Abstract

Given a graph, the b-Matching problem is to find an edge weighted matching of maximum weight with the constraint that every vertex v can match with at most b vertices. b-Matching is useful in various machine learning problems such as classification, spectral clustering, graph sparsification, graph embedding and data privacy. The exact algorithms for this problem have high time as well as space complexities, are inherently sequential, and therefore, are not practical on large problems. We propose a 1/2-approximation algorithm, we call it bSuitor, which runs in linear time in the number of edges and also requires linear storage. We show that our algorithm can solve large problems with billions of edges and can get up to 97% of weight of the optimal solution. We also show that our algorithm scales up to 11x on 16 cores of Intel Xeon machines and up to 50x on 60 cores of Intel Xeon Phi machines.

Experiments and Results

Figure 1: Quality of the Approximation

Figure 2: Relative runtimes with other algorithms

Table 1: Comparing single thread run times of k-Anonymity problem using exact b-Matching and bSuitor.

Table 2: Comparing the run times (seconds) of bSuitor based k-Anonymity algorithm with large problems.

Motivation

The fastest exact algorithms for maximum edge weighted b-Matching have the time complexity of $O(|V|^{1+1/|E|})$. Therefore, it is not practical to use these algorithms to solve larger problems. It turns out that b-Matching has practical use in many machine learning applications where approximate solutions suffice. Therefore, any good approximation algorithm can be used instead of exact algorithms. This approximation also has the benefit of being highly scalable nature. These are several applications where b-Matching is shown to be useful: 1) Classification 2) Spectral clustering (3) Graph embedding (4) graph sparsification and (5) Data privacy xi in k-Anonymity problem.

Application to k-Anonymity Privacy Problem

- We apply our algorithm to solve the k-Anonymity privacy problem.
- We show that by using approximate matching instead of exact matching makes the algorithm faster by two order of magnitude (Table 1).
- We reduce the overall memory complexity of k-Anonymity problem from quadratic to linear in number of data points by using partially sorted adjacency lists in bSuitor.
- This enables us to solve k-Anonymity problems that are two orders of magnitude larger than previously reported.

Contributions and Future Work

- We have shown that the bSuitor algorithm is the fastest algorithm for approximate b-Matching compared to other algorithms. We also show that this algorithm demonstrates near linear scalability both on Xeon and Xeon Phi multiprocessors.
- We identified an important application of bSuitor to privacy problem called k-Anonymity.
- By using bSuitor, we can solve problems with sizes larger than a factor of 100, which could not be solved before without significant change in the quality of the solution.
- Our goal is to continue developing faster b-Matching algorithms.
- We also plan to apply our algorithm to other contexts such as graph clustering and partitioning.

References


Acknowledgements

We also acknowledge the support of Fredrik Manne, M.T. Montaño, A. Patwary, Nadathur Satish and Narayan Sundaram. For our experiments we used Purdue Community Cluster Conte. Each compute node contains two Intel® Xeon® ES-2670 processors running at 2.60 GHz (16 cores in all). Each node also has an Intel® Xeon® Phi™ coprocessor running at 1.08 GHz (60 cores in all). Intel, Xeon, and Intel Xeon Phi are trademarks of Intel Corporation in the U.S. and/or other countries.