IMPROVING COARSENING FOR MULTILEVEL PARTITIONING OF COMPLEX NETWORKS

Roland Glantz, Henning Meyerhenke, Peter Sanders, and Christian Schulz

Institute of Theoretical Informatics, Karlsruhe Institute of Technology (KIT)

Introduction

Complex networks such as web graphs or social networks have become a research focus [1]. Such networks have many low-degree nodes and few high-degree nodes. They also have a small diameter, so that the whole network is discovered within a few hops. Various emerging applications produce massive complex networks whose analysis would benefit greatly from parallel processing. Parallel graph algorithms, in turn, often require a suitable network partition, motivating graph partitioning (GP).

Given a graph G = (V, E) with (optional) edge weight function ω and a number of blocks k > 0, the GP problem asks for a partition of V into blocks V_1, \ldots, V_k such that no block is larger than $(1 + \varepsilon) \cdot \lceil \frac{|V|}{k} \rceil$, where $\varepsilon \ge 0$ is the allowed imbalance. When GP is used for parallel processing, each processing element (PE) usually receives one block, and edges running between two blocks model communication between PEs. The most widely used objective function is the *edge cut*, the total weight of the edges between different blocks. To model the communication cost of parallel iterative graph algorithms, the maximum communication volume (MCV) can be more accurate [4]. MCV considers the worst communication volume taken over all blocks V_p $(1 \le p \le k)$ and thus penalizes imbalanced communication: $MCV(V_1, \ldots, V_k) :=$ $\max_p \sum_{v \in V_p} |\{V_i \mid \exists \{u, v\} \in E \text{ with } u \in V_i \neq V_p\}|.$

For solving optimization tasks such as GP on large networks, multilevel methods (consisting of recursive coarsening, initial partitioning, successive prolongation and local improvement) are preferred in practice. Partitioning static meshes this way is fairly mature. Yet, the structure of complex networks challenges current tools. One key issue for most multilevel graph partitioners is coarsening.

Here we present two independent improvements to coarsening. The first one uses the established framework of contracting edges computed as matching. Yet, it defines a new *edge rating* which indicates with non-local information how much sense it makes to contract an edge and thus guides the matching algorithm. The second approach uses cluster-based coarsening and contracts larger sets of nodes into a supernode, yielding fewer levels.

New Coarsening Approaches

Conductance-based Edge Rating. Let the terms *cut* and *cut-set* refer to a 2-partition (C, \overline{C}) of a graph and to the set S(C) of edges running between C and \overline{C} , respectively. The graph clustering measure *conductance* [5] relates the size (or weight) of the cut-set to the volumes of C and \overline{C} . More precisely, $\operatorname{cond}(G) := \min_{C \subset V} \frac{|S(C)|}{\min\{\operatorname{vol}(C), \operatorname{vol}(\overline{C})\}}$, where the volume $\operatorname{vol}(X)$ of a set X sums over the (weighted) degrees of the nodes in X.

An edge rating in a multilevel graph partitioner should yield a low rating for an edge e if e is likely to be contained in the cut-set of a "good" cut. In our approach a good cut is one that has low conductance and is thus at least moderately balanced. A loose connection between conductance and MCV can be established via isoperimetric graph partitioning [3]. Our approach to coarsen a graph with a new edge rating is as follows. (i) Generate a collection \mathcal{C} of moderately balanced cuts of G with a low conductance value. (ii) Define a measure $Cond(\cdot)$ such that Cond(e)is low [high] if e is [not] contained in the cut-set of a cut in \mathcal{C} with low conductance. (iii) Use the new edge rat- $\operatorname{ing} \operatorname{ex_cond}(\{u, v\}) = \omega(\{u, v\}) \operatorname{Cond}(\{u, v\}) / (c(u)c(v))$ as weights for an approximate maximum weight matching algorithm \mathcal{A} , where c(x) refers to the weight of node x. The higher $ex_cond(e)$, the higher the chances for e to be contracted. (iv) Run \mathcal{A} and contract the edges returned in the matching.

We arrive at a collection C of |V|-1 moderately balanced cuts of G by (i) computing connectivity-based "contrast" values for the edges of G, (ii) computing a minimum spanning tree T^m of G w.r.t. these values, and (iii) letting C consist of G's fundamental cuts w.r.t. T^m . We want the contrast value $\gamma(e)$ of an edge e to be high if e is part of "many" connections via shortest paths in G. Based on a collection \mathcal{T} of rooted spanning trees of G, this means that (i) e is contained in many trees from \mathcal{T} and (ii) e is not involved in small cuts that separate a small subgraph of G from G's "main body". We achieve this by setting $\gamma(\{u, v\}) = \min\{n_{\mathcal{T}}(u, v), n_{\mathcal{T}}(v, u)\}$, where $n_{\mathcal{T}}(u, v)$ denotes the number of trees in \mathcal{T} containing e such that u is closer to the tree's root than v. Cond(\cdot) is finally defined such that $\operatorname{Cond}(e)$ is low [high] if e is [not] contained in the cut-set of a cut in \mathcal{C} with low conductance: $\operatorname{Cond}(e) = \min_{C \in \mathcal{C}, e \in S(C)} (\operatorname{cond}(C)).$

