Multiway Partition problem (Hyper-MP). The input consists of an edge-weighted hypergraph $\mathcal{G} = (V, \mathcal{E})$ and k vertices $s_1, \ldots, s_k$ called terminals. A multiway partition of the hypergraph is a partition (or labeling) of the vertices of $\mathcal{G}$ into k sets $A_1, \ldots, A_k$ such that $s_i \in A_i$ for each $i \in [k]$. The cost of a multiway partition $(A_1, \ldots, A_k)$ is $\sum_{i=1}^{k} w(\delta(A_i))$, where $w(\delta(\cdot))$ is the hypergraph cut function[1]. The Hyper-MP problem asks for a multiway partition of minimum cost. We also parameterize the problem by the maximum cardinality of a hyperedge, denoted by c; the well-studied Graph Multiway Cut problem is the special case for which $c = 2$.

[*]Center for Computational Intractability, Princeton University; Department of Computer Science and DIMAP, University of Warwick. aene@cs.princeton.edu

[†]Department of Computer Science, Princeton University. hlnguyen@princeton.edu

[1]For each set $A \subseteq V$ of vertices, we use $\delta(A)$ to denote the set of all hyperedges leaving A, i.e., hyperedges $e \in \mathcal{E}$ such that $A \cap e$ and $(V - A) \cap e$ are both non-empty. We use $w(\delta(A))$ to denote the total weight of the edges leaving A, i.e., $w(\delta(A)) = \sum_{e \in \delta(A)} w(e)$.

The Hyper-MP problem was introduced by Lawler [9] and it has applications in information storage and retrieval, numerical taxonomy, packaging of electric circuits, and VLSI designs [9, 1]. Alpert *et al.* [1] emphasize that, in the context of VLSI design, the Hyper-MP objective function better reflects the true cost than simply counting the number of hyperedges being cut because a net (represented by a hyperedge) spanning more clusters consumes more I/O and timing resources.

Multiway cut and partition problems are well-studied from a theoretical point of view as well, with the Graph Multiway Cut problem receiving the most attention. Dahlhaus *et al.* [6] initiated the study of the Graph Multiway Cut problem; they showed that the problem is MAX SNP-hard even for $k = 3$ and they gave a combinatorial algorithm that achieves a $(2 - 2/k)$ approximation. In a breakthrough result, Calinescu, Karloff, and Rabani [3] obtained an $(1.5 - 1/k)$ approximation via a novel geometric relaxation. Since the work of [3], the approximation factor has been improved in a series of papers [7, 2, 10], culminating with the 1.30217 approximation of [10]. All of these improvements use the CKR relaxation as a starting point and they combine the single-threshold rounding strategy of [3] with other rounding schemes.

The progress on the Hyper-MP problem has been much slower, however. It was only recently that Chekuri and Ene [5] gave a $(1.5 - 1/k)$ approximation for the Hyper-MP problem, improving on a previous $(2 - 2/k)$ approximation [11]. The algorithm of [5] uses a single-threshold scheme to round a fractional solution to the CKR relaxation and it achieves the best approximation known for the problem. An interesting open question, which is the main motivation behind this work, is whether one can improve the $(1.5 - 1/k)$ factor by using other rounding schemes, and in particular the strategies underpinning the Graph Multiway Cut algorithms. Calinescu *et al.* [3] observed that, in the graph setting, one can assume without loss of generality that the fractional solution has certain properties, namely that each edge is mapped to a segment on the simplex that is arbitrarily small and it is aligned with the simplex. These properties are crucially exploited by the analyses of all of the rounding schemes for the graph problem. Unfortunately, it is unclear how to extend these simplifying assumptions to the hypergraph setting (in the graph setting, they can be easily achieved by subdividing the edges). The work of [5] provides an analysis of the single-threshold rounding scheme without any assumptions on the fractional solution, but this seems challenging for the other rounding schemes even for 3-uniform[2] hypergraphs.

Our contributions. Given the obstacles mentioned above, in this paper we focus on bridging the gap between the graph setting ($c = 2$) and the 3-uniform hypergraph setting ($c = 3$). Our main result is a 4/3 approximation for the Hyper-MP problem on 3-uniform hypergraphs, which is the first improvement over the $(1.5 - 1/k)$ approximation of [5]. We remark that the result immediately extends to the setting in which each hyperedge has at most 3 vertices (instead of exactly 3 vertices).

Theorem 1. *There is a* 4/3 *approximation algorithm for the* Hyper-MP *problem on* 3*-uniform hypergraphs.*

The algorithm of Theorem 1 combines the single-threshold rounding strategy of Calinescu *et al.* [3] with the rounding strategy of Kleinberg and Tardos [8], and it parallels the recent algorithm of Buchbinder *et al.* [2] for the Graph Multiway Cut problem. As we mentioned above, it seems to be very challenging to analyze these rounding strategies in the absence of simplifying assumptions (such as edge alignment in the graph case). A key ingredient in our approach is a replacement for the alignment property that allows us to simplify the instance and the fractional solution when the

[2]A hypergraph is ℓ-uniform if every hyperedge has size ℓ.

$$
\begin{array}{lll}
 & \text{(Hyper-MP LP)} & \\
\min & \displaystyle\sum_{e \in \mathcal{E}} \sum_{i=1}^{k} \Big(\max_{u \in e} x(u,i) - \min_{v \in e} x(v,i) \Big) & \\
 & \displaystyle\sum_{i=1}^{k} x(v,i) = 1 & \forall v \in V \\
 & x(s_i, i) = 1 & \forall i \in [k] \\
 & x(v,i) \geq 0 & \forall v \in V, i \in [k]
\end{array}
$$

Figure 1: LP relaxation for Hyper-MP.

hypergraph is 3-uniform. This ingredient together with some additional insights made the analysis tractable, although it remains quite technical and it is more involved than the analysis for graphs.

On the negative side, we show that the KT rounding scheme [8] and the exponential clocks rounding scheme [2] cannot break the $(1.5 - 1/k)$ barrier for arbitrary hypergraphs. More precisely, we give a family of instances with $c \gg k$ for which both rounding schemes have an approximation ratio bounded from below by $\Omega(\sqrt{k})$, and thus the Graph Multiway Cut rounding schemes may not be sufficient for the Hyper-MP problem when c is large. We remark that these instances have $k = \Theta(\log n)$. These results can be found in Appendix B.

Other related work. As we have already mentioned, the Graph Multiway Cut problem and its generalizations to hypergraphs and submodular functions have been studied extensively over the past two decades. We omit a detailed discussion of these results and we refer the reader to [10, 5, 4] for additional pointers and references.

2 LP Relaxation

We use a standard LP relaxation for the problem (see Figure 1). For each vertex $v \in V$ and each label $i \in [k]$, we have a variable $x(v,i)$ with the interpretation that $x(v,i) = 1$ if vertex v receives label i. It is convenient to write the LP in the form described in Fig. 1; although the objective function is not linear, we can easily rewrite it so that it becomes linear. We remark that the LP relaxation is equivalent to the relaxation of [5].

In the remainder of this section, we show that it suffices to round fractional solutions to the above LP that have some additional properties. We start by introducing some notation and a definition. For a vector $v \in \mathbb{R}^k$ and a set $S \subset [k]$, we denote by $v|_S$ the $|S|$-dimensional vector equal to the restriction of v to the coordinates in S.

