Interval-Asynchrony: Delimited Intervals of Localised Asynchrony for Fast Parallel SGD

Jacob Garby
<garby@chalmers.se>

Philippas Tsigas
<tsigas@chalmers.se>

relax
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Outline

I. The problem

II. Background

- SGD
- [A]synchrony

III. Interval-Asynchrony *(our contribution)*

- *The algorithm*
- *Results & Evaluation*

relax
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

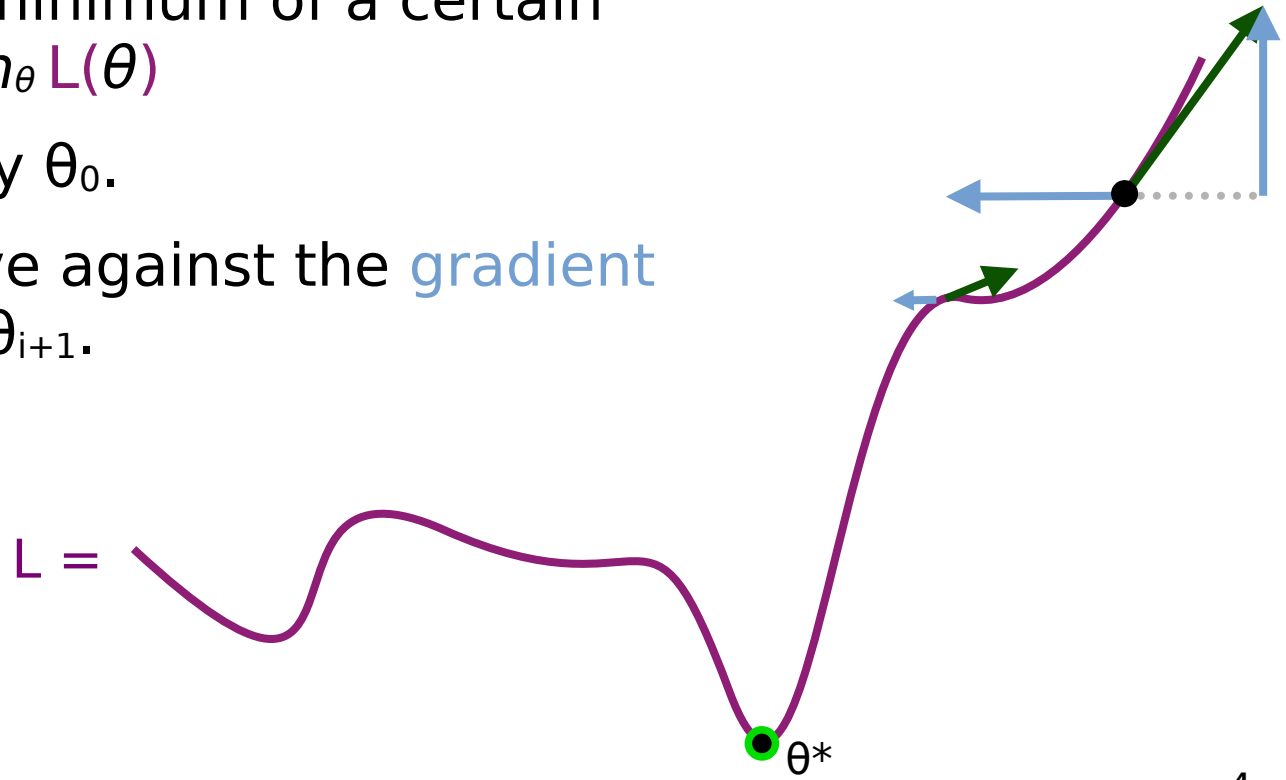
The Problem

- Scalable machine learning is important, with emphasis on **shared-memory many-core CPU systems**.
- Inter-thread synchronisation tends to introduce bottlenecks, and asynchronous designs lead to lower accuracy.
- How can we improve the scalability of machine learning training algorithms **(specifically SGD)** while maintaining good accuracy?

relax
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Quick Background: Gradient Descent

- We want to find the minimum of a certain function, $\theta^* = \textit{argmin}_\theta L(\theta)$
- First, pick an arbitrary θ_0.
- Then, iteratively move against the gradient of L at θ_i to produce θ_{i+1}.

