Centre for High Performance Computing 2021 National Conference

Contribution ID: **50** Type: **Student Micro-talk**

Towards Scalable Comparison of Large Scale Graphs

Keywords: Large Scale Graphs, Graph Comparison, Shared Memory Implementation

Authors: Krishna Sai Ujwal Kambhumpati(1), Patrick Bell(2)), Michela Taufer(2)
, and Sanjukta Bhowmick(1)
Affiliations:
1. The University of North Texas
2. The University of Tennessee, Knoxville
Corresponding Email Address: krishnasaiujwalkambhumpati@my.unt.edu

Comparing the structure of a pair or a group of graphs is a fundamental problem in graph theory, and has many applications ranging from bioinformatics, cybersecurity and recently in assessing non-determinism in HPC simulations [1,2]. There exist several approximate algorithms and heuristics to compare graphs, based on different metrics of similarity. Among them, one of the popular approaches is to compare motifs (or graphlets) in each graph.

Motifs are subgraphs of small size. Since the cost of computing motifs increases exponentially as the size increases, typically the size of the motif is restricted to five vertices. Each motif is associated with a set of unique automorphism orbits, that denote the position of the vertices within the structure of the motif. As presented in [3], each vertex is associated with a graphlet degree distribution vector(GDV). A GDV of a vertex is the number of times that the vertex is associated an automorphism orbit. Vertices with similar GDVs, tend to have the same local structure. Given a pair of graphs, the algorithm aims to align similar vertices from each graph, to evaluate the similarity between them.

Existing approaches of GDV computation does not scale to large graphs. First, the larger the graph, the more motifs have to be computed. Second, due to the automorphism of many small motifs, several redundant and overlapping computations occur. We propose a novel scalable algorithm to address these challenges by representing motif computation as tree traversal and leveraging the vertex ids to eliminate redundant computations.

In our algorithm, a vertex is designated as a source node. The graph is traversed from this source node to create all possible trees of size five or less, with the constraint that a leaf node in a tree cannot have an id that is lower than the source node. This ensures that trees generated from a source node are unique to that node only. Each tree is a motif and the positions of the vertices in the motif are computed to update their respective GDVs. In the next stage, back edges are added to the trees, with the condition that the back edge connects the vertices with the two lowest ids in the cycle. Again, this condition ensures that motifs with cycles are formed exactly once, and not repeated.

Our proposed algorithm has several advantages. First, our algorithm eliminates redundant computations by ensuring each motif is seen exactly once. Second, the computations can be easily parallelized per vertex leading to scalable executions. Third, our method allows us to prioritize which vertices to process earlier, based on vertex based properties, such as degree distribution. Finally, our method is flexible enough to find motifs in attributed as well as non-attributed graphs. So far most scalable algorithms are designed either for strictly attributed graphs (GraphPI [4]) or non-attributed graphs (PruneJuice [5]). Our method is a more generalized approach. In this presentation, we will provide an overview of our algorithm, along with strong scalability results on large graphs for shred memory systems, and demonstrate how our method can be used to compare across event graphs arising from large HPC simulations.

References.

[1] D. Chapp, N. Tan, S. Bhowmick and M. Taufer, "Identifying Degree and Sources of Non-Determinism in MPI Applications Via Graph Kernels," in IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 12, pp. 2936-2952, 1 Dec. 2021, doi: 10.1109/TPDS.2021.3081530.

[2] Patrick Bell, Kae Suarez, Dylan Chapp, Nigel Tan, Sanjukta Bhowmick,and Michela Taufer. Anacin-x: A software framework for studying non-determinism in mpi applications.Software Impacts, 10:100151, 2021.

[3] O. Kuchaiev, T. Milenkovic, V. Memisevic, W. Hayes, and, N. Przulj, "Topological network alignment uncovers biological function and phylogeny". J R Soc Interface. 2010;7(50):1341-1354. doi:10.1098/rsif.2010.0063

[4] T. Shi, M. Zhai, Y. Xu and J. Zhai, "GraphPi: High Performance Graph Pattern Matching through Effective Redundancy Elimination," SC20: International Conference for High Performance Computing, Networking, Storage and Analysis, 2020, pp. 1-14, doi: 10.1109/SC41405.2020.00104.

[5] T. Reza, M. Ripeanu, N. Tripoul, G. Sanders and R. Pearce, "PruneJuice: Pruning Trillion-edge Graphs to a Precise Pattern-Matching Solution," SC18: International Conference for High Performance Computing, Networking, Storage and Analysis, 2018, pp. 265-281, doi: 10.1109/SC.2018.00024.

Student?

Yes

Supervisor name

Dr.Sanjukta Bhowmick

Supervisor email

sanjukta.bhowmick@unt.edu

Primary authors: Mr KAMBHUMPATI, Krishna Sai Ujwal (University Of North Texas); Dr TAUFER, Michela (University Of Tennessee Knoxville); Mr BELL, Patrick (University Of Tennessee Knoxville); Dr BHOWMICK, Sanjukta (University Of North Texas)

Presenter: Mr KAMBHUMPATI, Krishna Sai Ujwal (University Of North Texas)

Session Classification: Micro-talks

Track Classification: HPC Techniques and Computer Science