Michael Svendsen
CprE
MS
Tirithapura, Srikanta

Mining maximal cliques from large graphs using MapReduce

Maximal clique enumeration (MCE), a fundamental task in graph analysis, can help us identify dense substructures within a graph, and has found applications in graphs arising in biological and chemical networks, and more. While MCE is well studied in the sequential case, a single machine can no longer process large graphs arising in today's applications, and effective ways are needed for processing these in parallel.

This work introduces PECO (Parallel Enumeration of Cliques using Ordering); a novel parallel MCE algorithm. Unlike previous works, which require a post-processing step to remove duplicate and non-maximal cliques, PECO enumerates only maximal cliques with no duplicates. This is achieved by inducing a strict ordering over the vertices. Then, as a subgraph is processed, a clique is only enumerated if the clique satisfies a certain condition with respect to the subgraph, and this ensures that each maximal clique is output exactly once by the algorithm. It is well recognized that in enumerating maximal cliques, the sizes of different subproblems can be non-uniform, and load balancing among the subproblems is a significant issue. Our algorithm uses the above vertex ordering technique to greatly improve load balancing when compared with straightforward approaches to parallelization. PECO has been designed and implemented for the MapReduce framework, but this technique is applicable to other parallel frameworks too.

Our experiments on a variety of large real world graphs using several ordering strategies show that PECO can enumerate cliques in large graphs of well over a million vertices and tens of millions of edges, and that it scales well to at least 64. A comparison of ordering strategies shows that an ordering based on the degrees of vertices performs the best.