Algorithmic issues in MapReduce

MapReduce is a standard programming framework for processing large-scale distributed data introduced 10 years ago by Google. In the map phase, the input data of an application are partitioned and assigned into a set of execution units which can run in parallel. Each execution unit produces an intermediate result. In the reduce phase, the intermediate results are combined in order to obtain the final result. Note that, in general, the reduce phase cannot start before all intermediate results are completely produced. A widely-used implementation of this framework is Apache Hadoop.

As an example, consider the application of counting the number of occurrences of words of a big-sized document. In the map phase, different parts of the document are assigned into different processing units. Each processing unit treats its assigned text and generates a set of pairs <word, number of occurrences> that correspond to the words appeared in this assigned text. In the reduce phase, the produced results are sent to a (usually) smaller set of processing units (the reducers) which produce the final result for the initial document.

Several optimization problems arise concerning the allocation [1,3] of the (map and reduce) tasks as well as the scheduling [2,3] of these tasks. The objective of this internship is to focus mainly on the allocation problems. More specifically, we aim to design algorithms in order to allocate the results of the map phase to a set of reducers such that to optimize several criteria, like the number of reducers or the communication cost of the transmission of the intermediate result to the reducers, etc. We are interested in both algorithms with proven approximation guarantees but also in fast/greedy algorithms and practical implementation.

The candidate should have a background in classical tools for analyzing algorithms. Programming skills are also appreciated.

The internship will take place at either the INRIA-Grenoble center (possibility of residence funding can be negotiated), or at LIP6.

Contact
Denis Trystram, LIG, http://moais.imag.fr/membres/denis.trystram/

Bibliography