Parallel Scalable Algorithms for Updating Dynamic Networks

Sriram Srinivasan, Graduate Student, Computer Science

Faculty Mentor: Sanjukta Bhowmick, Computer Science

The growth of social media increased the interest in analyzing network algorithms. The networks are highly unstructured and exhibit poor locality, which has been a challenge for developing scalable parallel algorithms. The state-of-the-art network algorithms such as Prim's algorithm for Minimum Spanning Tree, Dijkstra's algorithm for Single Source Shortest Path and iSpan algorithm for detecting strongly connected components are designed and optimized for static networks. The networks which change with time i.e. the dynamic networks such as social networks, the above-mentioned approaches can only be utilized if they are recomputed from scratch each time. Performing a re-computation from scratch for a significant amount of changes is not only computationally expensive, however, increases the memory footprint and the execution time. In the case of dynamic networks, developing scalable parallel algorithms is very challenging and there has been a very limited amount of research work that has been performed when compared to developing parallel scalable algorithms for static networks.

To address the above challenges, this research proposes a new high performance, scalable, portable, open source software package and an efficient network data structure to update the dynamic networks on the fly. This approach is different from the naive approach which is the re-computation from scratch and is scalable for random, small-world, scale-free, real-world and synthetic networks. The software package currently is implemented on a shared memory system and updates network properties such as Connected Components (CC), Minimum Spanning Tree (MST), Single Source Shortest Path (SSSP), and Strongly Connected Components(SCC). The key attributes of the software are faster insertions and deletions. Additionally, the software takes less time and memory for updating the networks when compared to the state of the art Galois. The shared memory implementation processes over 50 million updates on a real-world network under 30 seconds.