Definition 2. Consider an instance of the Hyper-MP problem on a 3-uniform hypergraph $\mathcal{G} = (V, \mathcal{E})$. Let $\mathbf{x}$ be a feasible LP solution for the instance. We classify the hyperedges of $\mathcal{E}$ as follows:

(A) A hyperedge e is of type (A) if there is a permutation a, b, c of the vertices of e such that:

- $\mathbf{x}^b = \mathbf{x}^c$
- $\mathbf{x}^a$ and $\mathbf{x}^b$ differ in only 2 coordinates

(B) A hyperedge e is of type (B) if there is a permutation a, b, c of the vertices of e and a partition (L_1, L_2, L_3, L_4) of $[k]$ such that:

- $\mathbf{x}^a|_{L_1} > \mathbf{x}^b|_{L_1} = \mathbf{x}^c|_{L_1}$
- $\mathbf{x}^b|_{L_2} > \mathbf{x}^c|_{L_2} = \mathbf{x}^a|_{L_2}$
- $\mathbf{x}^c|_{L_3} > \mathbf{x}^a|_{L_3} = \mathbf{x}^b|_{L_3}$
- $\mathbf{x}^a|_{L_4} = \mathbf{x}^b|_{L_4} = \mathbf{x}^c|_{L_4}$

(C) A hyperedge e is of type (C) if there is a permutation a, b, c of the vertices of e and a partition (L_1, L_2, L_3, L_4) of $[k]$ such that:

- $\mathbf{x}^a|_{L_1} < \mathbf{x}^b|_{L_1} = \mathbf{x}^c|_{L_1}$
- $\mathbf{x}^b|_{L_2} < \mathbf{x}^c|_{L_2} = \mathbf{x}^a|_{L_2}$
- $\mathbf{x}^c|_{L_3} < \mathbf{x}^a|_{L_3} = \mathbf{x}^b|_{L_3}$
- $\mathbf{x}^a|_{L_4} = \mathbf{x}^b|_{L_4} = \mathbf{x}^c|_{L_4}$

(D) A hyperedge e is of type (D) if it does not fall into any of the types above.

As shown in the following lemma, it suffices to consider instances of the problem where all the hyperedges fall into one of the first three types (that is, there is no hyperedge of type (D)). It is convenient to have the following definition.

Definition 3. A randomized rounding scheme for the Hyper-MP LP relaxation is *c-preserving* if it constructs an integral solution such that, for each hyperedge e, the expected number of parts in which e is split is at most c times the fractional cost for e.

Lemma 4. *Suppose that there is a rounding scheme for the* Hyper-MP LP *relaxation that is c-preserving for instances of the problem in which all of the hyperedges are of type (A), (B), or (C). Then there is an c-preserving rounding scheme for arbitrary instances of the problem on* 3*-uniform hypergraphs. Moreover, if the former rounding scheme runs in polynomial time then the latter rounding scheme also runs in polynomial time.*

Proof: Consider an instance of Hyper-MP on a 3-uniform hypergraph $\mathcal{G} = (V, \mathcal{E})$, and let $\mathbf{x}$ be a fractional solution for the instance. In the following, we modify the instance and the fractional solution in order to ensure that there are no hyperedges of type (D).

Let e be a hyperedge of type (D). Suppose that there exists a permutation a, b, c of the vertices of e and two labels $i, j \in [k]$ such that $x(a, i) > \max\{x(b, i), x(c, i)\}$ and $x(a, j) < \min\{x(b, j), x(c, j)\}$. We modify the instance and the fractional solution as follows. We add a new vertex a' and we replace the hyperedge $\{a, b, c\}$ by two hyperedges, $\{a, a'\}$ and $\{a', b, c\}$. We define a fractional assignment for a' as follows. Let

$$\epsilon = \min\{x(a, i) - \max\{x(b, i), x(c, i)\}, \min\{x(b, j), x(c, j)\} - x(a, j)\} > 0$$

We set $x(a', i) = x(a, i) - \epsilon$, $x(a', j) = x(a, j) + \epsilon$, and $x(a', \ell) = x(a, \ell)$ for all labels $\ell \neq i, j$. Note that the fractional cost of the hyperedges $\{a, a'\}$ and $\{a', b, c\}$ is equal to the fractional cost of the hyperedge $\{a, b, c\}$. Additionally, we can map a multiway partition of $V \cup \{a'\}$ to a multiway partition of V by simply removing a' from the part that contains it; a straightforward case analysis shows that this mapping does not increase the integral cost, since the total contribution of $\{a', b, c\}$ and $\{a, a'\}$ to the integral cost of the former partition is at most the contribution of $\{a, b, c\}$ to the integral cost of the latter partition.

By repeatedly applying the transformation above we may assume that, for any permutation a, b, c of the vertices of e, there do not exist two labels $i, j \in [k]$ such that $x(a, i) > \max\{x(b, i), x(c, i)\}$ and $x(a, j) < \min\{x(b, j), x(c, j)\}$. We now show that, for every permutation a, b, c of the vertices of e, there is a partition (L_1, L_2, L_3, L_4) of $[k]$ such that:

- $\mathbf{x}^a|_{L_1} \neq \mathbf{x}^b|_{L_1} = \mathbf{x}^c|_{L_1}$
- $\mathbf{x}^b|_{L_2} \neq \mathbf{x}^c|_{L_2} = \mathbf{x}^a|_{L_2}$
- $\mathbf{x}^c|_{L_3} \neq \mathbf{x}^a|_{L_3} = \mathbf{x}^b|_{L_3}$
- $\mathbf{x}^a|_{L_4} = \mathbf{x}^b|_{L_4} = \mathbf{x}^c|_{L_4}$

Consider a permutation a, b, c of the vertices of e. We define four sets $L_1, \ldots, L_4$ as follows:

- $L_1 = \{i \in [k] \colon x(a, i) \neq x(b, i) = x(c, i)\}$
- $L_2 = \{i \in [k] \colon x(b, i) \neq x(c, i) = x(a, i)\}$
- $L_3 = \{i \in [k] \colon x(c, i) \neq x(a, i) = x(b, i)\}$
- $L_4 = \{i \in [k] \colon x(a, i) = x(b, i) = x(c, i)\}$

We can verify that the sets $L_1, \ldots, L_4$ above partition the labels as follows. The sets are disjoint and thus it suffices to check that their union is $[k]$. Suppose for contradiction that there is a label i such that $i \notin L_1 \cup L_2 \cup L_3 \cup L_4$; thus $x(a, i) \neq x(b, i) \neq x(c, i)$. Let a', b', c' be the permutation of $\{a, b, c\}$ such that $x(a', i) < x(b', i) < x(c', i)$. For any label $j \neq i$, one of the following must hold:

- $x(a', j) \leq \min\{x(b', j), x(c', j)\}$, or
- $x(a', j) > \min\{x(b', j), x(c', j)\}$

Suppose that $x(a', j) > \min\{x(b', j), x(c', j)\}$. It follows from our assumption that $x(c', j) \geq \min\{x(a', j), x(b', j)\}$ and $x(a', j) = \max\{x(b', j), x(c', j)\}$, and therefore $x(a', j) = x(c', j) \geq x(b', j)$. Thus, for any label $j \neq i$, we have $x(a', j) \leq x(c', j)$. Since $x(a', i) < x(c', i)$, we have $\sum_{\ell=1}^{k} x(a', \ell) < \sum_{\ell=1}^{k} x(c', \ell)$. But this is impossible, since $\sum_{\ell=1}^{k} x(a', \ell) = \sum_{\ell=1}^{k} x(c', \ell) = 1$.

