Discovering Large Subsets with High Quality Partitions in Real World Graphs

Yongsub Lim*, Won-Jo Lee*, Ho-Jin Choi and U Kang
Department of Computer Science
KAIST
Email: {yongsub, mochagold, hojinc}@kaist.ac.kr, ukang@cs.kaist.ac.kr

***Abstract*—Given a real world graph, how can we find a large subgraph whose partition quality is much better than the original? Graph partitioning has received great attentions in graph mining, and especially balanced graph partitioning is required in many real world applications. However, the balanced graph partitioning is known to be NP-hard, and moreover it is known that there is no good cut at a large scale for real graphs. Due to this difficulty, in this paper, we propose a new paradigm for graph partitioning. Instead of dealing with the whole graph, our focus is on finding a large subgraph with high quality partitions, in terms of conductance. We show that removing problematic nodes, i.e. large degree nodes called hub nodes in real graphs, remarkably decreases conductance for the remaining giant connected component (GCC), while the number of nodes in the GCC is still significant. In experiments, we demonstrate that our method finds a subgraph of quite a large size with low conductance graph partitions, compared with competing methods. We also show that the competitors cannot find connected subgraphs while our method does, by construction. This improvement in partition quality for the subgraph is especially noticeable for large scale cuts—for a balanced partition, down to 14% of the original conductance with GCC size 70% of the total. As a result, the found subgraph has clear partitions at almost all scales compared with the original, and this result especially helps find communities which are well-formed, but hidden by hubs at various scales in real world graphs like social networks.**

***Keywords*—Graph Partition; Balanced Graph Partition; Conductance**

I. Introduction

In a real world graph, how can we choose a large subset of nodes for which high quality partitions exist compared with the whole graph? Graph partitioning has become an important task due to its wide applications in the real world, including community detection [1], load balancing in distributed systems [2], VLSI design [3], and image segmentation in computer vision [4]. The problem is conceptually well-described and involves grouping nodes so that a group has many internal edges and few external edges, which is usually evaluated by the number of edges across the groups. Especially, in practice, enforcing groups to have balanced sizes is often required. This constraint, however, makes the problem NP-hard, and thus various approaches have been proposed in wide research areas including data mining, computer vision, and theory [4], [5], [6], [7], [8]. Despite such extensive studies, there have been also negative results on graph partitioning targeted at all the nodes for real graphs. Precisely, it is known that there is no good cut at a large scale in real world graphs [9], [10].

In this paper, instead of dealing with all nodes in a graph, we focus on discovering a large subset of nodes that has high quality partitions. It can be understood to identify a large portion of the total for which the problem has a much better solution than for the total. This approach also has various applications like community detection in social networks where communities clearly exist but are hidden or blurred due to other structural properties of the networks. To measure quality of a partition, we use conductance [8], [9], [11], [12], a widely used measure described in Section II, which measures how clearly a group is separated from the other part, and thus especially consider bipartitioning which is used as a basic building block for more general multi-way graph partitioning.

Our main idea is simple and quite intuitive: remove *problematic* nodes, which we will define soon, and work with the remaining well-handled nodes. For the purpose of graph partitioning, there are two sorts of problematic nodes: 1) large degree nodes called hub nodes which increase interdependency between groups, and 2) spokes attached only to the hub nodes which do not contribute to homogeneity within any group. From this idea, we propose MTP (Minus Top-k Partition) which removes hub nodes and computes a partition only for the remaining giant connected component. After this, conductance of the resulting partition is much lower than that for the whole graph while the size of the giant connected component (GCC) remains quite large—remarkable for partitions at large scales like a balanced partition. MTP is also efficient in terms of time and space—excluding the partitioning step, the time and space complexities are linear on a graph size; empirically, using the state-of-the-art graph partitioning method METIS, we show that MTP has linear run time on a graph size.

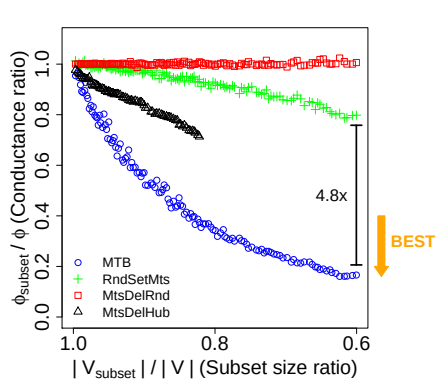
Fig. 1 summarizes our results. Fig. 1a shows the result for CondMat graph data where a subset of nodes found by MTP has a balanced partition whose conductance is lower than that for the whole graph, and also than that found by competing methods. Fig. 1b compares MTP and the competitors for all graph data used in our paper; note that MTP consistently outperforms the others. Fig. 1c shows that SUBSETS[1] found by MTP reduce conductance, compared with the whole graph, at all size scales.

Our main contributions are summarized as follows.

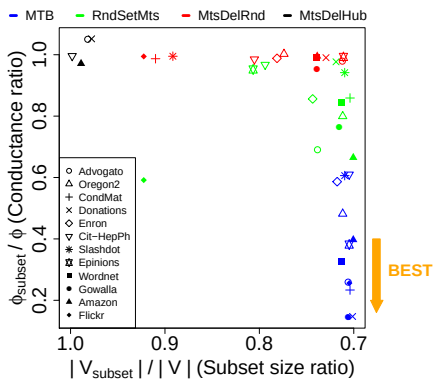
- New Paradigm: Rather than investigating the whole graph, we focus on finding a large subset of nodes that has partitions with much lower conductance.
- Novel Method: By excluding hub nodes and spokes which are problematic in graph partitioning, we find a partition from the remaining part. The method requires linear time and space complexities on a graph size, and due to its simplicity, implementation is quite easy.
- Performance: We show that as more hubs and the corresponding spokes are removed, conductance of a balanced partition for the remaining giant component gets much lower—down to 14% of the original while

*These authors contributed equally to this work.

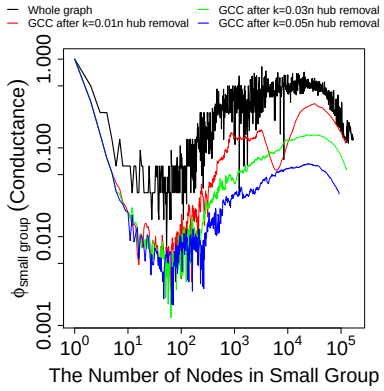
[1]We use SUBSET to indicate a set of nodes in a graph.