Cluster-based Coarsening. As an alternative approach to coarsening networks with a highly irregular structure, we propose a more aggressive coarsening algorithm that contracts size-constrained clusterings computed by a label propagation algorithm (LPA). LPA was originally proposed by Raghavan et al. [7] for graph clustering. It is a fast, near-linear time algorithm that locally optimizes the number of edges cut. Initially, each node is in its own cluster/block. In each of the subsequent rounds, the nodes of the graph are traversed in a random order. When a node v is visited, it is moved to the block that has the strongest connection to v (with some tie-breaking mechanism). The original process is repeated until convergence, each round takes $\mathcal{O}(n+m)$ time. Here, we perform at most ℓ iterations of the algorithm, where ℓ is a tuning parameter, and stop the algorithm if less then 5% of the nodes changed its cluster during one round. Hence, we do not face the occasional instabilities of the original algorithm. Most importantly, we adapt LPA such that clusters cannot grow beyond a certain size. This is done to respect the imbalance criterion of GP.

We integrate further algorithmic extensions such as modified iterations over the node set within LPA, ensemble clusterings, and iterated multilevel schemes. They are described in more detail in the corresponding full paper.

To compute a graph hierarchy, the clustering is contracted by replacing each cluster with a single node, and the process is repeated recursively until the graph is small.

Here we aim at partitioning for low edge cuts with this method. The intuition for achieving this goal is that a good clustering contains only few edges between clusters.

Implementation and Experimental Results

Experimental results have been obtained by implementing our new methods within the framework of the state-ofthe-art graph partitioner KAHIP [9].

Conductance-based Edge Rating. KAHIP contains a reference implementation of the edge rating $ex_alg(\cdot)$, which yielded the best quality for complex networks so far [8]. In addition to our new edge rating $ex_cond(\cdot)$, we have integrated a greedy postprocessing step that trades in small edge cuts for small MCVs into KAHIP. Our experiments show that greedy MCV postprocessing alone

improves the partitions of our complex network benchmark set in terms of MCV by about 11% with a comparable running time for both ex_alg(·) and ex_cond(·). Additional bipartitioning experiments (MCV postprocessing included) show that, compared to ex_alg(·), the fastest variant of our new edge rating further improves the MCVs by 10.3%, at the expense of an increase in running time by a factor of 1.8. Altogether, compared to previous work on partitioning complex networks with state-of-the-art methods [8], the total reduction of MCV amounts to 20.4%.

Cluster-based Coarsening. For the second set of experiments, KAHIP uses the hierarchy computed by clusterbased coarsening and its own initial partitioning as well as existing local search algorithms for refinement on each level, respectively. Some algorithm configurations also use the size-constrained LPA as local search procedure. We compare against the established tools kMetis, hMetis, and Scotch, all in graph partitioning mode.

Depending on the algorithm's configuration, we are able to compute the best solutions in terms of edge cut or partitions that are comparable to the best competitor in terms of quality, hMetis, while being nearly an order of magnitude faster on average. The fastest configuration partitions a web graph with 3.3 billion edges using a single machine in about ten minutes while cutting less than half of the edges than the fastest competitor, kMetis.

Accompanying Publications. Details can be found in the respective papers [2, 6] and their full arXiv versions.

References

- L. F. Costa, O. N. Oliveira Jr, G. Travieso, F. A. Rodrigues, P. R. V. Boas, L. Antiqueira, M. P. Viana, and L. E. C. Rocha. Analyzing and Modeling Real-World Phenomena with Complex Networks: A Survey of Applications. Adv. in Physics, 60(3):329–412, 2011.
- [2] R. Glantz, H. Meyerhenke, and C. Schulz. Tree-based coarsening and partitioning of complex networks. In Proc. 13th Intl. Symp. on Experimental Algorithms (SEA 2014), pages 364–375, 2014.
- [3] L. Grady and E. L. Schwartz. Isoperimetric graph partitioning for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(3):469–475, 2006.
- [4] B. Hendrickson and T. G. Kolda. Graph partitioning models for parallel computing. *Parallel Computing*, 26(12):1519–1534, 2000.
- [5] R. Kannan, S. Vempala, and A. Vetta. On clusterings: Good, bad and spectral. *Journal of the ACM*, 51(3):497–515, 2004.
- [6] H. Meyerhenke, P. Sanders, and C. Schulz. Partitioning complex networks via size-constrained clustering. In Proc. 13th Intl. Symp. on Experimental Algorithms (SEA 2014), pages 351-363, 2014.
- [7] U. N. Raghavan, R. Albert, and S. Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, 76(3):036106, 2007.
- [8] I. Safro, P. Sanders, and C. Schulz. Advanced coarsening schemes for graph partitioning. In Proc. 11th Int. Symp. on Experimental Algorithms, pages 369–380. Springer, 2012.
- [9] P. Sanders and C. Schulz. Think Locally, Act Globally: Highly Balanced Graph Partitioning. In *Proceedings of the 12th International* Symposium on Experimental Algorithms (SEA'13), LNCS, pages 164–175. Springer, 2013.