Therefore the sets $L_1, \ldots, L_4$ partition the label set $[k]$, as claimed. Since $\mathbf{x}^a|_{L_1} \neq \mathbf{x}^b|_{L_1} = \mathbf{x}^c|_{L_1}$, we have two cases: $\mathbf{x}^a|_{L_1} > \mathbf{x}^b|_{L_1} = \mathbf{x}^c|_{L_1}$ and $\mathbf{x}^a|_{L_1} < \mathbf{x}^b|_{L_1} = \mathbf{x}^c|_{L_1}$. We consider each of these cases in turn.

Suppose that $\mathbf{x}^a|_{L_1} < \mathbf{x}^b|_{L_1} = \mathbf{x}^c|_{L_1}$. In the following, we show that $\mathbf{x}^b|_{L_2} < \mathbf{x}^c|_{L_2}$ and $\mathbf{x}^c|_{L_3} < \mathbf{x}^a|_{L_3}$. Suppose for contradiction that $\mathbf{x}^b|_{L_2} > \mathbf{x}^c|_{L_2}$. Then $x(a, j) \leq x(b, j)$ for each $j \in [k]$ and $x(a, \ell) < x(b, \ell)$ for at least one label ℓ. Therefore $\sum_{j=1}^{k} x(a, j) < \sum_{j=1}^{k} x(b, j)$, which is a contradiction. A similar argument shows that we have $\mathbf{x}^c|_{L_3} < \mathbf{x}^a|_{L_3}$. Thus the hyperedge is of type (B).

Suppose that $\mathbf{x}^a|_{L_1} > \mathbf{x}^b|_{L_1} = \mathbf{x}^c|_{L_1}$. In the following, we show that $\mathbf{x}^b|_{L_2} > \mathbf{x}^c|_{L_2}$ and $\mathbf{x}^c|_{L_3} > \mathbf{x}^a|_{L_3}$. Suppose for contradiction that $\mathbf{x}^b|_{L_2} < \mathbf{x}^c|_{L_2}$. Then $x(a, j) \geq x(b, j)$ for each $j \in [k]$ and $x(a, \ell) > x(b, \ell)$ for at least one label ℓ. Therefore $\sum_{j=1}^{k} x(a, j) > \sum_{j=1}^{k} x(b, j)$, which is a contradiction. A similar argument shows that $\mathbf{x}^c|_{L_3} > \mathbf{x}^a|_{L_3}$. Thus the hyperedge is of type (C). $\square$

Algorithm 1: Single threshold rounding

Pick $\theta \in (0,1]$ with prob. density $\phi(\theta)$
Let $A_i \leftarrow \emptyset$ for each $i \in [k]$
Let $U \leftarrow V$ $\langle\langle$*Unlabeled vertices*$\rangle\rangle$
For $i = 1$ to $k-1$
  $A_i \leftarrow U \cap \{v \in V : x(v,i) \geq \theta\}$
  $U \leftarrow U - A_i$
$A_k \leftarrow U$
Return $(A_1, \ldots, A_k)$

Algorithm 2: Kleinberg-Tardos rounding

Let $A_i \leftarrow \emptyset$ for each $i \in [k]$
Let $U \leftarrow V$ $\langle\langle$*Unlabeled vertices*$\rangle\rangle$
While U is non-empty
  Pick $\theta \in (0,1]$ uniformly at random
  Pick $i \in [k]$ uniformly at random
  $A_i \leftarrow A_i \cup \Big(U \cap \{v \in V : x(v,i) \geq \theta\}\Big)$
  $U \leftarrow U - A_i$
Return $(A_1, \ldots, A_k)$

Algorithm 3: Combined rounding scheme

With probability 1/3, run Algorithm 1 with $\phi(t) = 2t$ for all $t \in [0,1]$
With probability 2/3, run Algorithm 2

Figure 2: The rounding algorithms.

3 Rounding Algorithms

In this section, we give a rounding scheme that achieves a 4/3 approximation for the Hyper-MP problem on 3-uniform hypergraphs. In the following, we let $\mathbf{x}$ be a fractional solution to the LP relaxation given in Section 2. The rounding strategies that we use have been studied in previous work for the Graph Multiway Cut and Uniform Metric Labeling problems [2, 10, 8], and they are given in Figure 2.

We devote the rest of this section to the analysis of Algorithm 3. As shown in Lemma 4, we may assume that we have a fractional solution $\mathbf{x}$ such that each hyperedge is of type (A), (B), or (C) (see Definition 2).

For any multiway partition $(A_1, \ldots, A_k)$, the contribution to the integral cost of a hyperedge e is the number of parts in which e is split, i.e., the number of labels i such that $e \in \delta(A_i)$. We consider each hyperedge in turn and we upper bound its expected contribution to the integral cost.

Theorem 5. *Let e be a hyperedge and let $p(e)$ be a random variable equal to the number of parts in which e is split by Algorithm 3. We have*

$$\mathbf{E}[p(e)] \leq \frac{4}{3} \sum_{i=1}^{k} \Big(\max_{v \in e} x(v,i) - \min_{v \in e} x(v,i) \Big).$$

We consider the hyperedges of each type in turn. It follows from the work of Buchbinder *et al.* [2] that the theorem holds of hyperedges of type (A).

Lemma 6 (Buchbinder *et al.* [2])**.** *Let e be a hyperedge and let $p(e)$ be a random variable equal to the number of parts in which e is split by Algorithm 3. If e is of type (A), we have*

$$\mathbf{E}[p(e)] \leq \frac{4}{3} \sum_{i=1}^{k} \Big(\max_{v \in e} x(v,i) - \min_{v \in e} x(v,i) \Big).$$

Now we consider hyperedges of type (B). Let e be a hyperedge of type (B). Let a, b, c be a permutation of the vertices of e and let $L_1, \dots, L_4$ be a partition of the labels that satisfy the conditions stated in Definition 2. Let $\alpha = \|\mathbf{x}^a|_{L_1} - \mathbf{x}^b|_{L_1}\|_1 = \|\mathbf{x}^b|_{L_2} - \mathbf{x}^a|_{L_2}\|_1$ and $\beta = \|\mathbf{x}^a|_{L_4}\|_1$. Note that the fractional cost of e is 3α. In the following, we analyze the expected contribution of e to the integral cost of the solutions constructed by Algorithms 1 and 2.

Lemma 7. *Let $p_1(e)$ be a random variable equal to the number of parts in which e is split in the partition constructed by Algorithm 1. We have*

$$\mathbf{E}[p_1(e)] \leq 2\Bigg(\sum_{i \in L_1} \Big(x(a,i)^2 - x(b,i)^2\Big) + \sum_{i \in L_2} \Big(x(b,i)^2 - x(c,i)^2\Big) + \sum_{i \in L_3} \Big(x(c,i)^2 - x(a,i)^2\Big)\Bigg).$$

Proof: Let $(A_1, \dots, A_k)$ be the multiway partition constructed by Algorithm 1. We say that e is split by L_ℓ, where $\ell \in \{1, 2, 3\}$, if there exists a label $i \in L_\ell$ such that $e \in \delta(A_i)$. (Note that none of the labels in L_4 can split e.)