(a) Subset size ratio vs. conductance ratio for balanced partitions (CondMat)



(b) Subset size ratio vs. conductance ratio for balanced partitions (best result for each graph)



(c) NCP plot for Flickr

Fig. 1: Our proposed MTP method outperforms competitors. Here, $|V|$ and ϕ denote the number of nodes in the original graph and its conductance of a balanced partition by METIS, respectively. (a) Performance of MTP for CondMat graph data, compared with other competitors described in Section IV. For a balanced partition, the SUBSET found by MTP has significantly lower conductance than the whole graph and also lower than for SUBSETS found by the competitors. (b) Ratio of subset size vs. ratio of conductance for a balanced partition for each graph and each method. Each point chosen is the one having the minimum conductance among the results with a subset size ratio at least 0.7 in Fig. 7. Note that for all graphs, MTP finds quite a large subset whose conductance for a balanced partition is effectively reduced compared with that for the whole graph. In contrast, the competitors fail to find such subsets. (c) NCP plot for Flickr data showing that the SUBSET found by MTP has imbalanced partitions at various size scales with lower conductance than does the whole graph. Here, $n = |V|$. Details of the NCP plot is explained in Section II-A.

TABLE I: Symbol table.

Symbol	Definition
G	a graph
V	a set of nodes in the whole graph
V_{SUBSET}	a set of nodes in the SUBSET
n	the number of nodes of the whole graph
m	the number of edges of the whole graph
k	the number of hub nodes removed
ϕ	conductance of a balanced partition for the whole graph
ϕ_{SUBSET}	conductance of a balanced partition for the SUBSET

the GCC size remains 70% of the total. We also show that the found SUBSET has partitions with lower conductance than the whole graph at *all* size scales, in addition to the balanced case. The running time of MTP with the state-of-the-art graph partitioning method METIS is linear on a graph size.

The codes and data used in this paper are available at `http://kdmlab.org/mtp`. The rest of the paper is organized as follows. In Section II, we give brief preliminaries and discuss related work. We describe the proposed method MTP based on our main idea and discuss complexities of MTP in Section III. After presenting experimental results including comparison of MTP with other competitors in Section IV, we conclude in Section V.

TABLE I lists the symbols used in this paper.

II. Background

A. Preliminaries

Graph Conductance: Conductance is a metric widely used to evaluate the quality of a graph partition [11], [12]. Roughly, this is related to how fast a random walker starting in one group can move to another group. Thus, as connectivity of a group gets internally stronger and externally weaker, its conductance gets lower. Given a graph $G = (V, E)$, the formal

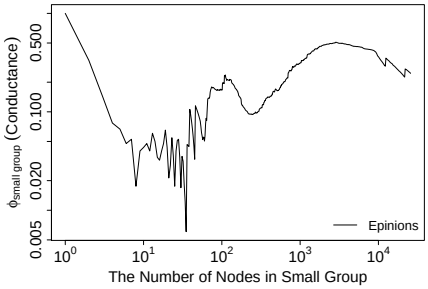


Fig. 2: Example of an NCP plot. This plot shows conductance changes of partitions into two groups at various scales. Recent work [9] reports that NCP plots of real world graphs exhibit V-shapes with the minimum at a small group size of $10 \sim 100$.

definition of conductance $\varphi(A)$ for $A \subseteq V$ is as follows.

$$\varphi(A) = \frac{cut(A)}{\min\{vol(A), vol(\bar{A})\}},$$

where $cut(A) = |\{(u, v) \in E : u \in A, v \in \bar{A}\}|$ and $vol(A) = \sum_{u \in V} \deg(u)$. Note that φ gets smaller as not only the number $cut(A)$ of cross edges tends to be small but also two groups tend to have similar volumes. However, minimizing φ over $A \subset V$ is known to be NP-hard [11]. This minimum value is called the graph conductance of G. Recent work reports that conductance shows the best performance in finding ground-truth communities [8].

Network Community Profile (NCP) Plot [9]: Given a graph, an NCP plot is a plot showing change of conductance over community sizes. Concretely, the x-axis corresponds to the community size and the y-axis to the corresponding conductance. In the original paper [9], drawing the NCP plot in a log-log scale, the authors observed the pattern that NCP plots of real world graphs form V-shapes where the valleys are found around community sizes of $10 \sim 100$. This states the

important structural property of real world graphs that only at a small scale, a good partition exists. Fig. 2 shows an example of the NCP plot for Epinions graph data[2]. The point at x and y implies that y is conductance for a partition of two groups with sizes x and $n - x$ where $x \leq \lfloor n/2 \rfloor$.

METIS*:* METIS is a graph partitioning method based on multilevel ℓ-way partitioning algorithms [6], which is able to compute a balanced bipartition. The overall sequence of METIS consists of three phases: coarsening, initial-partitioning, and refining. In the coarsening phase, a graph is coarsened by aggregating nodes. Starting with the original graph $G_0 = (V_0, E_0)$, for every iteration, nodes in V_i are coalesced to form 'larger' nodes, resulting in V_{i+1} of a smaller size than V_i. In the initial-partitioning phase, ℓ-way partitioning of G_T is computed, where T is the number of iterations in the first phase. Among several ℓ-way partitioning algorithms [13], [14], METIS adopts a multilevel recursive bisection algorithm [6]. In the refining phase, graph G_T is projected to the original graph G_0 by passing through $G_{T-1}, G_{T-2}, ...G_1$ with refinement. A simplified version of Kernighan-Lin partitioning algorithm [15] which incrementally swaps nodes to reduce cross edges of the partitioning was used for the refinement [16], [17]. Recently, METIS has been improved in performance especially for power-law graphs [18].

