Postdoc position

Asynchronous Approximate Distributed Computation for Machine Learning

Goal

A basic step in machine learning is to compute parameter models using some stochastic gradient method in a distributed way. The dataset is split among different executors, each using its data subset to compute a noisy estimation of the gradient of the loss function. The gradient estimates are then averaged during a synchronization phase and this improved estimate is used to update the parameter models. The synchronization phase is time consuming also because of stragglers, so that asynchronous solutions have been proposed, where nodes may work on stale data and read-write race conditions may arise. Empirically, on single-node systems, these asynchronous algorithms have yielded order-of-magnitude improvements in performance (see e.g. [Feng12]). While a significant performance improvement can be obtained, convergence is not guaranteed, and a faster computational throughput does not necessarily guarantee a faster convergence [Kadav16]. In particular, faster nodes can bias the convergence.

Large-scale data-parallel computation frameworks, like Spark, usually rely on the bulk synchronous parallel model with high synchronization overhead. [Gonzalez15] shows how asynchronous primitives can be introduced in Spark in order to implement stochastic gradient or alternating direction method of multipliers, but convergence is not guaranteed. [Kadav16] proposes another form of fine-grained synchronization, where each executor only depends on a few other executors, in the sense that requires only their state updates to be able to proceed in the computation. This partial dependency reduces communication overhead and may mitigate the effect of stragglers, but may also increase the number of iterations required for convergence.

Our goal is to propose and evaluate the performance of distributed asynchronous optimization algorithms for large-scale computation frameworks and in particular for Spark. There are a number of open research directions. First, as we mentioned, convergence to the optimum is not guaranteed for many of the proposed solutions. Second, performance models quantifying convergence speed improvement from asynchronous approaches are missing. Finally, when partial dependencies among executors can be introduced as in [Kadav16], it is not clear what is the optimal dependency graph.

The candidate is invited to work on (a subset of) these research directions.

Candidate's profile

We are interested in two possible profiles:

1) a candidate with strong theoretical background on distributed optimization and probability.
2) a candidate with strong hands-on experience in operating with Spark as well as modifying its code.

Monthly Salary

2653 € (gross with medical insurance included), the net salary is about 2130 €.

Contact person

Giovanni Neglia, giovanni.neglia@inria.fr, Inria, Sophia Antipolis Méditerranée, France
http://www-sop.inria.fr/members/Giovanni.Neglia/

How to apply

The candidate should apply online at the following address https://goo.gl/YoAlnL, but he/she is invited to send his/her cv directly to Giovanni Neglia.

References

[Feng12] X. Feng, A. Kumar, B. Recht, and C. Ré. Towards a unified architecture for in-RDBMS analytics. In SIGMOD, 2012.

[Gonzalez11] J. Gonzalez, Y. Low, A. Gretton, and C. Guestrin, "Parallel gibbs sampling: From colored fields to thin junction trees," in In Artificial Intelligence and Statistics (AISTATS), Ft. Lauderdale, FL, May 2011.

[Gonzalez15] J. E. Gonzalez, P. Bailis, M. I. Jordan, M. J. Franklin, J. M. Hellerstein, A. Ghodsi, and I. Stoica, "Asynchronous complex analytics in a distributed dataflow architecture," arXiv preprint arXiv:1510.07092, 2015.

[Kadav16] A. Kadav and E. Kruus, "Asap: Asynchronous approximate data-parallel computation," arXiv preprint arXiv:1612.08608, 2016.

[Recht11] B. Recht, C. Ré, S. Wright, and F. Niu. Hogwild: A lock-free approach to parallelizing stochastic gradient descent. In NIPS, 2011.

[Zhang13] C. Zhang and C. R´e, "Towards high-throughput gibbs sampling at scale: A study across storage managers," in Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data, ser. SIGMOD '13. New York, NY, USA: ACM, 2013, pp. 397–408. [Online]. Available: http://doi.acm.org/10.1145/2463676.2463702