Let X_ℓ be an indicator random variable equal to 1 iff e is split by L_ℓ. We claim that

$$p_1(e) \leq 2(X_1 + X_2 + X_3).$$

If none of the sets L_1, L_2, L_3 split e, we have $p_1(e) = 0$. Therefore we may assume that at least one of the sets splits e. The edge e is split into 3 parts only if it is split by at least 2 of the sets L_1, L_2, L_3, and it is split into 2 parts only if it is split by at least 1 of the sets L_1, L_2, L_3. Thus

$$p_1(e) \leq X_1 + X_2 + X_3 + 1 \leq 2(X_1 + X_2 + X_3),$$

where the second inequality follows from the assumption that $X_1 + X_2 + X_3 \geq 1$. Therefore we have

$$\begin{aligned}
\frac{1}{2}\,\mathbf{E}[p_1(e)] &\leq \sum_{\ell=1}^{3} \Pr[X_\ell = 1] \\
&\leq \sum_{\ell=1}^{3} \sum_{i \in L_\ell} \Pr[e \in \delta(A_i)] \\
&= \sum_{i \in L_1} \int_{x(b,i)}^{x(a,i)} \phi(t)dt + \sum_{i \in L_2} \int_{x(c,i)}^{x(b,i)} \phi(t)dt + \sum_{i \in L_3} \int_{x(a,i)}^{x(c,i)} \phi(t)dt \\
&= \sum_{i \in L_1} \Big(x(a,i)^2 - x(b,i)^2\Big) + \sum_{i \in L_2} \Big(x(b,i)^2 - x(c,i)^2\Big) + \sum_{i \in L_3} \Big(x(c,i)^2 - x(a,i)^2\Big)
\end{aligned}$$

□

Now we analyze the expected number of parts in which e is split by Algorithm 2. Recall that $\alpha = \|\mathbf{x}^a|_{L_1} - \mathbf{x}^b|_{L_1}\|_1 = \|\mathbf{x}^b|_{L_2} - \mathbf{x}^a|_{L_2}\|_1$.

Lemma 8. *Let $p_2(e)$ be a random variable equal to the number of parts in which e is split in the partition constructed by Algorithm 2. We have*

$$\begin{aligned}
\mathbf{E}[p_2(e)] = \frac{2}{1+2\alpha}\Bigg(3\alpha + \frac{3\alpha^3}{1+\alpha} - \sum_{i \in L_1} \frac{x(b,i)(x(a,i) - x(b,i))}{1+\alpha} \\
- \sum_{i \in L_2} \frac{x(c,i)(x(b,i) - x(c,i))}{1+\alpha} - \sum_{i \in L_3} \frac{x(a,i)(x(c,i) - x(a,i))}{1+\alpha}\Bigg)
\end{aligned}$$

Proof: We have

$$\mathbf{E}[p_2(e)] = 2\Pr[p_2(e) = 2] + 3\Pr[p_2(e) = 3] = 2\Pr[p_2(e) \geq 2] + \Pr[p_2(e) = 3].$$

We analyze the two probabilities separately. We start with $\Pr[p_2(e) \geq 2]$.

Let B be the event that none of the vertices a, b, c of e are assigned at the end of an iteration of Algorithm 2, conditioned on the event that none of the vertices of e are assigned at the beginning of the iteration. The probability that the vertices remain unassigned, given that the label selected in the current iteration is i, is equal to $(1 - \max\{x(a,i), x(b,i), x(c,i)\})$. Thus we have

$$\begin{aligned}
\Pr[B] &= \frac{1}{k}\sum_{i=1}^{k}\Big(1 - \max\{x(a,i), x(b,i), x(c,i)\}\Big)\\
&= 1 - \frac{1}{k}\sum_{i=1}^{k}\max\{x(a,i), x(b,i), x(c,i)\}\\
&= 1 - \frac{1}{k}\left(\sum_{i\in L_1} x(a,i) + \sum_{i\in L_2} x(b,i) + \sum_{i\in L_3} x(c,i) + \sum_{i\in L_4} x(c,i)\right)\\
&= 1 - \frac{1}{k}\left(\sum_{i\in L_1} x(a,i) + \sum_{i\in L_2} x(b,i) + 1 - \sum_{i\in L_1} x(b,i) - \sum_{i\in L_2} x(a,i)\right)\\
&= 1 - \frac{1+2\alpha}{k}
\end{aligned}$$

Let $(A_1, \dots, A_k)$ be the multiway partition constructed by Algorithm 2. For each label i, let P_i denote the probability that $e \subseteq A_i$. We have

$$\Pr[p_2(e) \geq 2] = 1 - \sum_{i=1}^{k} P_i$$

Thus, it suffices to analyze each probability P_i. If $i \in L_1$, the probability P_i satisfies the following recurrence:

$$P_i = \frac{x(b,i)}{k} + \frac{x(a,i) - x(b,i)}{k}\cdot\frac{x(b,i)}{1+\alpha} + \Pr[B]\cdot P_i$$

Indeed, consider an iteration and suppose that none of the vertices of e are assigned at the beginning of the iteration. Recall that, since $i \in L_1$, we have $x(b,i) = x(c,i) < x(a,i)$. Thus, in the current iteration, one of the following holds: all vertices get a label; a gets a label and b and c remain unassigned; all vertices remain unassigned. The first term of the recurrence above, $x(b,i)/k$, corresponds to the event that all the vertices of e are assigned label i in the current iteration. The third term, $\Pr[B]\cdot P_i$, corresponds to the event that all the vertices are assigned label i in a future iteration. The second term, $((x(a,i) - x(b,i))/k)\cdot(x(b,i)/(1+\alpha))$, corresponds to the event that a is assigned label i and b and c are assigned label i in future iterations: $(x(a,i) - x(b,i))/k$ is the probability that a is assigned label i in the current iteration and b and c remain unassigned at the end of the iteration; $x(b,i)/(1+\alpha)$ is the probability that b and c are assigned label i in future iterations (see below for a proof).

We can show that the probability that b and c are assigned label i is equal to $x(b,i)/(1+\alpha)$ as follows. Let Q_i denote the probability that b and c are assigned label $i \in L_1$. The probability Q_i

satisfies the following recurrence:

$$Q_i = \frac{x(b,i)}{k} + \frac{1}{k}\sum_{j=1}^{k}\Big(1 - \max\{x(b,j), x(c,j)\}\Big)\cdot Q_i$$

By rearranging, we get

$$Q_i = \frac{x(b,i)}{\sum_{j=1}^{k}\max\{x(b,j), x(c,j)\}}$$

Finally, we have

$$\begin{aligned}\sum_{j=1}^{k}\max\{x(b,j), x(c,j)\} &= \sum_{j\in L_2} x(b,j) + \sum_{j\in L_1\cup L_3\cup L_4} x(c,j)\\ &= \sum_{j\in L_2} x(b,j) + 1 - \sum_{j\in L_2} x(a,j) = 1+\alpha\end{aligned}$$

Therefore $Q_i = x(b,i)/(1+\alpha)$, as claimed. By rearranging the recurrence for P_i, we get

$$\text{For all } i \in L_1\text{: } P_i = \frac{x(b,i)}{1+2\alpha}\left(1 + \frac{x(a,i)-x(b,i)}{1+\alpha}\right)$$