B. Related Work

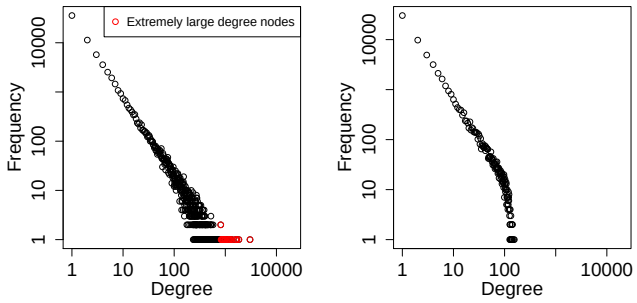
There have been a number of studies on graph partitioning, including METIS [6], spectral clustering [4], cross-association [19], co-clustering [20], and label propagation [2], [21]. Despite different objective functions, they explicitly or implicitly share a common concept of partitioned groups: many intra-edges and few inter-edges.

Overlapping Graph Partitioning: Often, the problem allows or requires overlapping. For example, in community detection for social networks, it may be more natural that people belong to several communities. For overlapping graph partitioning, in recent years various methods have been proposed, including an axiom based method [22], a probabilistic model based method [23], a matrix factorization based method [24], and line grouping [25], [26].

Balanced Graph Partitioning: One issue frequently encountered in practice for graph partitioning is about balancing sizes of partitioned groups. To handle this size constraint, researchers have proposed various metrics such as normalized cut [4], ratio cut [7], and conductance [11]. In general, directly optimizing such metrics is NP-hard, and thus many approximate algorithms and heuristics have been developed [7], [4], [27]. However, since they were not designed for strict balancing, optimizing those metrics often results in quite imbalanced partitioning. More strictly balanced partitioning has been also studied theoretically [28], [29] and empirically [6], [30], [31]. Recently, the problem for graph streams has been also studied [32], [33], [34], [35].

No Good Partition at Large Scale: Despite numerous graph partitioning methods, it has been shown in several studies [36], [9] that there is no good cut for real graphs at a large scale. One reason is that the degree distribution of real world graphs is heavy-tailed [37], [38], implying the existence of hub nodes that may seriously contribute to a large number of cross edges. Rather than finding a good cut in real graphs, researches aimed at finding and evaluating ground-truth communities have been also done with various approaches [8], [39], [40], [41], [42].

[2]http://snap.stanford.edu/data/index.html



(a) Original Graph (b) Shattered by k hubs

Fig. 3: Comparison of degree distributions of the original graph and GCC after removing k hub nodes for Epinions graph data. Here, k is set to 1% of the total nodes in the original graph. Note that in the original graph, there exist hub nodes with extremely large degrees while in the reduced graph, there are no such nodes.

Exploiting Hub Nodes: Recently, to analyze graph structure, there have been several studies that exploit the characteristic of the existence of hub nodes. Siganos et al. [43] proposed a method to group hub nodes first and recursively attach the remaining nodes, resulting in a hierarchical grouping model of a graph. In another study [42], the authors observed that the assortativity coefficient of ground-truth communities can be different from that of the whole graph, and proposed edge-weighting methods to decrease the influence of disassortative edges (e.g. hub-spoke edges), leading to finding communities with high similarity to the ground-truth. Other work [44], [45], sharing a basic idea with our work, was done on graph compression. They proposed an ordering method called SlashBurn that places hub nodes in front, and disconnected nodes appearing due to hub removal in back. These methods regard that hub nodes are few but play a considerably important role in graph structure, and thus specially handle such a property of the hubs. However, they focused on quickly shattering graphs by removing the hub nodes, and there was no discussion about graph partitioning after their removal. In this paper, following such a basic idea to analyze a graph having a heavy-tailed degree distribution, we show that removing hub nodes remarkably decreases conductance values of partitions of the remaining graph. SlashBurn was applied to other related tasks including graph summarization [46] and graph visualization [47].

III. Proposed Method

A. Motivation

One well-known characteristic of many real world graphs is that the degree distribution is heavy-tailed. This is distinct from a random graph with an exponential degree distribution. This implies that there exist hub nodes having very large degrees. In graph partitioning, particularly that with balancing, these hub nodes become seriously problematic: due to their diverse neighbors, assigning them to one group would greatly increase interdependency between groups.

Recent work shows that real world graphs are easily shattered by removing hub nodes [44]. Concretely, removing the hub nodes results in a giant connected component of a significant size, and many disconnected components of very

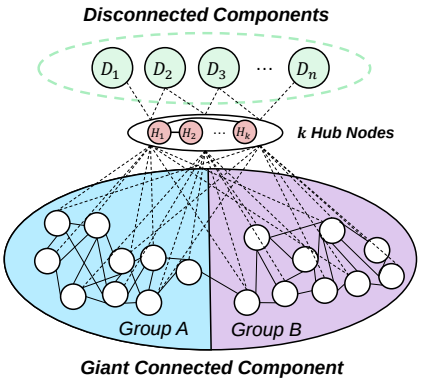


Fig. 4: Illustration of our main idea. We envision a graph consisting of three parts: hub nodes, the giant connected component and disconnected components appearing after hub nodes are removed. The dashed lines represent the edges removed with removal of hubs. Note that after removing the hub nodes, the corresponding giant connected component has a much clearer partition.

small sizes. Although the giant connected component has a structure of hub nodes similar to the whole graph, we observe that there is no hub node with an extremely large degree, as shown in Fig. 3. This observation motivated us to exploit the hubs and disconnected components for high quality partitions. Below, we explain our method, called Minus Top-k Partition (MTP), to find a large subset of nodes for which high quality partitions exist.

B. Minus Top-k Partition (MTP)

The main idea of MTP is to envision a graph as a collection of three parts: hub nodes, spokes only attached to the hub nodes, and the remaining part. Here, the spokes correspond to disconnected components and the remainders correspond to the giant connected component (GCC) after removing the hub nodes. Let $[n] = \{1, \ldots, n\}$ and $G(U)$ is the induced subgraph of $U \subseteq V$. If we remove the set H of hub nodes from a graph, the graph is divided into a set of p connected components $CCSET = \{CC_i \subset V \backslash H : i \in [p]\}$, satisfying

- $CC_1, \ldots, CC_p$ are mutually disjoint sets.
- For every $i \in [p]$ and any pair $(u, v) \in (CC_i)^2$, there is a path between u and v in $G(CC_i)$.
- For every pair $(i, j) \in [p]^2$ and any pair $(u, v) \in CC_i \times CC_j$, there is no path between u and v in $G(V \backslash H)$.

