A zealous parallel gradient descent algorithm

Gilles Louppe and Pierre Geurts
Department of EE and CS, University of Liège, Belgium

Abstract

Parallel and distributed algorithms have become a necessity in modern machine learning tasks. In this work, **we focus on parallel asynchronous gradient descent and propose a zealous variant that minimizes the idle time of processors to achieve a substantial speedup**. We then experimentally study this algorithm in the context of training a restricted Boltzmann machine on a large collaborative filtering task.

Mini-batch gradient descent

Minimize $\mathbb{E}_z[C(\theta,z)]$ where ***C*** is some (typically convex) cost function and the expectation is computed over training points ***z***. In mini-batch gradient descent, this is achieved using the update rule

$$\theta_{k+1} \leftarrow \theta_k - \alpha \sum_{t=s_k}^{s_k+b} \frac{\partial C(\theta_k, z_t)}{\partial \theta}$$

where **α** is some learning rate and ***b*** is the number of training points in a mini-batch.

Asynchronous mini-batch gradient descent

Parallel mini-batch gradient descent with shared memory [1, 2, 3]:
- Store **θ** in shared memory.
- Have multiple processors process asynchronously and independently multiple mini-batches.
- Update **θ** in mutual exclusion using a synchronization lock.

Drawbacks:

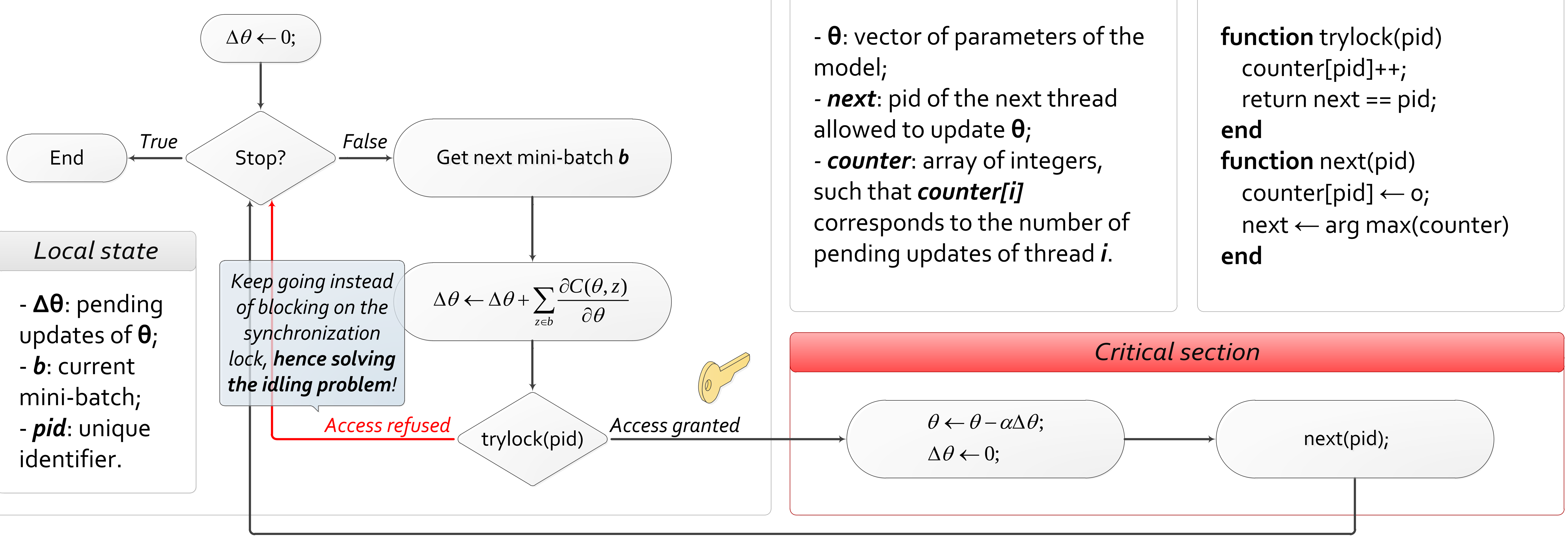
✗ Some delay might occur between the time gradient components are computed and the time they are eventually used to update **θ**. Hence, **processors might use stale parameters** that do not take into account the very last updates. Yet, [1, 4] showed that convergence is still guaranteed under some conditions.