A similar argument shows that:

$$\text{For all } i \in L_2\text{: } P_i = \frac{x(c,i)}{1+2\alpha}\left(1 + \frac{x(b,i)-x(c,i)}{1+\alpha}\right)$$

and

$$\text{For all } i \in L_3\text{: } P_i = \frac{x(a,i)}{1+2\alpha}\left(1 + \frac{x(c,i)-x(a,i)}{1+\alpha}\right)$$

Now consider a label $i \in L_4$; recall that $x(a,i) = x(b,i) = x(c,i)$. The probability P_i satisfies the following recurrence:

$$P_i = \frac{x(a,i)}{k} + \Pr[B]\cdot P_i$$

By rearranging, we get

$$\text{For all } i \in L_4\text{: } P_i = \frac{x(a,i)}{1+2\alpha}$$

Therefore

$$\begin{aligned}1 - \Pr[p_2(e)\geq 2] = \sum_{i=1}^{k} P_i &= \frac{1-\alpha}{1+2\alpha} + \sum_{i\in L_1}\frac{x(b,i)(x(a,i)-x(b,i))}{(1+2\alpha)(1+\alpha)}\\ &+ \sum_{i\in L_2}\frac{x(c,i)(x(b,i)-x(c,i))}{(1+2\alpha)(1+\alpha)} + \sum_{i\in L_3}\frac{x(a,i)(x(c,i)-x(a,i))}{(1+2\alpha)(1+\alpha)}\end{aligned}$$

Finally, we analyze $\Pr[p_2(e) = 3]$. The hyperedge e is split into 3 parts iff a receives a label in L_1, b receives a label in L_2, and c receives a label in L_3. For each triple (i, j, ℓ), where $i \in L_1$, $j \in L_2$, and $\ell \in L_3$, the probability that first a receives label i then b receives j and finally c receives label ℓ is equal to

$$\frac{(x(a,i)-x(b,i))\cdot(x(b,j)-x(c,j))\cdot(x(c,\ell)-x(a,\ell))}{(1+2\alpha)(1+\alpha)}$$

We prove the identity above as follows. Let R_c be the probability that c is assigned label ℓ, given that a and b are assigned at the beginning of the current iteration and c is unassigned. The probability R_c satisfies the recurrence

$$R_c = \frac{x(c,\ell)-x(a,\ell)}{k} + \frac{1}{k}\sum_{t=1}^{k}\Big(1 - x(c,t)\Big)\cdot R_c.$$

The first term is the probability that c receives label ℓ in the current iteration and the second term is the probability that c receives label ℓ in a future iteration. By rearranging, we get

$$R_c = \frac{x(c,\ell)-x(a,\ell)}{\sum_{t=1}^{k} x(c,t)} = x(c,\ell)-x(a,\ell)$$

Let $R_{b,c}$ be the probability that b is assigned label j and then c is assigned label ℓ, given that a is assigned at the beginning of the current iteration and b and c are unassigned. The probability $R_{b,c}$ satisfies the following recurrence:

$$R_{b,c} = \frac{x(b,j)-x(c,j)}{k}\cdot R_c + \frac{1}{k}\sum_{t=1}^{k}\Big(1 - \max\left\{x(b,t), x(c,t)\right\}\Big)\cdot R_{b,c}$$

The first term is the probability that b receives label j in the current iteration and c receives label ℓ in a future iteration. The second term is the probability that, in future iterations, first b receives label j and then c receives label ℓ. By rearranging, we get

$$R_{b,c} = \frac{x(b,j)-x(c,j)}{\sum_{t=1}^{k}\max\left\{x(b,t), x(c,t)\right\}}\cdot R_c = \frac{x(b,j)-x(c,j)}{1+\alpha}\cdot R_c$$

Finally, let $R_{a,b,c}$ be the probability that first a receives label i, then b receives j, and then c receives label ℓ, given that a, b, c are unassigned at the beginning of the current iteration. The probability $R_{a,b,c}$ satisfies the following recurrence:

$$R_{a,b,c} = \frac{x(a,i)-x(b,i)}{k}\cdot R_{b,c} + \frac{1}{k}\sum_{t=1}^{k}\Big(1 - \max\left\{x(a,t), x(b,t), x(c,t)\right\}\Big)\cdot R_{a,b,c}$$

The first term is the probability that a receives label i in the current iteration and, in future iterations, first b receives label j and then c receives label ℓ. the second term is the probability that, in future iterations, first a receives label i, then b receives label j, and then c receives label ℓ. By rearranging, we get

$$R_{a,b,c} = \frac{x(a,i)-x(b,i)}{\sum_{t=1}^{k}\max\left\{x(a,t), x(b,t), x(c,t)\right\}}\cdot R_{b,c} = \frac{x(a,i)-x(b,i)}{1+2\alpha}\cdot R_{b,c}$$

Using a similar argument, we can analyze the probability that a, b, c receive labels i, j, ℓ in a different order. By summing over all possible choices of the labels i, j, ℓ and all possible orders in which a, b, c receive labels i, j, ℓ (respectively), we get

$$\Pr[p_2(e) = 3] = \frac{6\alpha^3}{(1+2\alpha)(1+\alpha)}$$

By putting everything together, we get

$$\begin{aligned}\mathbf{E}[p_2(e)] &= 2\Pr[p_2(e) \geq 2] + \Pr[p_2(e) = 3] \\ &= \frac{2}{1+2\alpha}\Bigg(3\alpha + \frac{3\alpha^3}{1+\alpha} - \sum_{i \in L_1} \frac{x(b,i)(x(a,i) - x(b,i))}{1+\alpha} \\ &\quad - \sum_{i \in L_2} \frac{x(c,i)(x(b,i) - x(c,i))}{1+\alpha} - \sum_{i \in L_3} \frac{x(a,i)(x(c,i) - x(a,i))}{1+\alpha}\Bigg)\end{aligned}$$

□

Using Lemma 7 and Lemma 8, we can analyze the expected integral cost of e as follows. The proof of the lemma follows from a somewhat lengthy calculation and it can be found in Appendix A.

Lemma 9. *Let e be a hyperedge of type (B). Let $p(e)$ be a random variable equal to the number of parts in which e is split in the partition constructed by Algorithm 3. We have*

$$\mathbf{E}[p(e)] \leq \frac{4}{3} \sum_{i=1}^{k} \Big(\max_{v \in e} x(v,i) - \min_{v \in e} x(v,i) \Big).$$

Finally, we consider hyperedges of type (C). The analysis is similar as for edges of type (B) and it is even simpler, since a hyperedge of type (C) cannot be split into 3 parts.

Let a, b, c be a permutation of the vertices of e and let $L_1, \ldots, L_4$ be a partition of the labels that satisfy the conditions stated in Definition 2. Let $\alpha = \|\mathbf{x}^b|_{L_1} - \mathbf{x}^a|_{L_1}\|_1 = \|\mathbf{x}^a|_{L_2} - \mathbf{x}^b|_{L_2}\|_1$ and $\beta = \|\mathbf{x}^a|_{L_4}\|_1$. Note that the fractional cost of e is 3α. In the following, we analyze the expected contribution of e to the integral cost of the solutions constructed by Algorithms 1 and 2.

