Beginner's Practical: Hypergraph Conductance-based Clustering

MARIIA MESHCHANINOVA, Heidelberg University, Germany

Hypergraph clustering is a problem of partitioning the nodes of a hypergraph into "true clusters" which are densely connected internally and sparsely connected externally. Conductance is a metric measuring the quality of a cut which is defined as the ratio of the weight of the cut to the minimal volume of one side of the cut. In this project, we implement a hypergraph clustering algorithm that heuristically tries to minimize the maximum value of the conductance over all cuts between clusters to find a high-quality hypergraph clustering. The algorithm is based on the existing shared-memory parallel algorithm for hypergraph partitioning `Mt-KaHyPar`.

1 INTRODUCTION

Hypergraph clustering is a problem of partitioning the nodes of a hypergraph into "true clusters" which are densely connected internally and sparsely connected externally. It is useful in many applications, such as semi-supervised learning [6], gene expression analysis [5] and image segmentation [3]. Conductance metric, which is defined as the ratio of the weight of a given cut to the minimal volume of one side of the cut, is widely used in graph clustering [2, 4] but to the best of our knowledge, it has only been applied as a metric for evaluating the quality of a hypergraph clustering [1]. Therefore in this project, we implement a heuristic hypergraph clustering algorithm with the objective of minimizing the maximum value of the conductance over all cuts between the clusters while partitioning the hypargraph in at most k clusters.

Our algorithm is based on the existing shared-memory parallel hypergraph k-way partitioner `Mt-KaHyPar` and uses the same multilevel approach as the original algorithm. First, the hypergraph is coarsened in stages by detecting and merging node commubities, then the coarsest hypergraph is partitioned into k clusters, and finally the initial partitioning is refined by uncoarsening the hypergraph nodes and applying a local search algorithm to improve the given objective. We implement two objective functions for the local search algorithm - *conductance_local* and *conductance_global* - ans compare their performance on a benchmark set `ibm01...18` against a state of the art hypergraph partitioner `HyperModularity`.

In following, we define the used concepts in Section 2, then we describe related work on hypergraph partitioning and provide an overview of the existing conductance-based clustering algorithms in Section 3. In Section 4, we describe the implementation of the new objectives for optimizing the conductance. Afterwards, we present the experimental results of our implementation in Section 5. Finally, we conclude the paper and discuss future work in Section 6.

2 PRELIMINARIES

In this section, we give an overview of the used concepts and notations and define conductance of a hypergraph.

Hypergraph is a generalization of a graph, where a hyperedge connects one, two or more nodes. Formally, a hypergraph $H = (V, E, c, \omega)$ consists of a non-empty set of nodes V, a set of hyperedges (also called *nets* and simply *edges*) $E \subseteq 2^V$ and weight functions for hypernodes and hyperedges $c : V \longrightarrow \mathbb{N}$, $\omega : E \longrightarrow \mathbb{N}$. If case of unweighted nodes or hyperedges, we set $c(v) = 1$ for all $v \in V$ or $\omega(e) = 1$ for all $e \in E$, respectively.

k**-way clustering** of a hypergraph $H = (V, E, c, \omega)$ is a partitioning of the set of nodes V into k disjoint subsets $V_1, \ldots, V_k$ called *clusters*. Empty clusters are allowed, so k is only an upper bound

Author's address: Mariia Meshchaninova, mariia.meshchaninova@stud.uni-heidelberg.de,InformatikB.Sc.100%,4740180, Heidelberg University, Im Neuenheimer Feld 205, Heidelberg, Baden-Württemberg, Germany, 69120.

on the number of clusters. The algorithm is allowed to eventially move all the nodes of a cluster to other clusters if it would be beneficial for the objective function.

Cut ∂S of a hypergraph $H = (V, E, c, \omega)$ is partitioning of the set of nodes V into two disjoint non-empty subsets S and $V \backslash S$. A hyperedge $e \in E$ is called a **cutting edge** of the cut ∂S if it has at least one pin in S and at least one pin in $V \backslash S$. The weight of the cut is defined as the sum of the weights of the cutting edges of the cut ∂S:

$$\omega(\partial S) = \sum_{e \in \partial S} \omega(e)$$

Therefore In case of unweighted hyperedges, the weight of the cut is equal to the number of cutting edges of the cut.

Pin of a hyperedge $e \in E$ is a node $v \in V$ such that $v \in e$. A hyperedge with only one pin is called a **sinle-pin net**. In the implementation, we use number of pins $PinNum_i(e)$ of a hyperedge $e \in E$ in a cluster V_i to decide whether to move a node v from one cluster V_i to another cluster V_j.

Weighted degree of a node $v \in V$ is defined the same way in hypergraphs as in graphs. It is the sum of the weights of all hyperedges $e \in E$ that contain the node v:

$$\deg_\omega(v) = \sum_{e \in E: v \in E} \omega(e)$$

As during the coarsening phase of the algorithm we merge nodes, we also use **original weighted degree** of a node $v \in V$ $\mathrm{origDeg}_\omega(v)$ which is defined as the sum of the weighted degrees of all nodes $v' \in V$ that were merged into v during the coarsening phase of the algorithm.

Volume of a hypergraph $H = (V, E, c, \omega)$ is defined as the sum of the weighted degrees of all nodes $v \in V$. Analogously, we define the **volume of a cluster** V_i as the sum of the weighted degrees of all nodes $v \in V_i$:

$$\mathrm{vol}(H) = \sum_{v \in V} \deg_\omega(v) \qquad \mathrm{vol}(V_i) = \sum_{v \in V_i} \deg_\omega(v)$$

And for the same reason as for the original weighted degree, we also define the **original volume** of a hypargraph and a cluster:

$$\mathrm{origVol}(H) = \sum_{v \in V} \mathrm{origDeg}_\omega(v) \qquad \mathrm{origVol}(V_i) = \sum_{v \in V_i} \mathrm{origDeg}_\omega(v)$$

This way, the original volume origVol of a cluster or the whole hypergraph in the coarsened hypergraph is equal to the corresponding volume in the initial - *original* - hypergraph.

STOPPED here

Conductance of a cut $S \subseteq V$:

$$\varphi(\mathrm{S}) = \frac{\sum_{e \in \partial S} \omega(e)}{\min\{vol(S), vol(V \backslash S)\}}$$

k-way partition of H: $V = \sqcup_{i=1}^{k} V_k$

Conductance of a k-way partition:

$$\varphi(V_1, \ldots, V_k) = \max_{\emptyset \neq I \subset \{1, \ldots, k\}} \{\varphi(\sqcup_{i \in I} V_i)\}$$

3 RELATED WORK

4 IMPLEMENTATION

5 EXPERIMENTAL RESULTS

6 CONCLUSION

REFERENCES

[1] Ali Aghdaei, Zhiqiang Zhao, and Zhuo Feng. Hypersf: Spectral hypergraph coarsening via flow-based local clustering. In *2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, pages 1–9. IEEE, 2021.

[2] Reid Andersen, Fan Chung, and Kevin Lang. Local graph partitioning using pagerank vectors. In *2006 47th annual IEEE symposium on foundations of computer science (FOCS'06)*, pages 475–486. IEEE, 2006.

[3] Sungwoong Kim, Sebastian Nowozin, Pushmeet Kohli, and Chang Yoo. Higher-order correlation clustering for image segmentation. In J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper_files/paper/2011/file/98d6f58ab0dafbb86b083a001561bb34-Paper.pdf.

[4] Longlong Lin, Ronghua Li, and Tao Jia. Scalable and effective conductance-based graph clustering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4471–4478, 2023.

[5] Ze Tian, TaeHyun Hwang, and Rui Kuang. A hypergraph-based learning algorithm for classifying gene expression and arraycgh data with prior knowledge. *Bioinformatics*, 25(21):2831–2838, 07 2009. ISSN 1367-4803. doi: 10.1093/bioinformatics/btp467. URL https://doi.org/10.1093/bioinformatics/btp467.

[6] Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. Learning with hypergraphs: Clustering, classification, and embedding. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006. URL https://proceedings.neurips.cc/paper_files/paper/2006/file/dff8e9c2ac33381546d96deea9922999-Paper.pdf.