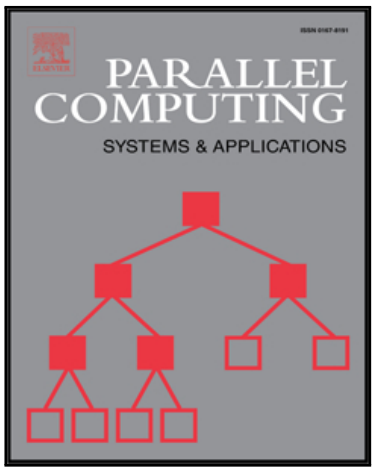

A Study of Graph Partitioning Schemes for Parallel Graph Community Detection

Jianping Zeng, Hongfeng Yu

PII: S0167-8191(16)30040-0
DOI: 10.1016/j.parco.2016.05.008
Reference: PARCO 2328

To appear in: *Parallel Computing*

Received date: 2 December 2015
Revised date: 6 May 2016
Accepted date: 15 May 2016

Please cite this article as: Jianping Zeng, Hongfeng Yu, A Study of Graph Partitioning Schemes for Parallel Graph Community Detection, *Parallel Computing* (2016), doi: 10.1016/j.parco.2016.05.008

A Study of Graph Partitioning Schemes for Parallel Graph Community Detection

Jianping Zeng, Hongfeng Yu

University of Nebraska-Lincoln

Abstract

This paper presents a study of graph partitioning schemes for parallel graph community detection on distributed memory machines. We investigate the relationship between graph structure and parallel clustering effectiveness, and develop a heuristic partitioning algorithm suitable for modularity-based algorithms. We demonstrate the accuracy and scalability of our approach using several real-world large graph datasets compared with state-of-the-art parallel algorithms on the Cray XK7 supercomputer at Oak Ridge National Laboratory. Given the ubiquitous graph model, we expect this high-performance solution will help lead to new insights in numerous fields.

Keywords: Large graph, Community detection, Graph clustering, Parallel and distributed processing

1. Introduction

Community detection, also named as graph clustering, is a powerful technique for researchers to explore hidden patterns existing in graphs. However, it is still an open problem to design a scalable and accurate parallel community detection algorithm to tackle large graphs using a parallel machine with distributed memory. This is mainly due to the challenges in graph partitioning. First, traditional graph partitioning schemes cannot well preserve the global structure information of a graph on a processor, which can lower the cluster quality of local community detection and impair the accuracy of final aggregated results. Second, for real-world graphs (in particular scale-free graphs), it is difficult to create a balanced edge partitioning. Traditional 1D partitioning schemes often assign all the incident edges of one vertex to one processor, which can incur severe workload imbalance among processors and impair the scalability of parallel community detection. On