Lemma 10. *Let $p_1(e)$ be a random variable equal to the number of parts in which e is split in the partition constructed by Algorithm 1. We have*

$$\mathbf{E}[p_1(e)] \leq 2\Bigg(\sum_{i \in L_1} \Big(x(b,i)^2 - x(a,i)^2\Big) + \sum_{i \in L_2} \Big(x(c,i)^2 - x(b,i)^2\Big) + \sum_{i \in L_3} \Big(x(a,i)^2 - x(c,i)^2\Big) \Bigg).$$

Proof: Notice that the hyperedge cannot be split into 3 parts. Thus we have

$$\begin{aligned}&\mathbf{E}[p_1(e)] = 2\Pr[p_1(e) = 2] \\ &\leq 2\Bigg(\sum_{i \in L_1} \int_{x(a,i)}^{x(b,i)} \phi(t)dt + \sum_{i \in L_2} \int_{x(b,i)}^{x(c,i)} \phi(t)dt + \sum_{i \in L_3} \int_{x(c,i)}^{x(a,i)} \phi(t)dt \Bigg) \\ &= 2\Bigg(\sum_{i \in L_1} \Big(x(b,i)^2 - x(a,i)^2\Big) + \sum_{i \in L_2} \Big(x(c,i)^2 - x(b,i)^2\Big) + \sum_{i \in L_3} \Big(x(a,i)^2 - x(c,i)^2\Big) \Bigg)\end{aligned}$$

□

Now we analyze the expected number of parts in which e is split by Algorithm 2. Recall that $\alpha = \|\mathbf{x}^b|_{L_1} - \mathbf{x}^a|_{L_1}\|_1 = \|\mathbf{x}^a|_{L_2} - \mathbf{x}^b|_{L_2}\|_1$.

Lemma 11. *Let $p_2(e)$ be a random variable equal to the number of parts in which e is split in the partition constructed by Algorithm 2. We have*

$$\mathbf{E}[p_3(e)] = \frac{2}{1+\alpha}\Bigg(3\alpha - \sum_{i \in L_1} x(a,i)(x(b,i) - x(a,i)) \\ - \sum_{i \in L_2} x(b,i)(x(c,i) - x(b,i)) - \sum_{i \in L_3} x(c,i)(x(a,i) - x(c,i))\Bigg)$$

Proof: Notice that the hyperedge cannot be split into 3 parts. Thus we have

$$\mathbf{E}[p_2(e) = 2\Pr[p_2(e) = 2] = 2\Pr[p_2(e) \geq 2].$$

The analysis of $\Pr[p_2(e) \geq 2]$ follows the outline in the proof of Lemma 8.

Let B be the event that none of the vertices a, b, c of e are assigned at the end of an iteration of Algorithm 2, conditioned on the event that none of the vertices of e are assigned at the beginning of the iteration. The probability that the vertices remain unassigned, given that the label selected in the current iteration is i, is equal to $(1 - \max\{x(a,i), x(b,i), x(c,i)\})$. Thus we have

$$\begin{aligned}
\Pr[B] &= \frac{1}{k}\sum_{i=1}^{k}\Big(1 - \max\{x(a,i), x(b,i), x(c,i)\}\Big) \\
&= 1 - \frac{1}{k}\sum_{i=1}^{k}\max\{x(a,i), x(b,i), x(c,i)\} \\
&= 1 - \frac{1}{k}\left(\sum_{i \in L_1} x(b,i) + \sum_{i \in L_2 \cup L_3 \cup L_4} x(a,i)\right) \\
&= 1 - \frac{1+\alpha}{k}
\end{aligned}$$

Let $(A_1, \ldots, A_k)$ be the multiway partition constructed by Algorithm 2. For each label i, let P_i denote the probability that $e \subseteq A_i$. We have

$$\Pr[p_2(e) \geq 2] = 1 - \sum_{i=1}^{k} P_i$$

Thus is suffices to analyze each probability P_i. We have

$$\text{For all } i \in L_1\text{: } P_i = \frac{x(a,i)}{1+\alpha}\Big(1 + x(b,i) - x(a,i)\Big)$$

$$\text{For all } i \in L_2\text{: } P_i = \frac{x(b,i)}{1+\alpha}\Big(1 + x(c,i) - x(b,i)\Big)$$

$$\text{For all } i \in L_3\text{: } P_i = \frac{x(c,i)}{1+\alpha}\Big(1 + x(a,i) - x(c,i)\Big)$$

$$\text{For all } i \in L_4\text{: } P_i = \frac{x(a,i)}{1+\alpha}$$

The proofs of the identities above are very similar to the proof of Lemma 8 and we omit them. Therefore we have

$$1 - \Pr[p_2(e) \geq 2] = \sum_{i=1}^{k} P_i = \frac{1-2\alpha}{1+\alpha} + \sum_{i \in L_1} \frac{x(a,i)(x(b,i)-x(a,i))}{1+\alpha} + \sum_{i \in L_2} \frac{x(b,i)(x(c,i)-x(b,i))}{1+\alpha} + \sum_{i \in L_3} \frac{x(c,i)(x(a,i)-x(c,i))}{1+\alpha}$$

Therefore

$$\Pr[p_2(e) \geq 2] = \frac{3\alpha}{1+\alpha} - \sum_{i \in L_1} \frac{x(a,i)(x(b,i)-x(a,i))}{1+\alpha} - \sum_{i \in L_2} \frac{x(b,i)(x(c,i)-x(b,i))}{1+\alpha} - \sum_{i \in L_3} \frac{x(c,i)(x(a,i)-x(c,i))}{1+\alpha}$$

□

Using Lemma 10 and Lemma 11, we can analyze the expected integral cost of e as follows. The proof of the lemma follows from a somewhat lengthy calculation and it can be found in Appendix A.

Lemma 12. *Let e be a hyperedge of type (C). Let $p(e)$ be a random variable equal to the number of parts in which e is split in the partition constructed by Algorithm 3. We have*

$$\mathbf{E}[p(e)] \leq \frac{4}{3} \sum_{i=1}^{k} \Big(\max_{v \in e} x(v,i) - \min_{v \in e} x(v,i) \Big).$$

Theorem 5 follows immediately from Lemmas 6, 9, and 12. This completes the analysis of the rounding algorithm.

References

[1] C. J. Alpert and A. B. Kahng. Recent directions in netlist partitioning: a survey. *Integration, the VLSI Journal*, 19(1–2):1–81, 1995.

[2] N. Buchbinder, J. S. Naor, and R. Schwartz. Simplex partitioning via exponential clocks and the multiway cut problem. In *Proc. of ACM STOC*, pages 535–544. ACM, 2013.

[3] G. Calinescu, H. J. Karloff, and Y. Rabani. An improved approximation algorithm for multiway cut. *Journal of Computer and System Sciences*, 60(3):564–574, 2000. Preliminary version in STOC 1998.

[4] C. Chekuri and A. Ene. Approximation algorithms for submodular multiway partition. In *Proc. of IEEE FOCS*, pages 807–816, 2011.

[5] C. Chekuri and A. Ene. Submodular cost allocation problem and applications. In *Proc. of ICALP*, pages 354–366, 2011.

[6] E. Dahlhaus, D. S. Johnson, C. H. Papadimitriou, P. D. Seymour, and M. Yannakakis. The complexity of multiterminal cuts. *SIAM Journal on Computing*, 23(4):864–894, 1994. Preliminary version in STOC 1992.

[7] D. R. Karger, P. N. Klein, C. Stein, M. Thorup, and N. E. Young. Rounding algorithms for a geometric embedding of minimum multiway cut. *Mathematics of Operations Research*, 29(3):436–461, 2004. Preliminary version in STOC 1999.