Then, GCC and spokes are formally defined as follows:

$$GCC = \underset{CC \in CCSET}{\operatorname{argmax}} |CC|,$$
$$SPOKES = V \backslash (H \cup GCC).$$

The hub nodes become a major obstacle in finding a good partition because their diverse connectivity makes partitioned groups have high interdependency. Our approach is to exclude those problematic nodes and take the remaining giant connected component as a subgraph for which we hope to obtain a high quality partition (see Fig. 4).

MTP first finds and removes the top-k hub nodes from a graph. As a result, the graph is shattered into a number of connected components as described above. Next, MTP finds the GCC among them, which can be done using a standard graph traversal algorithm like the breadth first search (BFS). Last, it computes a partition (A, B) for the GCC, and then outputs (A, B). Although any partitioning method can be applied, in this paper we use METIS, which is considered the state-of-the-art graph partitioning method [9]. Algorithm 1 describes the whole MTP procedure.

Algorithm 1 Minus Top-k Partition (MTP)

Input: Graph G, the number of removed hubs k.
Output: SUBSETPARTITION (A, B).
1: Find the top-k high degree nodes in G.
2: Remove them from G.
3: Find the giant connected component (GCC).
4: Partition the GCC into (A, B).
5: **Return** (A, B).

MTP is simple, intuitive, and easily implementable. As described in Section IV, MTP discovers a SUBSETPARTITION, a partition of a subgraph, with quite low conductance. Moreover, we compare MTP with other baseline methods to demonstrate non-triviality of our results. We will see that the baseline methods are not effective in reducing conductance, or that they choose a subgraph consisting of many small connected components for which a partition is not very meaningful.

C. Complexity Analysis

Our proposed method MTP is quite efficient in terms of time consumption and space usage. Excluding the partitioning step, the time complexity and the space complexity of MTP are linear on a graph size: $O(n + m)$ and $O(n)$, respectively. The detailed analysis is given below.

Lemma 1: The time complexity of MTP excluding the partitioning step is $O(n + m)$.

Proof: Without computing a partition, MTP consists of the two main steps: 1) removing the top-k hub nodes, and 2) identifying the giant connected component. Step 1) involves finding the top-k hub nodes which can be done in $O(n)$ using Hoare's selection algorithm [48]; Step 2) is done by finding connected components using a standard graph traversal algorithm like the breath-first search (BFS), which takes $O(n+m)$. Hence, the total time complexity excluding the partitioning step becomes $O(n + m)$. ■

Although we exclude the partitioning step in the analysis since its time complexity varies with algorithms used, we empirically show in Section IV that MTP with METIS is fast.

The next lemma states the space complexity of MTP.

Lemma 2: The space complexity of MTP excluding the partitioning step is $O(n)$.

Proof: As we stated, MTP involves the two main computation steps: running the selection algorithm for the first k largest degree nodes, and running a connected component algorithm. In the first step, computing degrees of nodes require $O(n)$ space, and Hoare's selection algorithm require no additional space; in the second step, finding connected components requires $O(n)$. Combining all the space requirements, the lemma is proved. ■

IV. EXPERIMENTS

In this section, experimental results are used to answer:

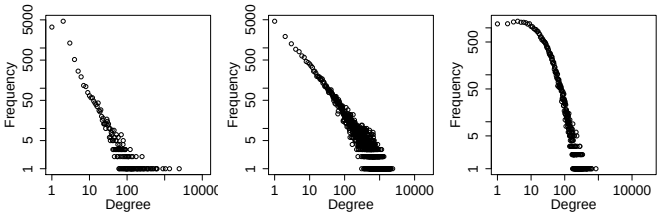
Q1 How low conductance does a SUBSETBALPARTITION[3] by MTP have compared with the whole graph and with other naive methods? (Answers in Observation 1 and 3)

[3] a balanced partition for a subset of nodes.

TABLE II: Summary of the graphs used in our experiments. The number of nodes and edges are counted after taking the giant connected component with removing direction, weights, and self-edges.

Graph	Nodes	Edges	Description
Advogato[1]	5,054	49,821	Trust network
Oregon2[2]	11,461	32,730	Router connections
CondMat[2]	21,363	91,286	Collaboration network
Donations[3]	23,033	877,625	Who donated whom
Enron[2]	33,695	180,810	Enron email data
Cit-HepPh[2]	34,401	420,784	Citation network
Slashdot[1]	51,083	116,573	Reply network
Epinions[2]	75,877	405,739	Trust network
Wordnet[1]	142,505	642,207	Word association network
Gowalla[2]	196,591	950,327	Online social network
Amazon[2]	334,863	925,872	Co-purchasing network
Flickr[4]	404,733	2,110,078	Social network in Flickr

[1]http://konect.uni-koblenz.de [2]http://snap.stanford.edu/data/index.html
[3]http://download.srv.cs.cmu.edu/~mmcgloho/fec/data/fec_data.html
[4]http://www.flickr.com



(a) Oregon2 (b) Donations (c) Cit-HepPh

Fig. 5: Degree distributions of some graphs described in TABLE II. Note that all the data used, exhibits heavy-tailed degree distributions, which means that our main assumption of the existence of hub nodes holds. The other graphs not shown here also show similar patterns.

- Q2 How do conductance values of SUBSETBALPARTITIONS found by MTP change over increasing k? (Answer in Observation 2)
- Q3 How low conductance do SUBSETPARTITIONS by MTP at various size scales have compared with the whole graph? (Answer in Observation 4)
- Q4 How fast is MTP? (Answer in Observation 5)

A. Settings

To verify our method, we gathered graph data from diverse domains such as social networks, collaboration networks, internet connections, and word association. We took only the giant connected component from each graph and made them have no direction, weight, and self-edges. The statistics and brief description of the graph data are presented in TABLE II. Fig. 5 shows degree distributions of some of the graphs. All of them follow heavy-tailed distributions, which means that our assumption of the existence of hub nodes holds.

For partitioning, we use the METIS library of version 5.1.0 given at http://glaros.dtc.umn.edu/gkhome/views/metis.

B. Performance of MTP

Now, we show how good SUBSETBALPARTITION MTP discovers through extensive experiments. We examine conductance of SUBSETBALPARTITIONS found by MTP over the number k of removed hub nodes. To this end, while varying k from 0 to 0.1 with interval 0.001, we 1) remove $\lfloor kn \rfloor$ number of hub nodes from each graph, 2) run METIS to obtain a balanced partition for the giant connected component, and 3) compute conductance for the partition. Note that $k = 0$ implies applying METIS to the whole graph.

