Abstract

Contemporary technological advancements, besides affecting all facets of human society in a multitude of ways, often demand the aggregated processing of ever-increasing amounts of data. This is often achieved with the use of *computer clusters*, a practice popularised by industry behemoths such as *Google* and *Amazon*. The *Cloud Computing* era, reminiscent of the older one of the *mainframes* but differentiated by the startling expansion of the Internet, is associated with the distant harnessing of this computing power and remote storage space, turning Cloud Computing into a viable solution for a variety of applications. Simultaneously, economic and ecological reasons dictate that this harnessing should be achieved in an efficient and cost-effective way; the computational resources are finite and the *Big Data* field, especially, requires the harmonious coordination of numerous processing nodes.

From the programmer's perspective, parallel processing –in its various forms– is notoriously difficult to get right and reason about. In order to surpass the inherent difficulties of the domain and use the aforementioned infrastructure with optimal efficiency, reliability and ease, a variety of *parallel processing models* have been proposed and implemented over time. These models often differ in their purpose and their level of abstraction. In this work, we present a brief overview of certain promising ones, mostly oriented towards Big Data processing in cluster environments. We focus on Valiant's abstract *Bulk Synchronous Parallel* model and on the technical aspects of *Apache Giraph*, one of its open-source implementations and member of the wider *Apache Hadoop* software ecosystem. As a proof-of-concept, we attempt to lay the foundations for an implementation of *SybilRank* on the programming environment of Giraph; SybilRank is an efficient algorithm for the detection of "*Sybil*" nodes in large *Online Social Networks*. Finally, we describe the deployment process of the resulting program on the *Amazon EC2* cluster and attempt to draw conclusions from our experience.