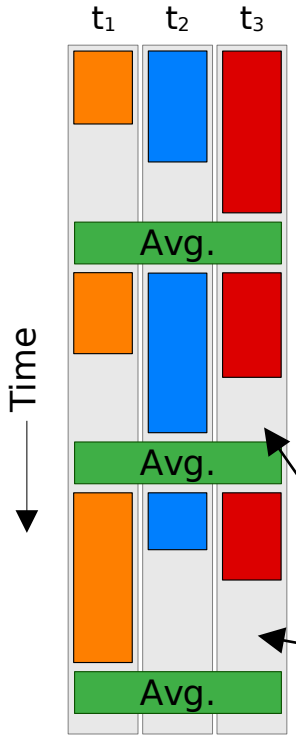


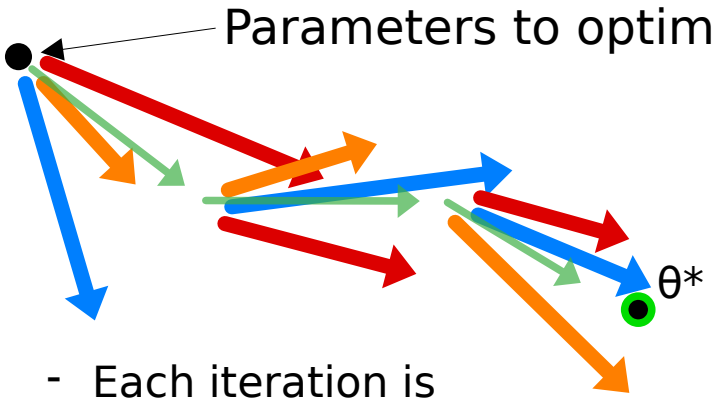
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Prior Work: How to Parallelise SGD?

Synchronous

Asynchronous



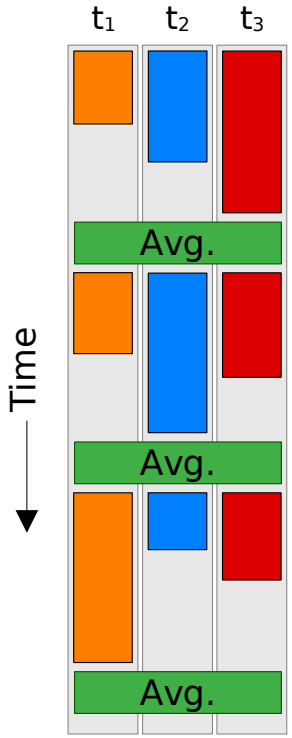


- Each iteration is **accurate**; follows sequential semantics.
- But, there is a lot of **idle time**. It could be faster!

relax
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

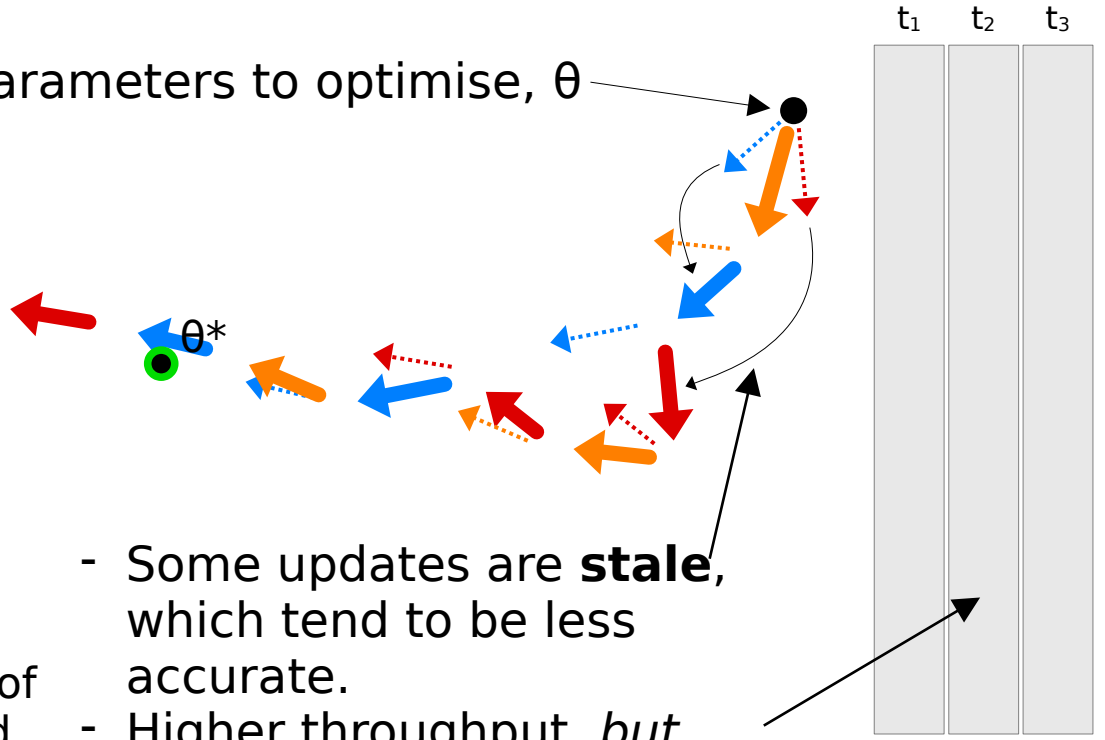
Prior Work: How to Parallelise SGD?

Synchronous



- Each iteration is **accurate**; follows sequential semantics.
- But, there is a lot of **idle time**. It could be faster!

Asynchronous



- Some updates are **stale**, which tend to be less accurate.
- Higher throughput, *but* **more steps** to converge.

relax
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Interval-Asynchrony

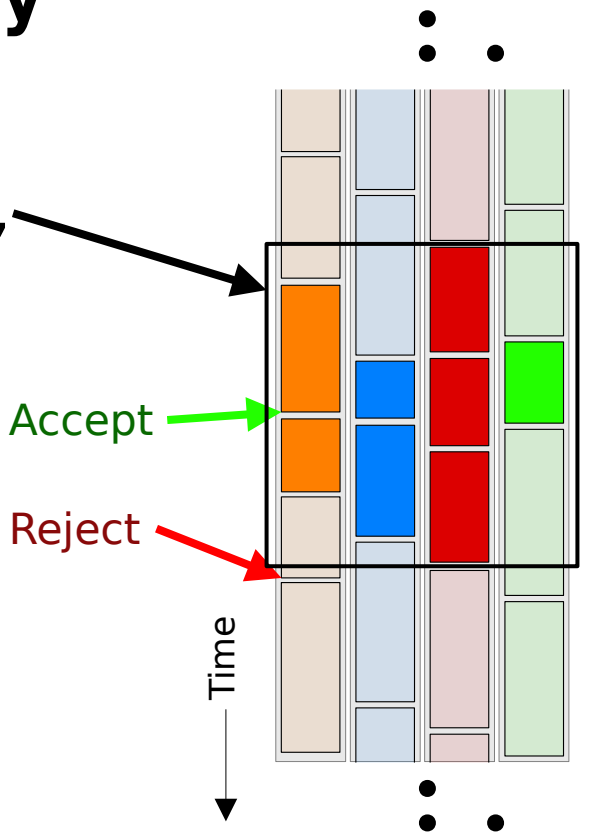
Proposal: A semi-asynchronous thread synchronisation scheme which can change *how asynchronous* it is during runtime.

- Semi-asynchrony may allow us to balance the throughput of async with the accuracy of synchronous execution.
- Runtime-adaptiveness is important since we found staleness is more impactful later in the execution.

relax
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Interval-Asynchrony

- **Idea:** A series of asynchronous intervals, defined by their size in number of updates.
  - SGD runs asynchronously within intervals, but an update may not cross an interval border.
  - Interval size is parameterised *(y)*, and can be changed at runtime.

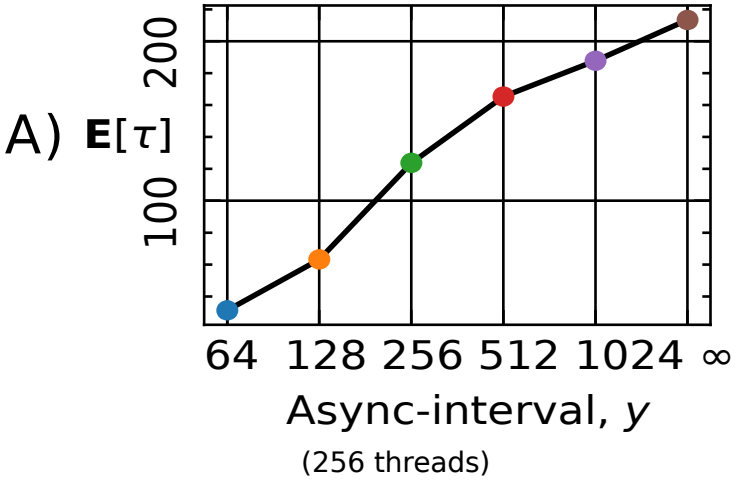


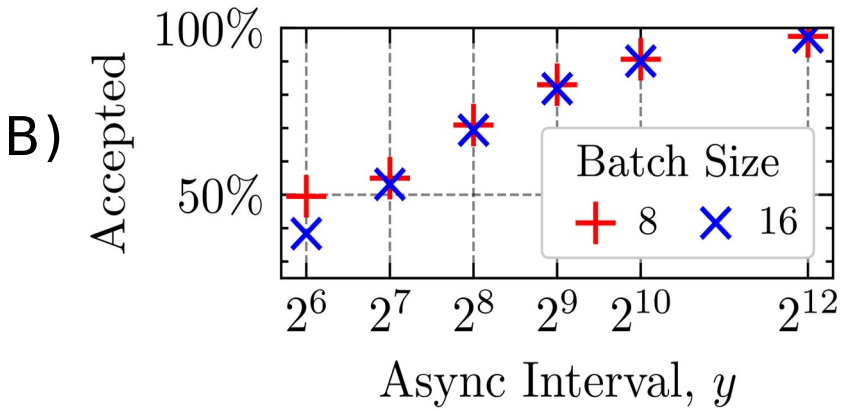
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Interval-Asynchrony: Interval Size

A) Staleness decreases, and is bounded ≤ y.

B) Fraction of accepted steps increases logarithmically with y.





Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Interval-Asynchrony: **Adjusting Interval Size**

Option 1: **Decay**

- Staleness is more impactful closer to convergence, so just reduce interval size over time.
- How to pick the rate though?

Option 2: **Probing**

- Attempt to automatically find time-varying optimal interval size by intermittently evaluating nearby values.
- Inspired by some recent work[1] on adjusting number of threads.

[1]K. Bäckström, 'Adaptiveness, Asynchrony, and Resource Efficiency in Parallel Stochastic Gradient Descent', Chapter D, Chalmers University of Technology, 2023. Available: https://research.chalmers.se/en/publication/535694

relax
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Evaluation

A: Does Interval-Asynchrony train a model to an acceptable accuracy faster than previous methods?

B: Does it scale well (i.e., is it able to utilise more threads without loss of accuracy)?

relax
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Overall Comparison to Baselines

	CIFAR-10				CIFAR-100			
	Accuracy		Time to 50.9%		Accuracy		Time to 5.1%	
	Best	Avg.	Best	Avg.	Best	Avg.	Best	Avg.
(Sync)	52.7%	52.7%	1071.5s	1760.7s	6.7%	6.7%	1485	>5400s
Async	50.9%	46.0%	43.9s	*	5.1%	4.5%	251.8s	*
ElAsync[1]	52.6%	49.0%	54.4s	*	5.5%	4.6%	264.5s	*
IntAsync	51.8%	50.1%	**30.0s**	**38.2s**	5.4%	5.1%	**193.3s**	**269.0s**

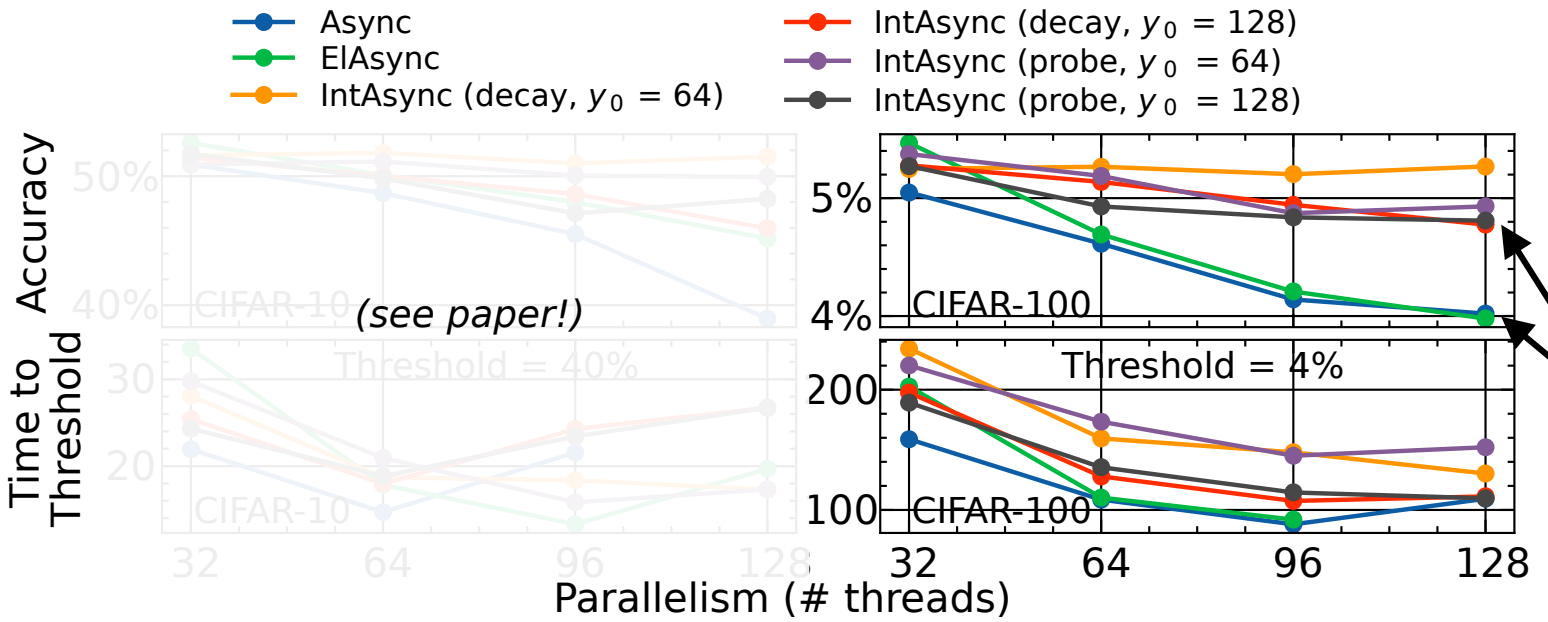
(see paper!)

*In these cases the target accuracy was never reached for some m.

[1]K. Bäckström, 'Adaptiveness, Asynchrony, and Resource Efficiency in Parallel Stochastic Gradient Descent', Chapter D, Chalmers University of Technology, 2023. Available: https://research.chalmers.se/en/publication/535694

Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Scalability Comparison



Interval-Asynchrony is scalable due to its lower staleness distribution.

Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

Conclusions

- We introduced a new algorithm for scheduling SGD threads, which is **semi-asynchronous** to a runtime-variable degree.
- We proposed two methods to **adjust the asynchrony** parameter (decay and probing).
- We showed its **superior scalability** against several baseline approaches.

relax
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456

relax
Funded by the European Union
CHALMERS | GÖTEBORGS UNIVERSITET
Engineering and Physical Sciences Research Council Grant number EP/X029174/1
Horizon Europe 2021-2027 Framework Programme Grant Agreement number 101072456