✗ **Contention might appear on the synchronization lock, hence causing the processors to queue and idle.** This is likely to happen when updating **θ** takes a non-negligeable amount of time or as the number of processors increases.

This is what we address in this work.

Zealous parallel gradient descent algorithm

Procedure followed by each individual thread



Local state

- **Δθ**: pending updates of **θ**;
- ***b***: current mini-batch;
- ***pid***: unique identifier.

Global state

- **θ**: vector of parameters of the model;
- ***next***: pid of the next thread allowed to update **θ**;
- ***counter***: array of integers, such that ***counter[i]*** corresponds to the number of pending updates of thread ***i***.

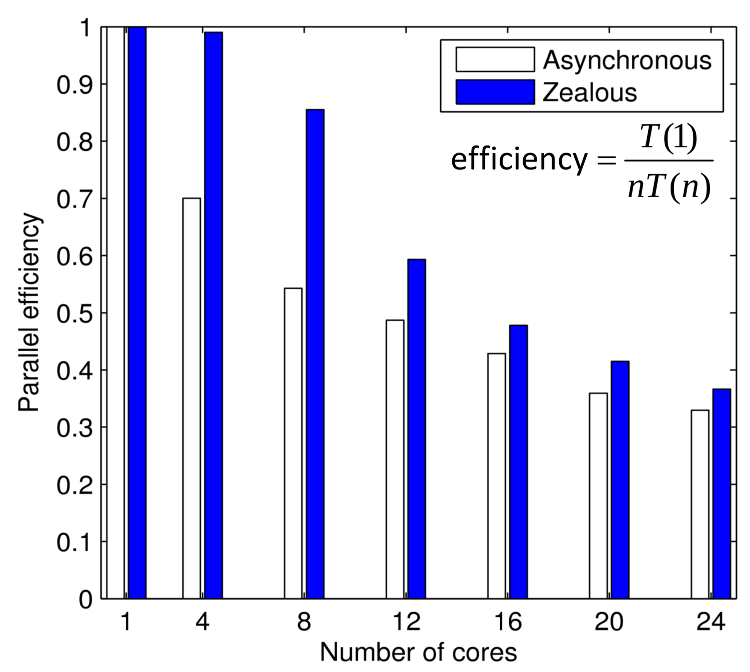
Policy functions

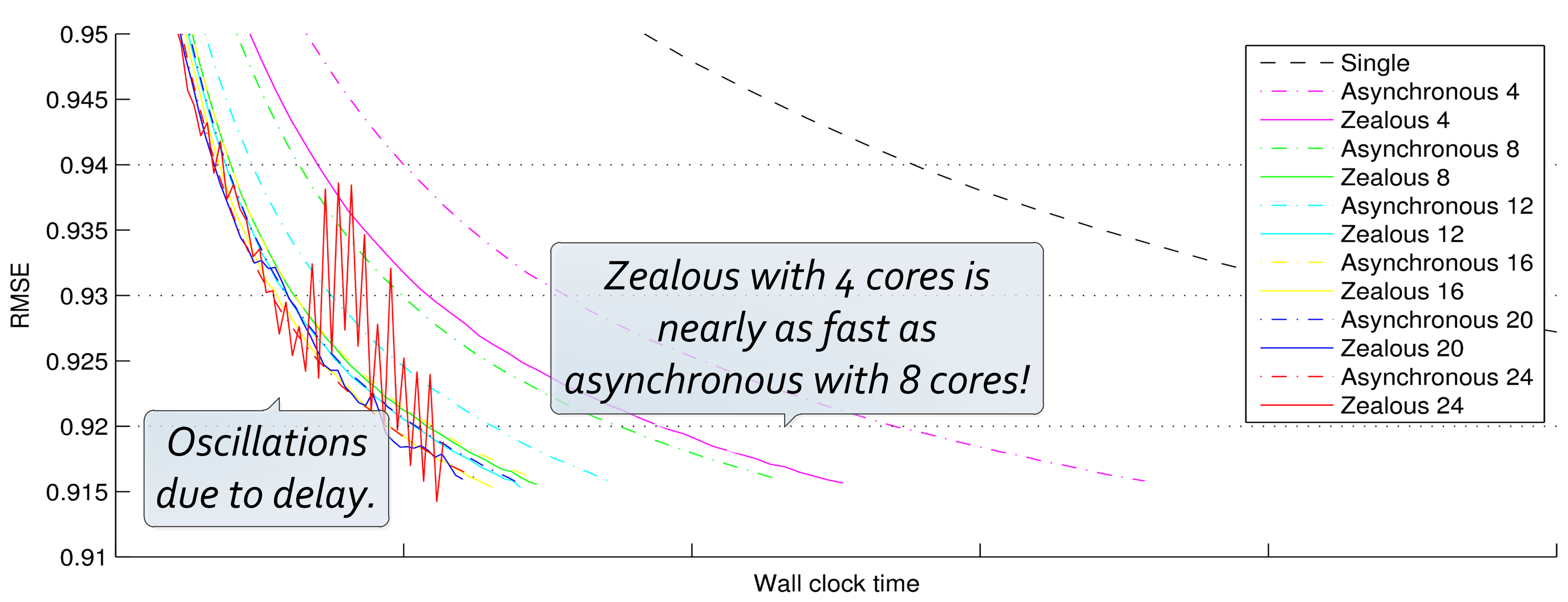
function trylock(pid)
  counter[pid]++;
  return next == pid;
end
function next(pid)
  counter[pid] ← 0;
  next ← arg max(counter)
end

Experimental results

Setting

- **Train a restricted Boltzmann machine** on a large collaborative filtering task [3, 5].
- **θ** counts 10M+ of values, hence executing **the critical section takes a fair amount of time**.
- Experiments carried out on a dedicated **24-core machine**.





Conclusions and future work

✓ **Significant speedup** over the asynchronous parallel gradient descent algorithm.
Future work: corroborate the results obtained in this work with more thorough experiments.