Observation 1 (High Quality SUBSETBALPARTITION*):* Conductance of a SUBSETBALPARTITION discovered by MTP is effectively lower than that of a balanced partition for the whole graph.

Observation 2 (Better as k Gets Larger): As the number k of removed hub nodes gets larger, quality gap between a SUBSETBALPARTITION by MTP and a global balanced partition gets much significant. The conductance of the SUBSETBALPARTITION is down to 14% of the global one with SUBSET size 70% of the total.

Fig. 6 shows changes of conductance of SUBSETBALPARTITIONS by MTP and sizes of the corresponding SUBSETS over the number k of removed hub nodes. In general, the conductance of the SUBSETBALPARTITIONS is smaller than that for the whole graph. Notably, as k gets larger, the conductance gap gets much significant, which is consistently exposed by all the used graphs.

We note that size decreases of the SUBSETS are positively correlated with conductance decreases of the corresponding SUBSETBALPARTITIONS. For example, the conductance decrease of a SUBSETBALPARTITION is most remarkable in Oregon2 whose SUBSET size is dramatically reduced over k while Cit-HepPh graph shows the opposite example. Moreover, for all cases, the conductance decrease is much significant compared with the SUBSET size decrease. For example, compared with METIS applied to the whole graph, MTP finds a SUBSET of a size at least 80% of the total, but conductance of the corresponding SUBSETBALPARTITION becomes less than half of the original.

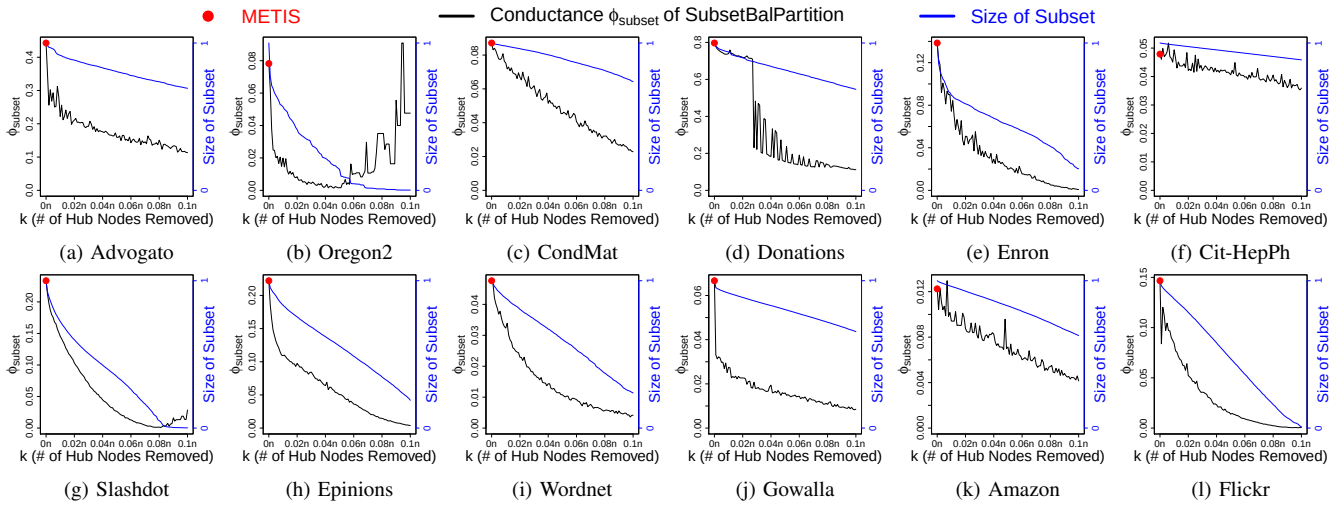
Next, we demonstrate the non-triviality of MTP by comparing with other competitors to find a SUBSETBALPARTITION. Below, the competitors that we consider here are described.

- RndSetMts: Select a random subset of nodes, and apply METIS to that set.
- MtsDelRnd: Compute a balanced partition for the whole graph using METIS, and randomly remove the same number of nodes from each group.
- MtsDelHub: Compute a balanced partition for the whole graph using METIS, and remove the same number of hub nodes from each group.

Observation 3 (Non-triviality of MTP*):* The competitors for finding a SUBSETBALPARTITION do not decrease conductance effectively, or they result in SUBSETS consisting of the giant connected component of an insignificant size and many disconnected components of very small sizes.

Fig. 7 shows the comparison of the SUBSETBALPARTITIONS computed by MTP and the three competitors described above. Given k, the results of RndSetMts and MtsDelRnd are computed by running the methods 10 times and taking averages of the 10 conductance values. For all the competitors, we exclude results if the corresponding SUBSET for which the conductance is computed has a giant connected component of a size less than half of the SUBSET size since the case is less meaningful to compute a balanced partition.

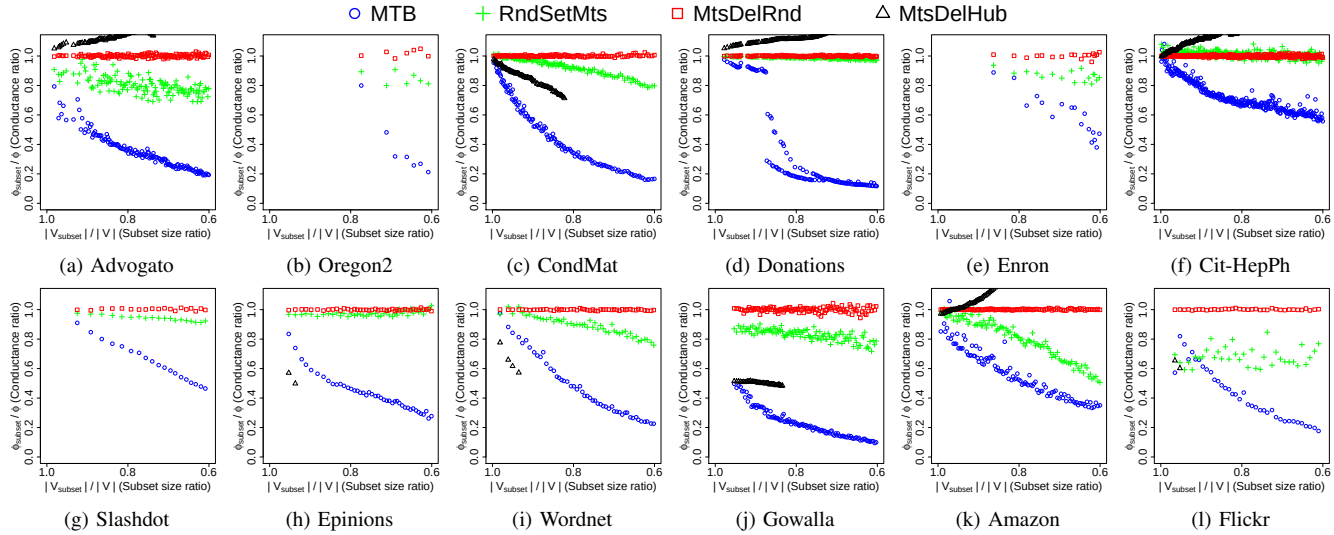
Overall, MTP results in SUBSETBALPARTITIONS with much smaller conductance than those made by competitors,



(a) Advogato (b) Oregon2 (c) CondMat (d) Donations (e) Enron (f) Cit-HepPh

(g) Slashdot (h) Epinions (i) Wordnet (j) Gowalla (k) Amazon (l) Flickr

Fig. 6: MTP finds a large subset of nodes whose conductance for balanced partition is fairly reduced compared with that for the whole graph. For each plot, k denotes the number of hub nodes removed; the red dot denotes conductance computed by METIS for the whole graph; the black line denotes the conductance ϕ_{SUBSET} of the SUBSETBALPARTITIONS; and the blue line denotes the ratio of subset sizes over n. Note that the red dot also corresponds to the case of MTP with $k = 0$. Overall, conductance consistently decreases as k gets larger, and its amount is larger than the decrease of SUBSET sizes.



(a) Advogato (b) Oregon2 (c) CondMat (d) Donations (e) Enron (f) Cit-HepPh

(g) Slashdot (h) Epinions (i) Wordnet (j) Gowalla (k) Amazon (l) Flickr

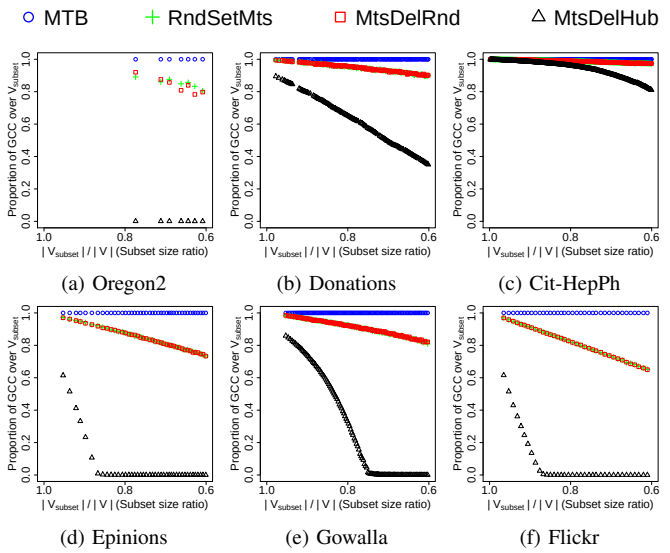
Fig. 7: MTP outperforms competitors in terms of conductance of a SUBSETBALPARTITION. For RndSetMts and MtsDelRnd involving random processes, each value is computed by the average of results over 10 trials. Note that if the GCC of a computed SUBSET is of a size less than half of the subset size, it is not presented. This is because the case is less meaningful to compute a balanced partition. For example, in CondMat, only the black line is cut off, and in extreme cases like Oregon2, there is no black line at all in the plot. Overall, RndSetMts and MtsDelRnd are not effective in reducing the conductance. Although MtsDelHub seems to reduce the conductance effectively for a few graphs, its corresponding GCC size decreases very rapidly (Fig. 8), implying that the computed SUBSET consists of many small connected components.

especially as k gets larger (e.g., in CondMat), MTP is 4.8x better than RndSetMts, 6.1x better than MtsDelRnd, and 1.9x better than MtsDelHub. Although RndSetMts finds SUBSETBALPARTITIONS with low conductance for some graphs like Amazon, MTP still outperforms it. The best result for each method and each graph in Fig. 7, is shown in Fig. 1b.

Fig. 8 shows sizes of GCCs in SUBSETS found by the four methods. None of the competitors find a connected SUBSET at all, while SUBSETS by MTP are always connected by construction. For RndSetMts and MtsDelRnd, their GCCs in computed SUBSETS are quite large, but the corresponding conductance values are not reduced effectively (Fig. 7). Especially, the GCC size of a SUBSET by MtsDelHub decreases fast with increasing k, implying that the SUBSET consists of small connected components in which a balanced partition becomes less meaningful.

Observation 4 (Good Partitions at All Scales): A SUBSET found by MTP has partitions at all scales whose conductance is lower than that of the whole graph at the same scales.

Fig. 9 depicts Network Community Profile (NCP) plots, which we explained in Section II, for SUBSETS found by MTP with $k \in \{0, 0.01n, 0.03n, 0.05n\}$ for each graph. Each line corresponds to an NCP plot for the SUBSET obtained with the



(a) Oregon2 (b) Donations (c) Cit-HepPh

(d) Epinions (e) Gowalla (f) Flickr

Fig. 8: MTP finds a connected SUBSET by construction while SUBSETS by competitors are disconnected. The plots show sizes of GCCs belonging to the SUBSETS found by each method. By construction, SUBSETS by MTP are always connected, leading to the value of 1. For RndSetMts and MtsDelRnd, the decrease of the GCC size is linear. On the other hand, for MtsDelHub, the GCC size dramatically decreases, which means that the subsets found become less meaningful even though it has a balanced partition with low conductance. The graphs not shown here also exhibit similar patterns.