[8] J. M. Kleinberg and É. Tardos. Approximation algorithms for classification problems with pairwise relationships: Metric labeling and Markov random fields. *Journal of the ACM*, 49(5):616–639, 2002. Preliminary version in FOCS 1999.

[9] E. L. Lawler. Cutsets and partitions of hypergraphs. *Networks*, 3(3):275–285, 1973.

[10] A. Sharma and J. Vondrák. Multiway cut, pairwise realizable distributions, and descending thresholds. In *Proc. of ACM STOC*, 2014.

[11] L. Zhao, H. Nagamochi, and T. Ibaraki. Greedy splitting algorithms for approximating multiway partition problems. *Mathematical Programming*, 102(1):167–183, 2005.

A Omitted proofs from section 3

Proof of Lemma 9: As before, we let $p_1(e)$ and $p_2(e)$ denote the number of parts in which e is split by Algorithm 1 and 2, respectively. Recall that $\alpha = \|\mathbf{x}^a|_{L_1} - \mathbf{x}^b|_{L_1}\|_1 = \|\mathbf{x}^b|_{L_2} - \mathbf{x}^a|_{L_2}\|_1$ and $\beta = \|\mathbf{x}^a|_{L_4}\|_1$. We have

$$\mathbf{E}[p(e)] = \frac{1}{3}\,\mathbf{E}[p_1(e)] + \frac{2}{3}\,\mathbf{E}[p_2(e)].$$

By Lemma 7 and Lemma 8, we have

$$\begin{aligned}
\mathbf{E}[p(e)] &\leq \frac{2}{3}\left(\sum_{i\in L_1}\Big(x(a,i)^2 - x(b,i)^2\Big) + \sum_{i\in L_2}\Big(x(b,i)^2 - x(c,i)^2\Big) + \sum_{i\in L_3}\Big(x(c,i)^2 - x(a,i)^2\Big)\right) \\
&\quad + \frac{4}{3(1+2\alpha)}\left(3\alpha + \frac{3\alpha^3}{1+\alpha} - \sum_{i\in L_1}\frac{x(b,i)(x(a,i)-x(b,i))}{1+\alpha} - \sum_{i\in L_2}\frac{x(c,i)(x(b,i)-x(c,i))}{1+\alpha}\right. \\
&\quad \left. - \sum_{i\in L_3}\frac{x(a,i)(x(c,i)-x(a,i))}{1+\alpha}\right) \\
&= \frac{4}{1+2\alpha}\left(\alpha + \frac{\alpha^3}{1+\alpha}\right) + \sum_{i\in L_1}\Big(x(a,i)-x(b,i)\Big)\left(\frac{2x(a,i)+2x(b,i)}{3} - \frac{4x(b,i)}{3(1+2\alpha)(1+\alpha)}\right) \\
&\quad + \sum_{i\in L_2}\Big(x(b,i)-x(c,i)\Big)\left(\frac{2x(b,i)+2x(c,i)}{3} - \frac{4x(c,i)}{3(1+2\alpha)(1+\alpha)}\right) \\
&\quad + \sum_{i\in L_3}\Big(x(c,i)-x(a,i)\Big)\left(\frac{2x(c,i)+2x(a,i)}{3} - \frac{4x(a,i)}{3(1+2\alpha)(1+\alpha)}\right) \\
&\leq \frac{4}{1+2\alpha}\left(\alpha + \frac{\alpha^3}{1+\alpha}\right) + \sum_{i\in L_1}\alpha\left(\frac{2x(a,i)+2x(b,i)}{3} - \frac{4x(b,i)}{3(1+2\alpha)(1+\alpha)}\right) \\
&\quad + \sum_{i\in L_2}\alpha\left(\frac{2x(b,i)+2x(c,i)}{3} - \frac{4x(c,i)}{3(1+2\alpha)(1+\alpha)}\right) \\
&\quad + \sum_{i\in L_3}\alpha\left(\frac{2x(c,i)+2x(a,i)}{3} - \frac{4x(a,i)}{3(1+2\alpha)(1+\alpha)}\right) \\
&= \frac{4}{1+2\alpha}\left(\alpha + \frac{\alpha^3}{1+\alpha}\right) + \alpha\left(\frac{4+2\alpha-4\beta}{3} - \frac{4(1-\alpha-\beta)}{3(1+2\alpha)(1+\alpha)}\right) \qquad (1) \\
&\leq \frac{4}{1+2\alpha}\left(\alpha + \frac{\alpha^3}{1+\alpha}\right) + \alpha\left(\frac{4+2\alpha}{3} - \frac{4(1-\alpha)}{3(1+2\alpha)(1+\alpha)}\right) \qquad \Big(\text{Since } \alpha, \beta \geq 0\Big) \\
&= \alpha\left(4 + \frac{2\alpha(\alpha-1)(2\alpha+3)}{3(1+2\alpha)(1+\alpha)}\right) \\
&\leq 4\alpha \qquad \Big(\text{Since } 0 \leq \alpha \leq 1\Big)
\end{aligned}$$

In Equation (1), we used the fact that

$$\sum_{i\in L_1} x(b,i) + \sum_{i\in L_2} x(c,i) + \sum_{i\in L_3} x(a,i) = 1-\alpha-\beta$$

and

$$\sum_{i\in L_1} \Big(x(a,i)+x(b,i)\Big) + \sum_{i\in L_2} \Big(x(b,i)+x(c,i)\Big) + \sum_{i\in L_3} \Big(x(c,i)+x(a,i)\Big) = 2+\alpha-2\beta$$

Since the fractional cost for e is 3α, the lemma follows. $\square$

Proof of Lemma 12: As before, we let $p_1(e)$ and $p_2(e)$ denote the number of parts in which e is split by Algorithm 1 and 2, respectively. Recall that $\alpha = \|\mathbf{x}^b|_{L_1} - \mathbf{x}^a|_{L_1}\|_1 = \|\mathbf{x}^a|_{L_2} - \mathbf{x}^b|_{L_2}\|_1$ and $\beta = \|\mathbf{x}^a|_{L_4}\|_1$. We have

$$\mathbf{E}[p(e)] = \frac{1}{3}\,\mathbf{E}[p_1(e)] + \frac{2}{3}\,\mathbf{E}[p_2(e)].$$