✗ Updates of **θ** may become too much delayed if the number of cores becomes too large, which can impair convergence.
Future work: Explore strategies to counter the effects of delay. Derive theoretical guarantees on the convergence of the algorithm.

References and acknowledgements

[1] A. Nedic, D.P. Bertsekas, and V.S. Borkar. ***Distributed asynchronous incremental subgradient methods***. Studies in Computational Mathematics, 8:381–407, 2001.
[2] K. Gimpel, D. Das, and N.A. Smith. ***Distributed asynchronous online learning for natural language processing***. In Proceedings of the Conference on Computational Natural Language Learning, 2010.
[3] G. Louppe. ***Collaborative filtering: Scalable approaches using restricted Boltzmann machines***. Master's thesis, University of Liège, 2010.
[4] M. Zinkevich, A. Smola, and J. Langford. ***Slow learners are fast***. In Advances in Neural Information Processing Systems 22, pages 2331–2339. 2009.
[5] R. Salakhutdinov, A. Mnih, and G. E. Hinton. ***Restricted Boltzmann machines for collaborative filtering***. In Proceedings of the 24th international conference on Machine learning, page 798. ACM, 2007.

Gilles Louppe and Pierre Geurts are respectively research fellow and research associate of the FNRS Belgium. This paper presents research results of the Belgian Network BIOMAGNET (Bioinformatics and Modeling: from Genomes to Networks), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors.