What are the Optimal Communication Weights for Decentralized Optimization ?

Sebastien Colla[1] and Julien M. Hendrickx
ICTEAM Institute, UCLouvain, 1348 Louvain-la-Neuve, Belgium.
Email: { sebastien.colla, julien.hendrickx } @uclouvain.be

1 Introduction

We consider a decentralized optimization problem in which a set of agents $\{1,\dots,N\}$ collaborate to minimize the average of their private local functions $f_i : \mathbb{R}^d \to \mathbb{R}$:

$$\underset{x \in \mathbb{R}^d}{\text{minimize}} \quad f(x) = \frac{1}{N}\sum_{i=1}^{N} f_i(x).$$

Each agent i holds a local copy x_i of the decision variable to performs local computations. The agents exchange local information with their neighbors to come to an agreement on the minimizer x^* of the global function f. These exchanges often take the form of an average consensus on some quantity, e.g., on the x_i. The consensus step can be represented using a multiplication by an averaging matrix $W \in \mathbb{R}^{N\times N}$, for which $W_{ij} \neq 0$ only when there is a communication link between i and j. We will focus on the case where the matrix W is symmetric and doubly-stochastic, which is required for the convergence of many decentralized algorithms. In general, the performance of a decentralized optimization method is largely impacted by the averaging matrix W. While the zero elements are imposed by the network topology, the values of the non-zero elements should be carefully determined to obtain efficient algorithms, as for any other parameter of a method.

2 State of the Art in Averaging Consensus

For the pure averaging consensus $\mathbf{x}^{k+1} = W\mathbf{x}^k$, it has been shown in [1] that the symmetric matrix W leading to the smallest per-step convergence factor is the doubly-stochastic matrix with the smallest *Second Largest Eigenvalue Magnitude* (SLEM), denoted W_{λ_2}. Obtaining such a matrix requires (at least) one agent to have an entire knowledge of the communication network. Another choice for W, based only on local degree information, is to choose the Metropolis weights, denoted W_M: for any edge between i and j ($i \neq j$),

$$W_{ij} = \frac{1}{\max\{d_i, d_j\}},$$

where d_i and d_j are the degree of agents i and j. The diagonal weights W_{ii} are chosen so that W_M is doubly stochastic. The Metropolis weights ensure convergence of the consensus to the average, but often at a slower rate.

The next section shows that W_{λ_2} is not especially the best choice for decentralized optimization algorithms, where the consensus is constantly perturbed by local updates based on the local gradients.

[1] S. Colla is supported by the French Community of Belgium through a FRIA fellowship (F.R.S.-FNRS).

3 Results for Decentralized Optimization

To compute the optimal weights for a given algorithm, a given class of function and a given performance measure, we rely on the Performance Estimation (PEP) framework, allowing to compute numerically the exact worst-case performance for a given averaging matrix W [2]. We therefore have a numerical function *Perf* that computes the algorithm's performance depending on the problem settings (class of functions, network topology, etc) and the algorithm parameters (step-sizes and averaging weights). A priori, this function is non-smooth and non-convex in the parameters. In this work, we use a zero-order method (pattern search) to find the parameter values leading to the best worst-case performance, for the given problem settings. We therefore identify, the optimal weights W, but also the other parameters of the method, e.g. the step-size, since the optimal parameter values are often interdependent.

To reduce the computational load, we only look for averaging matrices with equal weights for every edge. We expect better results when allowing different weights. In our preliminary results (see Table 1), we observe that the optimal matrix W_* does not generally match W_{λ_2} nor W_M, and allows the algorithms to work with larger step-sizes and improve the performance up to 40 % for 10 iterations. This means that λ_2, which is used in every theoretical bound, is not the best characterization of the network we can have for decentralized optimization. As future work, we would like to explore which network characterizations are more suitable.

Topology	complete			star			cycle		
Weights	W_*	W_{λ_2}	W_M	W_*	W_{λ_2}	W_M	W_*	W_{λ_2}	W_M
DIGing	0.24	0.40	0.40	0.64	0.86	0.73	0.41	0.67	0.67
EXTRA	0.25	0.25	0.25	0.44	0.45	0.58	0.32	0.33	0.33
DNGD	0.24	0.27	0.27	0.50	0.63	0.75	0.32	0.47	0.47

Table 1: Worst-case performance of 10 iterations of different decentralized optimization algorithms, for different weighting matrices and different network topologies. The network size is fixed to $N = 4$ agents. Local functions are 1-smooth and 0.1-strongly-convex.

References

[1] L. Xiao, S. Boyd, "Fast linear iterations for distributed averaging", *Control & System Letters*, 2004.

[2] S. Colla, J. M. Hendrickx, "Automatic Performance Estimation for Decentralized Optimization", *TAC*, 2023.