Automatic Performance Estimation for Decentralized Optimization

The goal of my thesis is to boost the research in decentralized optimization by creating tools allowing one to automatically compute the worst-case performance of any decentralized optimization algorithm***,*** and to identify the bottleneck instances, providing insight on what limits the performance. This will help develop a better understanding of the decentralized optimization algorithms and allow for rapid exploration of ideas and their iterative improvement. Before going into details, let's first introduce what optimization is.

Optimization algorithms are explicit iterative steps executed by a computer to find a point that minimizes the value of a given function. They are used everywhere in applied mathematics, where they help minimize consumption or cost, or maximize impact or profit, in a large variety of applications across the sciences, engineering, industry and business. In machine learning for example, an optimization algorithm finds the parameters of the predictive model that minimizes the sum of errors produced by each data.

Generally, the optimization methods start with a point and update it iteratively in a direction where the function decreases. To work efficiently, the direction and the step-size to be executed in this direction should be chosen carefully.

In many situations, it is not always possible or desirable that the entire objective function is stored in a unique computer. In machine learning, we can imagine situations where multiple computers each hold a part of the data and that they want to collaborate to find the best model describing all the data distributed across computers but without needing to centralize them in one place. This is useful for reasons of efficiency (several computers are working at the same time), data privacy (the data can be used without sharing it), storage or transmission capacity (if you have a large amount of data, it might be impossible to store or send everything on a single computer). In that purpose, the different computers, or agents, should perform iterative steps combining local computations and communications in order to reach an agreement about the optimal solution for the global problem. This is known as ***decentralized optimization***.

Experience has shown in many cases that algorithms that perform the best in practice are those for which we possess a good theoretical understanding. In particular, one usually seeks to analyze the worst-case behavior of a given algorithm on a typical class of input problems.

Once identified and formally proved, this worst-case provides strong guarantees on the performance of the algorithm, such as an upper bound on the error that will be observed after a certain number of iterations, or the maximum distance between the final iterate and the optimal solution. Obtaining those worst-case performances for decentralized algorithms through theoretical analysis can often be a challenging task, requiring to combine the effects of the optimization and of the interconnection network, and leaving a lot of room for improvement in the obtained bounds. However, having access to precise bounds on the performances of the different algorithms is crucial for correctly tuning their parameters (such as the step-size) and correctly comparing them to each other.

In this work, we consider an alternative approach, known as Performance Estimation Problem (PEP), that consists ***in posing the worst-case performance as an optimization problem itself***, where we search for the specific problem that leads the algorithm to the maximum error after k iterations. The foundations of this approach were laid in a thesis at UCLouvain and allow for a complete automation of the process of analyzing optimization algorithms. This led to a variety of new and sometimes surprising results in the field of optimization. Motivated by the success of this approach for classical centralized optimization, we have just developed a PEP formulation allowing to automatically analyze the ***worst-case performance of decentralized optimization methods***, including the identification of the worst communication network for an algorithm. We have already used our new PEP formulation for analyzing one of the simplest decentralized optimization methods, which allows us to largely improve the existing theoretical performance guarantees. Future work can apply our new spectral formulation to many different decentralized methods in order to improve their worst-case guarantees and their parameters tuning. Eventually, this work can also help in the creation of new decentralized methods.