Invitation à la soutenance publique de thèse

Pour l'obtention du grade de Docteur en Sciences de l’Ingénieur

Mademoiselle Vasiliki KALAVRI
Erasmus Mundus Doctorante in Distributed Computing

Performance Optimization Techniques and Tools for Distributed Graph Processing

Distributed, shared-nothing architectures of commodity machines are a popular design choice for the implementation and deployment of big data platforms. The introduction of MapReduce, a simple programming model for parallel data analysis, has greatly simplified data-parallel programming by abstracting the details of data partitioning, node communication, and fault tolerance. While MapReduce is a powerful model for simple tasks, such as text processing and web log analysis, it is a poor fit for more complex tasks, such as graph analysis. Inter-connected data that can be modeled as graphs appear in several application domains, including machine learning, recommendation, web search, and social network analysis. Graph algorithms are very diverse and expose various computation and communication patterns.

MapReduce revolutionized the area of distributed data processing and inspired the development of similar high-level graph processing models and platforms. However, as graphs grow bigger, delivering high performance for graph analysis tasks becomes challenging. Existing distributed graph processing platforms often deliver disappointing performance, while demanding expensive resources, as compared to sequential or multi-threaded algorithms running on a single machine. Processing graphs on a single machine is often not a viable solution. First of all, graphs rarely appear as raw data. Instead, they are derived from processing, filtering, and transformation of other, often distributed, data sources. Second, graph analysis tasks are usually part of a larger data analysis pipeline. Thus, previous and succeeding processing might require distribution over several machines. Finally, the nature of the graph analysis problem might require distribution.

Thus, developing optimization techniques and tools to improve the performance of distributed graph processing platforms is essential. In this thesis, we propose optimization techniques for distributed graph processing on general-purpose data processing engines and on specialized graph processors. Our optimizations leverage both data and algorithmic properties. Driven by a real-world graph problem, we design performance optimization techniques and tools. First, we describe a data processing pipeline that leverages an iterative graph algorithm for automatic classification of web trackers. Using this application as a motivating example, we examine how *asymmetrical convergence* of iterative graph algorithms can be used to reduce the amount of computation and communication in large-scale graph analysis. We propose an optimization framework for fixpoint algorithms and a declarative API for writing fixpoint applications. Our framework uses a cost model to automatically exploit asymmetrical convergence and evaluate execution strategies during runtime. We show that our cost model achieves speedup of up to 1.7x and communication savings of up to 54%. Next, we propose to use the concepts of *semi-metricity* and the *metric backbone* to reduce the amount of data that needs to be processed in large-scale graph analysis. We provide a distributed algorithm for computing the metric backbone using the vertex-centric programming model. Using the backbone, we can reduce graph sizes up to 88% and achieve speedup of up to 6.7x.

Membres du jury :
Prof. Peter Van Roy (UCL), promoteur
Prof. Vladimir Vlassov (KTH, Sweden), promoteur
Prof. Olivier Bonaventure (UCL), président
Prof. Ramin Sadre (UCL), secrétaire
Prof. Seif Haridi (KTH, Sweden)
Prof. Siegfried Nijssen (UCL)

Lundi 10 octobre 2016 à 13h
KTH Electrum
Hall C
Kista
Sweden

A video conference will be organized in Bibliothèque des Sciences et technologies (BST), visio room