By Lemma 10 and Lemma 11, we have

$$\begin{aligned}
\mathbf{E}[p(e)] &\le \frac{2}{3}\Bigg(\sum_{i\in L_1}\Big(x(b,i)^2 - x(a,i)^2\Big) + \sum_{i\in L_2}\Big(x(c,i)^2 - x(b,i)^2\Big) + \sum_{i\in L_3}\Big(x(a,i)^2 - x(c,i)^2\Big)\Bigg)\\
&+ \frac{12\alpha}{3(1+\alpha)} - \frac{4}{3}\Bigg(\sum_{i\in L_1}\frac{x(a,i)(x(b,i)-x(a,i))}{1+\alpha} - \sum_{i\in L_2}\frac{x(b,i)(x(c,i)-x(b,i))}{1+\alpha}\\
&\quad - \sum_{i\in L_3}\frac{x(c,i)(x(a,i)-x(c,i))}{1+\alpha}\Bigg)\\
&= \frac{4\alpha}{1+\alpha} + \sum_{i\in L_1}\Big(x(b,i)-x(a,i)\Big)\left(\frac{2x(a,i)+2x(b,i)}{3} - \frac{4x(a,i)}{3(1+\alpha)}\right)\\
&+ \sum_{i\in L_2}\Big(x(c,i)-x(b,i)\Big)\left(\frac{2x(b,i)+2x(c,i)}{3} - \frac{4x(b,i)}{3(1+\alpha)}\right)\\
&+ \sum_{i\in L_3}\Big(x(a,i)-x(c,i)\Big)\left(\frac{2x(c,i)+2x(a,i)}{3} - \frac{4x(c,i)}{3(1+\alpha)}\right)\\
&\le \frac{4\alpha}{1+\alpha} + \sum_{i\in L_1}\alpha\left(\frac{2x(a,i)+2x(b,i)}{3} - \frac{4x(a,i)}{3(1+\alpha)}\right)\\
&+ \sum_{i\in L_2}\alpha\left(\frac{2x(b,i)+2x(c,i)}{3} - \frac{4x(b,i)}{3(1+\alpha)}\right)\\
&+ \sum_{i\in L_3}\alpha\left(\frac{2x(c,i)+2x(a,i)}{3} - \frac{4x(c,i)}{3(1+\alpha)}\right)\\
&= \frac{4\alpha}{1+\alpha} + \alpha\left(\frac{4-2\alpha-4\beta}{3} - \frac{4(1-2\alpha-\beta)}{3(1+\alpha)}\right) \qquad (2)\\
&\le \frac{4\alpha}{1+\alpha} + \alpha\left(\frac{4-2\alpha}{3} - \frac{4(1-2\alpha)}{3(1+\alpha)}\right) \qquad \Big(\text{Since } \alpha,\beta \ge 0\Big)\\
&= \alpha\left(4 - \frac{2\alpha}{3}\right)
\end{aligned}$$

$$\leq 4\alpha \qquad \Big(\text{Since } 0 \leq \alpha \leq 1 \Big)$$

In Equation (2), we used the fact that

$$\sum_{i \in L_1} x(a,i) + \sum_{i \in L_2} x(b,i) + \sum_{i \in L_3} x(c,i) = 1 - 2\alpha - \beta$$

and

$$\sum_{i \in L_1} \Big(x(a,i) + x(b,i) \Big) + \sum_{i \in L_2} \Big(x(b,i) + x(c,i) \Big) + \sum_{i \in L_3} \Big(x(c,i) + x(a,i) \Big) = 2 - \alpha - 2\beta$$

Since the fractional cost for e is 3α, the lemma follows. □

B A hard example for Kleinberg-Tardos rounding scheme and the exponential clock scheme

In this section, we give a family of instances of the Hyper-MP problem for which the Kleinberg-Tardos rounding scheme (Algorithm 2 in Section 3) and the exponential clock scheme of [2] achieve an approximation that is bounded from below by $\Omega(\sqrt{k})$.

Consider a hyperedge with 2^{k-1} vertices whose fractional assignments are $\{a_1/(k-1), a_2/(k-1), \ldots, a_{k-1}/(k-1), b\}$, where $a_i \in \{0,1\}$ and $b = 1 - \sum_i a_i/(k-1)$. It is clear that the fractional cost of this hyperedge is at most 2. In the following, we analyze the expected integral cost for the hyperedge in the two rounding schemes. We first consider the Kleinberg-Tardos rounding scheme.

Claim 13. *The expected integral cost of the Kleinberg-Tardos rounding scheme is $\Omega(\sqrt{k})$.*

Proof: Let $t = \sqrt{k}/100$ and $(\theta_1, i_1), \ldots, (\theta_t, i_t)$ be the first t pairs of thresholds and labels picked by the algorithm that fall into the region $\cup_{i=1}^{k-1} [0, 1/(k-1)] \times \{i\} \cup [0,1] \times \{k\}$. If a pair of a threshold and a label picked by the algorithm falls outside of this region, then it does not assign any label to any vertex of the hyperedge, so it is safe to ignore such a pair. By the Chernoff bound, with high probability, the number of labels among $i_1, \ldots, i_t$ that are equal to k is concentrated around $n_k = t/2 \pm \Theta(\sqrt{t})$. Similarly, the number of labels among $i_1, \ldots, i_t$ that are smaller than k is concentrated around $r = t/2 \pm \Theta(\sqrt{t})$.

First, consider the θ_j's with $i_j = k$. The probability that all such θ_j's are at least $1/(2n_k) > 1/(t + \Theta(\sqrt{t}))$ is $(1 - 1/(2n_k))^{n_k} > e^{-0.4}$. Thus, with probability at least $e^{-0.4}$, all such θ_j's are at least $1/(2n_k)$. In the rest of the proof, we condition on this event.

By the birthday paradox, with probability at least 1-1/10, the labels among $i_1, \ldots, i_t$ that are smaller than k are all distinct. Without loss of generality, assume they are $i_{\pi_1}, \ldots, i_{\pi_r}$ where $\pi_1 < \pi_2 < \cdots < \pi_r$. For each $j \in \{1, \ldots, r\}$, partition i_{π_j} picks up the vertex v whose first $k-1$ coordinates are either 0 if they are among$\pi_1, \ldots, \pi_{j-1}$ and $1/(k-1)$ otherwise. Note that this vertex is not picked up by partition k because its kth coordinate is $(j-1)/(k-1) < 1/(2n_k)$.

Thus, we conclude that with constant probability, partitions $i_{\pi_1}, \ldots, i_{\pi_r}$ all pick up some vertices, implying that the expected integral cost is $\Omega(\sqrt{k})$. □

Next, we consider the exponential clock rounding scheme of [2]. For convenience, we state the rounding scheme below.

Algorithm 4: Exponential clock rounding scheme

Choose i.i.d. random $Z_i \sim \exp(1)$ for $i = 1, \ldots, k$.
Assign u to $\arg\min_i \{Z_i/u_i : i = 1, \ldots, k\}$.

Claim 14. *The expected integral cost of the exponential clock scheme is* $\Omega(\sqrt{k})$.

Proof: Let $t = \sqrt{k}/100$ and $Z_{\pi_1} \leq Z_{\pi_2} \leq \cdots \leq Z_{\pi_t}$ be the smallest among $Z_1, \ldots, Z_{k-1}$. By the Chernoff bound, with high probability, the number of variables among $Z_1, \ldots, Z_{k-1}$ that are bounded by $10t/k$ is at least t. In other words, $Z_{\pi_t} \leq 10t/k$. With constant probability, $Z_k > 1$. For the rest of the proof, we condition on this event.

Fix $j \in \{1, \ldots, t\}$. Consider the vertex v whose first $k-1$ coordinates are either 0 if it is among $\pi_1, \ldots, \pi_{j-1}$, or $1/(k-1)$ otherwise. We now argue that v is assigned to partition j. For $i = 1, \ldots, k-1$, the value Z_i/v_i is clearly minimized when $i = \pi_j$. Furthermore, the kth coordinate of v is $(j-1)/(k-1)$. Therefore

$$\frac{Z_k(k-1)}{j-1} > \frac{k-1}{j-1} > \frac{10t/k}{1/(k-1)} \geq \frac{Z_{\pi_j}}{v_{\pi_j}}$$

Thus, the partitions $\pi_1, \ldots, \pi_t$ all pick up some vertices from the hyperedge, implying that the expected integral cost is $\Omega(\sqrt{k})$. □