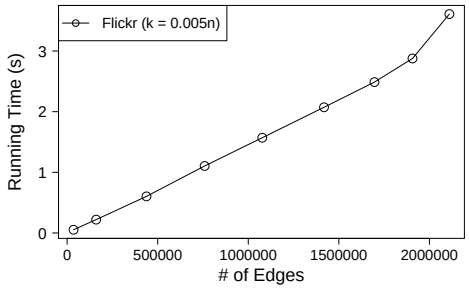


Fig. 10: MTP runs in linear time on the number of edges in a graph. We use principal submatrices of the adjacency matrix of the Flickr graph data.

specified k. From the figure, we observe that an NCP plot gets lower as k becomes larger. For most of the graphs, the NCP plots are clearly separated—it is remarkable especially for Slashdot, Wordnet and Flickr. This means that MTP finds a SUBSET in which partitions at various scales have lower conductance compared with those for the whole graph at the same scales. In other words, the found SUBSET by MTP is partitioned much clearly compared with the whole graph at any scale while keeping V-shape patterns observed in real world graphs.

Observation 5 (Linear Running Time): Running time of MTP is linear on the number of edges in a graph.

With METIS used for the partitioning step, the running time of MTP is linear on the number of edges in a graph as shown in Fig. 10. We took principal submatrices of the adjacency matrix of Flickr to make graphs with appropriate sizes.

V. CONCLUSION

In this paper, we propose MTP, a simple, elegant, and fast method for finding a subset of nodes providing high quality partitions in real world graphs. The main contributions of this work are the followings.

- New Paradigm: Instead of finding a global partition, we focus on finding a large subgraph for which we can find partitions with significantly lower conductance.
- Novel Method: We propose a simple, elegant, and linear time method, called MTP, to find high quality SUBSETPARTITIONS. It removes hub nodes and related disconnected components, and computes a partition only for the giant connected component.
- Performance: We show that in general, MTP discovers a SUBSET of a significant size with lower conductance than the whole graph for a balanced partition, down to 14% of the original conductance with a SUBSET of size 70% of the total. We also show that in a SUBSET by MTP, conductance of partitions at various scales is lower compared with the whole graph. Lastly, MTP runs in linear time on a graph size.

Research on graph partitioning can benefit significantly from the high quality SUBSETPARTITIONS, simplicity, and ease of implementation provided by MTP. Future research directions include scaling up the graph partitioning methods for very large graphs, using distributed systems.

ACKNOWLEDGMENT

This work was supported by the IT R&D program of MSIP/IITP. [10044970, Development of Core Technology for Human-like Self-taught Learning based on Symbolic Approach].

REFERENCES

[1] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3, pp. 75–174, 2010.

[2] J. Ugander and L. Backstrom, "Balanced label propagation for partitioning massive graphs." in *WSDM*, 2013.

[3] A. Sen, H. Deng, and S. Guha, "On a graph partition problem with application to vlsi layout." *Inf. Process. Lett.*, vol. 43, no. 2, pp. 87–94, 1992.

[4] J. Shi and J. Malik, "Normalized cuts and image segmentation," *PAMI*, vol. 22, no. 8, pp. 888–905, 2000.

[5] J. Naor and R. Schwartz, "Balanced metric labeling." in *STOC*, 2005.

[6] G. Karypis and V. Kumar, "Multilevel k-way partitioning scheme for irregular graphs." *J. Parallel Distrib. Comput.*, vol. 48, no. 1, pp. 96–129, 1998.

[7] S. Wang and J. M. Siskind, "Image segmentation with ratio cut," *PAMI*, vol. 25, no. 6, pp. 675–690, 2003.

[8] J. Yang and J. Leskovec, "Defining and evaluating network communities based on ground-truth." in *ICDM*, 2012.

[9] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, "Statistical properties of community structure in large social and information networks," in *WWW*, 2008.

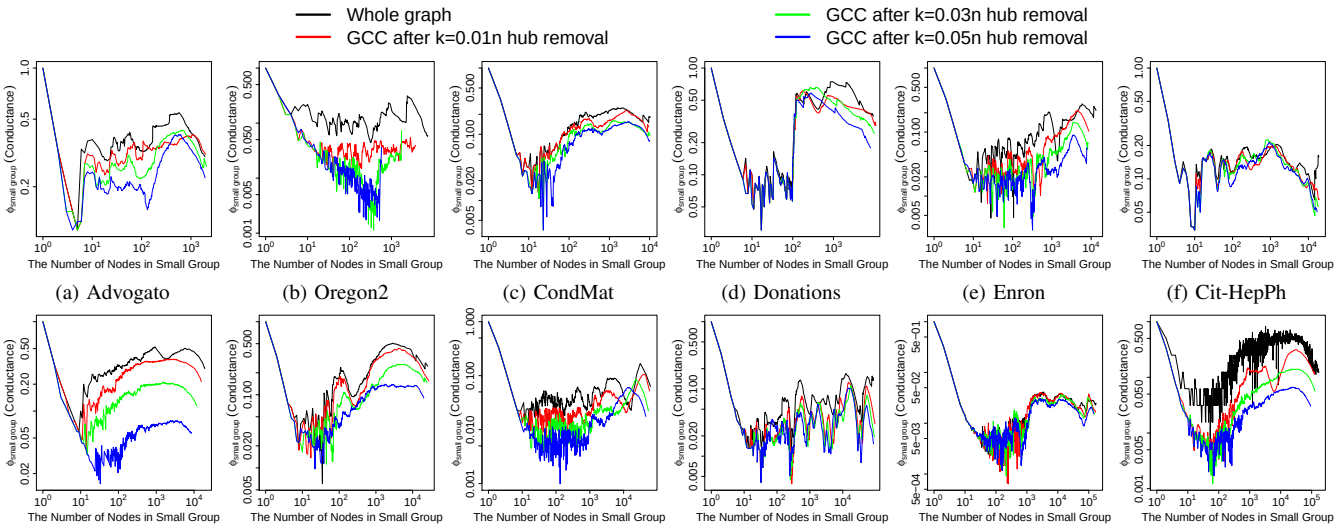
[10] ——, "Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters," Tech. Rep., Oct. 2008.

[11] R. Kannan, S. Vempala, and A. Vetta, "On clusterings: Good, bad and spectral," *J. ACM*, vol. 51, no. 3, pp. 497–515, 2004.

[12] S. E. Schaeffer, "Graph clustering," *Computer Science Review*, vol. 1, no. 1, pp. 27–64, 2007.

[13] C. Fiduccia and R. Mattheyses, "A linear-time heuristic for improving network partitions," *Design Automation, 1982. 19th Conference on*, pp. 175–181, June 1982.

[14] B. Hendrickson and R. W. Leland, "A multi-level algorithm for partitioning graphs." *SC*, vol. 95, p. 28, 1995.



(a) Advogato (b) Oregon2 (c) CondMat (d) Donations (e) Enron (f) Cit-HepPh

(g) Slashdot (h) Epinions (i) Wordnet (j) Gowalla (k) Amazon (l) Flickr

Fig. 9: Conductance of a SUBSET by MTP is lower than the whole graph not only for a balanced partition but also for partitions of various imbalanced sizes. The plots show Network Community Profile (NCP) plots [9], explained in Section II, for each graph with different k values. For each plot, each line is computed by the SNAP library [49] for a SUBSET found by MTP with the specified k. Note that for almost all cases, the NCP plot tends to move down as k gets larger—the pattern is fairly clear, though slightly weaker for Donations, Cit-HepPh and Amazon.

[15] S. L. B. W. Kernighan, "An efficient heuristic procedure for partitioning graphs," *The Bell system technical journal*, 1970.

[16] B. Hendrickson and R. Leland, "The chaco user's guide version 2.0," Sandia National Laboratories, Tech. Rep., 1995.

[17] ——, "An improved spectral graph partitioning algorithm for mapping parallel computations," *SIAM Journal on Scientific Computing*, vol. 16, no. 2, pp. 452–469, 1995.

[18] A. Abou-Rjeili and G. Karypis, "Multilevel algorithms for partitioning power-law graphs," in *IPDPS*, 2006.

[19] D. Chakrabarti, S. Papadimitriou, D. S. Modha, and C. Faloutsos, "Fully automatic cross-associations," in *KDD*, 2004.

[20] I. S. Dhillon, S. Mallela, and D. Modha, "Information-theoretic co-clustering," in *KDD*, 2003.

[21] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Physical Reveiw E*, vol. 76, no. 3, 2007.

[22] S. Arora, R. Ge, S. Sachdeva, and G. Schoenebeck, "Finding overlapping communities in social networks: toward a rigorous approach." in *ACM Conference on Electronic Commerce*, 2012.

[23] P. Gopalan, D. M. Mimno, S. Gerrish, M. J. Freedman, and D. M. Blei, "Scalable inference of overlapping communities." in *NIPS*, 2012.

[24] J. Yang and J. Leskovec, "Overlapping community detection at scale: a nonnegative matrix factorization approach." in *WSDM*, 2013.

[25] T. S. Evans and R. Lambiotte, "Line graphs, link partitions, and overlapping communities," *Physical Review E*, 2009.

[26] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann, "Link communities reveal multiscale complexity in networks," *Nature*, vol. 466, pp. 761–764, 2010.

[27] K. Lang and S. Rao, "A flow-based method for improving the expansion or conductance of graph cuts," in *IPCO*, 2004.

[28] R. Andersen, F. R. K. Chung, and K. J. Lang, "Local graph partitioning using pagerank vectors." in *FOCS*, 2006.

[29] D. Spielman and S. Teng, "Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems," in *STOC*, 2004.

[30] V. Satuluri and S. Parthasarathy, "Scalable graph clustering using stochastic flows: applications to community discovery," in *KDD*, 2009.

[31] F. Bourse, M. Lelarge, and M. Vojnović, "Balanced graph edge partition," in *KDD*, 2014.

[32] I. Stanton and G. Kliot, "Streaming graph partitioning for large distributed graphs." in *KDD*, 2012.

[33] C. Tsourakakis, C. Gkantsidis, B. Radunovic, and M. Vojnovic, "Fennel: streaming graph partitioning for massive scale graphs," in *WSDM*, 2014.

[34] I. Stanton, "Streaming balanced graph partitioning for random graphs," in *SODA*, 2014.

[35] J. Nishimura and J. Ugander, "Restreaming graph partitioning: simple versatile algorithms for advanced balancing." in *KDD*, 2013.

[36] F. Chung and L. Lu, "The average distances in random graphs with given expected degrees," *PNAS*, vol. 99, no. 25, pp. 15 879–15 882, 2002.

[37] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," in *SIGCOMM*, 1999.

[38] R. Albert, H. Jeong, and A. Barabási, "Internet: Diameter of the world-wide web," *Nature*, vol. 401, no. 6749, pp. 130–131, 1999.

[39] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, vol. 69, no. 2, p. 026113, 2004.

[40] T. Yang, R. Jin, Y. Chi, and S. Zhu, "Combining link and content for community detection: a discriminative approach," in *KDD*, 2009.

[41] A. Khadivi, A. A. Rad, and M. Hasler, "Network community-detection enhancement by proper weighting," *Physical Review E*, vol. 83, no. 4, p. 046104, 2011.

[42] M. Ciglan, M. Laclavík, and K. Nørvåg, "On community detection in real-world networks and the importance of degree assortativity," in *KDD*, 2013.

[43] G. Siganos, S. L. Tauro, and M. Faloutsos, "Jellyfish: A conceptual model for the as internet topology," *Communications and Networks, Journal of*, vol. 8, no. 3, pp. 339–350, 2006.

[44] U. Kang and C. Faloutsos, "Beyond'caveman communities': Hubs and spokes for graph compression and mining," in *ICDM*, 2011.

[45] Y. Lim, U. Kang, and C. Faloutsos, "Slashburn: Graph compression and mining beyond caveman communities," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 12, pp. 3077–3089, 2014.

[46] D. Koutra, U. Kang, J. Vreeken, and C. Faloutsos, "Vog: Summarizing and understanding large graphs," in *SDM*, 2014.

[47] U. Kang, J.-Y. Lee, D. Koutra, and C. Faloutsos, "Net-ray: Visualizing and mining billion-scale graphs," in *PAKDD*, 2014.

[48] C. A. R. Hoare, "Algorithm 65: Find," *Communications of the ACM*, vol. 4, no. 7, pp. 321–322, July 1961.

[49] J. Leskovec. http://snap.stanford.edu/snap/.